

Федеральное агентство по образованию и науке Российской Федерации  
Ярославский государственный педагогический  
университет имени К. Д. Ушинского

Учебное пособие по курсам

**Информатика и информационные технологии**  
**(для студентов специальности 050201.65 – «Математика»)**

Ярославль, 2009

Жохова Е.Ю., Заводчикова Н.И., Кокорева И.Е., Корнилов П.А., Московская Л.Я., Плясунова У.В., Семенова О.Г. Информатика и информационные технологии (для студентов специальности 050201.65 «Математика»). Учебное пособие. Ярославль: Изд-во ЯГПУ им. К.Д.Ушинского, 2009. 158 с.

Данная работа предназначена для студентов первого-второго курсов специальности «Математика Ярославского государственного педагогического университета. Кроме того, работа может быть полезна для других студентов других специальностей, изучающих информатику. Работа составлена на основе опыта преподавания информатики и информационных технологий в математике на физико-математическом факультете ЯГПУ.

Работа написана в соответствии с образовательным стандартом по курсам «Информатика» и «Информационные технологии в математике». Теоретический материал по ряду разделов курсов не был включен в работу из-за его большого объема

© Ярославский государственный педагогический университет имени К.Д.Ушинского, 2009

© Е.Ю. Жохова, Н.И. Заводчикова, И.Е. Кокорева, П.А. Корнилов, Л.Я. Московская, У.В. Плясунова, О.Г. Семенова

# Оглавление

|   |     |
|---|-----|
| ИНФОРМАЦИЯ. ЕДИНИЦЫ ИЗМЕРЕНИЯ ИНФОРМАЦИИ.....           | 4   |
| СИСТЕМЫ СЧИСЛЕНИЯ.....                                  | 10  |
| КОДИРОВАНИЕ ИНФОРМАЦИИ .....                            | 15  |
| ЭЛЕМЕНТЫ МАТЕМАТИЧЕСКОЙ ЛОГИКИ .....                    | 18  |
| РЕКУРРЕНТНЫЕ СООТНОШЕНИЯ .....                          | 30  |
| ОСНОВЫ ПРОГРАММИРОВАНИЯ.....                            | 37  |
| ЛАБОРАТОРНЫЕ РАБОТЫ ПО ОСНОВАМ ПРОГРАММИРОВАНИЯ.....    | 48  |
| ПРИМЕРНЫЕ ВОПРОСЫ К СОБЕСЕДОВАНИЯМ .....                | 87  |
| ТЕКСТЫ ПРОГРАММ ДЛЯ ВЫПОЛНЕНИЯ ЛАБОРАТОРНЫХ РАБОТ ..... | 115 |
| РЕШЕНИЕ ЗАДАЧ В ЭТ EXCEL .....                          | 121 |
| КОМПЬЮТЕРНАЯ МАТЕМАТИЧЕСКАЯ СИСТЕМА MATHCAD .....       | 128 |

## **Информация. Единицы измерения информации**

**Информация** – одно из базовых понятий информатики.

Наряду с материей и энергией информация является первичным понятием нашего мира и поэтому в строгом смысле не может быть определена. Можно лишь перечислить ее основные свойства, например, такие, как:

- 1) информация приносит сведения об окружающем нас мире, которых в рассматриваемой точке не было до ее получения;
- 2) информация не материальна, но она проявляется в форме материальных носителей дискретных знаков или сигналов;
- 3) знаки и сигналы несут информацию только для получателя, способного распознать их.

Вместе с тем слово «информация» является одним из тех терминов, которые достаточно часто встречаются не только в научных трудах специального характера, но и во множестве обиходных ситуаций и являются интуитивно понятными каждому человеку. При этом в узком, практическом смысле под **информацией** обычно понимают совокупность сведений об окружающем мире, являющихся объектом хранения, передачи и преобразования.

**Сообщение** - это информация, представленная в какой-либо форме (или совокупность знаков или сигналов содержащих информацию). Пример сообщений: текст телеграммы, данные на выходе ЭВМ, речь, музыка и т.д.

Алфавит – это вся совокупность знаков (символов), используемых в некотором языке для представления информации. Мощность алфавита – это число символов в нем.

Как определяют понятие «информации» учебники?

- Информация – сведения, знания, содержащиеся в сообщении.
- Информация – сведения об окружающих нас объектах, которые повышают уровень осведомленности человека.
- В бытовом смысле под информацией понимают любые данные или сведения, которые кого-либо интересуют.
- В научном смысле слово «информация» означает сведения об объектах и явлениях окружающей среды, их параметрах, свойствах и состоянии, которые воспринимают информационные системы (живые организмы, управляющие машины и др.) в процессе жизнедеятельности и работы.
- Применительно к компьютерной обработке данных под информацией понимают некоторую последовательность символических обозначений (букв, цифр, звуков, графиков, рисунков и т.п.), несущую смысловую нагрузку и представленную в понятном компьютеру виде.

Свойства любого объекта можно разделить на два больших класса: внешние и внутренние свойства.

**Внутренние свойства** – это свойства, органически присущие объекту. Они обычно «скрыты» от изучающего объект и проявляют себя косвенным образом при взаимодействии данного объекта с другими.

**Внешние свойства** – это свойства, характеризующие поведение объекта при взаимодействии с другими объектами.

Подобное деление свойств можно привести и для информации. Для любой информации можно указать три объекта взаимодействия: источник информации, приемник (получатель, потребитель) информации и объект или явление, которые данная информация представляет.

Важнейшие внешние свойства информации с точки зрения потребителя (приемника):

- **Понятность** – выражена на языке, понятном получателю.
- **Полнота** – имеются все необходимые данные для решения задачи или проблемы.
- **Достоверность** – логичность, непротиворечивость, не имеет скрытых ошибок.

- **Новизна** – содержит неизвестные для получателя сведения.
- **Актуальность** – своевременно (вовремя) доставлена.
- **Объективность** – не зависит от мнения конкретного человека.
- **Доступность** – возможность ее получения данным потребителем.
- **Защищенность** – характеризует невозможность несанкционированного использования или изменения.

Важнейшие свойства информации, связанные с процессом ее хранения:

- **Живучесть** – способность информации сохранять свое качество с течением времени.
- **Уникальной** называют информацию, хранящуюся в единственном экземпляре.

Среди внутренних свойств информации важнейшими являются внутренняя организация, структура и объем (количество) информации.

По способу ее внутренней организации информацию делят на две группы:

- Данные или простой, логически неупорядоченный набор сведений.
- Логически упорядоченные, организованные наборы данных.

Упорядоченность данных достигается наложением на данные некоторой структуры (отсюда часто используемый термин – структура данных).

Во второй группе выделяют особым образом организованную информацию – *знания*. Знания, в отличие от данных, представляют собой информацию не о каком-то единичном и конкретном факте, а о том, как устроены все факты определенного типа.

Как измерить объем (количество) информации?

Для информатики информация играет такую же роль, как и вещество в физике. И подобно тому, как веществу можно приписывать довольно большое количество характеристик: массу, заряд, объем и т. д., так и для информации имеется пусть и не столь большой, но достаточно большой набор характеристик. Как и для характеристик вещества, так и для характеристик информации имеются единицы измерения, что позволяет некоторой порции информации приписывать числа – *количественные характеристики информации*.

На сегодняшний день используются различные способы измерения количества информации.

**Объемный** (или **алфавитный**) – это самый простой и грубый способ измерения информации. Соответствующую количественную оценку информации естественно назвать объемом информации. **Объем информации в сообщении – это количество символов в сообщении**. Поскольку, например, одно и то же число может быть записано разными способами (с использованием разных алфавитов), то этот способ чувствителен к форме представления (записи) сообщения. При таком подходе считается, что каждый символ алфавита имеет определенный «информационный вес». «Информационный вес» символа зависит от мощности алфавита (числа символов в алфавите). В вычислительной технике вся обрабатываемая и хранимая информация вне зависимости от ее природы (число, текст, рисунок, звук и т. д.) представлена в двоичной системе счисления (с использованием алфавита, состоящего из двух символов 0 и 1). Информационный вес символа двоичного алфавита принят за единицу измерения количества информации и называется **1 бит**. Полное количество информации, заключенное в тексте, записанном с помощью двоичного алфавита, равно числу нулей и единиц в этом тексте и выражается это количество соответствующим числом бит.

Для измерения больших объемов информации используются более крупные единицы:

1 байт = 8 бит

1 Кбайт = 1 Кб = 1 килобайт =  $2^{10}$  байт = 1024 байт  $\approx$  1000 байт

1 Мбайт = 1 Мб = 1 мегабайт =  $2^{10}$  Кбайт = 1024 Кбайт  $\approx$  1000 Кбайт

1 Гбайт = 1 Гб = 1 гигабайт =  $2^{10}$  Мбайт = 1024 Мбайт  $\approx$  1000 Мбайт

## Примеры задач

### 1. Минимальной единицей измерения количества информации является

а) байт б) гигабайт в) бит г) символ

Правильный ответ: в.

**2. Сколько мегабайт информации содержится в сообщении объемом  $2^{24}$  байт?**

- а) 64 б) 8 в) 16 г) 4

Решение:

$$2^{24} \text{ байт} = \frac{2^{24}}{2^{10}} \text{ Кбайт} = \frac{2^{24}}{2^{10} \times 2^{10}} \text{ Мбайт} = 2^4 \text{ Мбайт} = 16 \text{ Мбайт}$$

Правильный ответ: в.

**3. Информационному сообщению объемом 12 288 бит соответствует**

- а) 1536 Кбайт б) 1,5 Кбайт в) 1,536 Кбайт г) 12 Кбайт д) 1,5 Гбайт

Решение:

- 1) объем сообщения в байтах  $12288 \text{ (бит)} : 8 = 1536 \text{ (байт)}$
- 2) объем сообщения в Кбайтах  $\approx 1536 \text{ (байт)} : 1000 = 1,536 \text{ (Кбайт)}$

Правильный ответ: в.

**4. Сколько бит информации содержится в сообщении объемом четверть килобайта?**

- а) 2032 б) 2048 в) 250 г) 2000 д) 256

Решение:

$$\frac{1}{4} \text{ (Кбайт)} = \frac{1}{4} \times 1024 \text{ (байт)} = \frac{1}{4} \times 1024 \times 8 \text{ (бит)} = \frac{2^{10} \times 2^3}{2^2} \text{ (бит)} = 2^{11} \text{ (бит)} = 2048 \text{ (бит)}$$

Правильный ответ: б.

**5. Сколько бит информации в слове «байт» (информационный вес символа – 1 байт)?**

- а) 8 б) 32 в) 1024 г) 4

Решение:

- 1) **информационный вес символа** 1 байт = 8 бит
- 2) в слове «байт» 4 символа, следовательно, в заданном слове  $8 \text{ бит} \times 4 = 32 \text{ бита}$  информации.

Правильный ответ: б.

### ***Содержательный подход к измерению количества информации***

Если рассматривается система, которая может принимать одно из  $N$  возможных состояний, то актуальна задача оценки выбора одного из состояний. Такой оценкой будет мера неопределенности (*количество*) *информации*. Неопределенность знаний о выборе одного из  $N$  состояний – это количество возможных состояний системы.

Эту интуитивно ясную связь между обыденными представлениями об информации и о неопределенности можно сформулировать так: *появление информации суть уменьшение неопределенности*.

Пусть имеются монета и игральный кубик.

У монеты две стороны: «герб» (Г) и «решка» (Р). Если бросить ее на стол, монета обязательно упадет вверх либо гербом, либо решкой. Таким образом, монету, лежащую на столе, можно рассматривать как простейшую систему, которая может находиться в одном из двух возможных состояний.

Игральный кубик – это кубик, грани которого пронумерованы от 1 до 6. Аналогично монете игральный кубик, лежащий на столе, – это система, находящаяся в одном из шести возможных состояний (по номерам граней, обращенных вверх).

Здесь предполагается, что стол имеет горизонтальную, твердую и гладкую поверхность: только в этих условиях монета и кубик обязательно будут занимать на нем одно из возможных состояний (если же, например, стол покрыт пластилином, то монета может встать на ребро).

Выбор одного из возможных состояний ликвидирует неопределенность, создавая тем самым информацию, при этом происходит «запоминание» результата выбора: попав в определенное состояние, система продолжает там оставаться.

Важнейшей особенностью рассматриваемых систем является то, что выбор происходит случайно, причем все состояния равновозможные (проверить, так ли это на самом деле, можно,

проделав большее число бросаний, регистрируя каждый раз состояние, в котором оказалась система).

Однако и в случае монеты, и в случае кубика мера неопределенности выбора, которая существует перед возникновением системы, различна. Действительно, при подбрасывании монеты происходит выбор всего лишь из двух возможных состояний системы, а при подбрасывании кубика выбирается одно из шести возможных состояний. Перед выпадением монеты неопределенным является одно состояние из двух, а перед выпадением кубика – одно из шести, т.е. степень неопределенности в случае с кубиком больше.

В общем, чем больше число возможных состояний системы, тем больше мера неопределенности, и наоборот, т.е. мера неопределенности выбора есть возрастающая функция числа равновероятных состояний системы. Если обозначить меру неопределенности выбора для некоторой системы как  $I(N)$ , где  $N$  – число возможных состояний системы, то данную связь можно записать следующим образом:

$$\text{если } N_1 < N_2, \text{ то } I(N_1) < I(N_2).$$

Отсюда ясно, что минимальная мера неопределенности выбора будет у систем с наименьшим числом состояний.

В качестве единицы измерения меры неопределенности примем меру неопределенности для системы с минимальным числом состояний. Поэтому в качестве единицы возьмем меру неопределенности для системы с двумя возможными состояниями, т.е. положим

$$I(2) = 1.$$

В соответствии с этим мера неопределенности для системы «монета» равна 1.

Пусть теперь у нас имеются две монеты. Будем считать, что мера неопределенности для системы, состоящей из двух монет, в 2 раза больше, чем мера неопределенности для системы, состоящей из одной монеты, т.е. равна 4. Две монеты могут лежать на столе 4 способами:

$$ГГ, ГР, РГ, РР,$$

где буквами обозначены стороны монет, обращенных вверх. Значит, число состояний такой системы равно 4. Имеем следующее соотношение:

$$I(4) = 2.$$

Рассуждая аналогично, для системы из  $N = 2^k$  монет получим:

$$I(2^k) = k.$$

Учитывая, что  $I(N) = I(2^k) = k = \log_2 2^k = \log_2 N$ , получим с учетом сделанных допущений:

$$I(N) = \log_2 N.$$

Указанная мера была предложена американским ученым Р Хартли в 1928 году: если во множестве  $X = \{x_1, x_2, \dots, x_N\}$  выделить произвольный элемент  $x_i \in X$ , то чтобы найти его, необходимо получить не менее  $\log_2 N$  (единиц) информации.

При этом единица неопределенности по предложению Гьюки называется *двоичной единицей* или *битом* и представляет собой неопределенность выбора из двух равновероятных событий.

Мера неопределенности для системы «монета» равна 1.

Результат бросания игрального кубика – это количество информации

$$I = \log_2 6 = 2,58496 \text{ (с точностью до 5 знаков после запятой).}$$

Мера неопределенности удовлетворяет ряду естественных условий. Одним из них является необходимость монотонного возрастания с увеличением возможностей выбора, т.е. числа возможных состояний источника  $N$ . Кроме того, мера неопределенности отвечает *требованию аддитивности*, состоящему в следующем.

Если два независимых источника с числом равновероятных состояний  $N$  и  $M$  рассматривать как один источник, то неопределенность объединенного источника должна равняться сумме неопределенностей исходных источников.

Предложенная мера позволяет решать определенные практические задачи. Однако она рассчитана на слишком грубую модель источника информации, приписывающую всем его возможным состояниям одинаковую вероятность.

Итак, неопределенность знаний о некотором событии – это количество возможных результатов события. Сообщение, уменьшающее неопределенность знаний в 2 раза, несет 1 бит информации.

Для определения количества информации  $I$ , содержащейся в сообщении о том, что произошло одно из  $N$  равновозможных событий, нужно решить показательное уравнение:

$$2^I = N, \text{ откуда } I = \log_2 N.$$

**Примеры задач.**

**6. Какое количество информации несет в себе сообщение о том, что нужная вам программа находится на одной из восьми дискет?**

- а) 2 бит   б) 3 бит   в) 4 бит   г) 8 бит

Решение:

Количество возможных результатов события  $N = 8$

Количества информации определяется по формуле  $I = \log_2 N = \log_2 8 = 3$  (бита)

Правильный ответ: б.

**7. При угадывании целого числа в некотором диапазоне было получено 6 бит информации. Сколько чисел содержит этот диапазон?**

Решение:

$$I = 6 \text{ бит}$$

$$N = 2^I, 2^6 = 64 \text{ (чисел).}$$

**Ответ:** 64 числа.

**8. Какое количество информации требуется для двоичного кодирования одного символа алфавита, состоящего из 256 символов?**

- А) 1 БАЙТ   Б) 4 БИТ   В) 16 БИТ   Г) 2 БАЙТА

Решение:

Мощность алфавита  $N = 256$

$$I = \log_2 N = \log_2 256 = 8 \text{ (бит)} = 1 \text{ байт}$$

Правильный ответ: а.

**9. Какое количество информации содержит один разряд шестнадцатеричного числа?**

- а) 4 бита   б) 1 бит   в) 1 байт   4) 16 бит

Решение:

Мощность алфавита шестнадцатеричной системы счисления  $N = 16$ .

$$I = \log_2 16 = 4 \text{ (бита)}$$

Правильный ответ: а.

**10. Какое количество информации получит второй игрок при игре в крестики-нолики на поле  $16 \times 16$ , после первого хода первого игрока, играющего крестиками?**

- а) 5 бит   б) 6 бит   в) 7 бит   г) 8 бит

Решение:

Число полей, в которые может поставить крестик первый игрок первым ходом

$$N = 16 \times 16 = 256$$

$$\text{Количество информации } I = \log_2 N = \log_2 256 = 8 \text{ (бит)}$$

Правильный ответ: г.

**11. В школьной библиотеке 64 стеллажа с книгами. На каждом стеллаже 8 полок. Библиотекарь сообщил, что нужная Вам книга находится на пятом стеллаже на третьей сверху полке. Какое количество информации передал библиотекарь?**

Решение.

Сообщение содержит две группы равновозможных событий:

1) выбор одного из 64 стеллажей; 2) выбор одной из 8 полок стеллажа.

Неопределенность объединения этих групп есть сумма неопределенностей каждой группы, т.е. количество информации, которое передал библиотекарь, находится по формуле:  $\log_2 64 + \log_2 8 = 6 + 3 = 9$  (бит).

**Ответ:** 9 бит.



**12. В процессе преобразования растрового графического изображения количество цветов уменьшилось с 65536 до 256. Во сколько раз уменьшится объем, занимаемый им в памяти?**

- а) в 2 раза    б) в 8 раз    в) в 16 раз    г) в 256 раз

Решение:

До преобразования количество цветов  $N_1 = 65536$

Информационный вес одного цвета (количество информации)  $I_1 = \log_2 N_1 = \log_2 65536 = 16$  (бит)

После преобразования количество цветов  $N_2 = 256$

Информационный вес одного цвета в этом случае  $I_2 = \log_2 N_2 = \log_2 256 = 8$  (бит)

Во сколько раз изменился информационный вес?  $16 \text{ бит} : 8 \text{ бит} = 2$  (раза), т. е. в 2 раза уменьшился, т. е. объем всего изображения уменьшится в 2 раза.

Правильный ответ: а.

**13. Информационное сообщение объемом 1,5 Кбайта содержит 3072 символа, кодируемых одинаковым числом бит. Какое наибольшее количество символов может содержать алфавит, с помощью которого было записано это сообщение?**

Решение.

Объем сообщения 1,5 Кбайт =  $1,5 \times 1024 \times 8$  (бит) = 12288 бит

Информационный вес символа в сообщении можно вычислить следующим образом:

$$\text{Информационный вес} = I = \frac{\text{Объем сообщения}}{\text{количество символов}} = \frac{12288}{3072} = 4 \text{ (бит)}$$

Какое количество символов можно закодировать 4 битами?

Используя формулу Хартли, вычислим  $N$  – мощность алфавита сообщения

$$N = 2^I = 2^4 = 16 \text{ (символов).}$$

Ответ: 16 символов.

**14. Для хранения текста требуется 10 Кбайт. Сколько страниц займет этот текст, если на странице размещается 40 строк по 64 символа в строке?**

Решение.

Объем текста 10 Кбайт =  $10 \times 1024 = 10240$  (байт)

Информационный вес каждого символа текста – 1 байт.

Размер страницы  $64 \times 40 \times 1 = 2560$  (байт)

$$\text{Число страниц текста} = \text{объем текста} / \text{размер страницы} = 10240 / 2560 = 4 \text{ (страницы)}$$

Ответ: 4 страницы.

**15. Видеопамять компьютера имеет объем 500 Кбайт. Размер графической сетки  $640 \times 200$ . Сколько страниц экрана одновременно разместится в видеопамяти при палитре из 16 цветов?**

Решение.

Для кодирования 16 различных цветов требуется  $I = \log_2 16 = 4$  (бит).

Количество пикселей на экране  $640 \times 200 = 128000$ .

Объем страницы экрана (палитра 16 цветов)  $4 \text{ (бит)} \times 128000 = 512000$  (бит)

Объем видеопамяти 500 Кбайт =  $500 \times 1024 \times 8 = 4096000$  (бит)

Количество страниц экрана = объем видеопамяти / объем страницы экрана =  $4096000 / 512000 = 8$  (страниц).

Ответ: 8 страниц.

## Системы счисления

**Система счисления** – это совокупность приемов наименования и записи чисел с помощью знаков (цифр) и соответствующие им правила действия над числами.

**Основание системы счисления** – это количество знаков (цифр), с помощью которых производится запись чисел. Если основание системы  $p$ , то система счисления называется  $p$ -ичной и для записи чисел в  $p$ -ичной системе счисления необходимо  $p$  различных цифр.

**Алфавит** системы счисления – это множество различных цифр, используемых для записи чисел.

| Система счисления | Основание | Количество цифр | Цифры  |
|-------------------|-----------|-----------------|--|
| Двоичная          | 2         | 2               | 0, 1   |
| Пятеричная        | 5         | 5               | 0, 1, 2, 3, 4                                  |
| Восьмеричная      | 8         | 8               | 0, 1, 2, 3, 4, 5, 6, 7                         |
| Десятичная        | 10        | 10              | 0, 1, 2, 3, 4, 5, 6, 7, 8, 9                   |
| Шестнадцатеричная | 16        | 16              | 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F |

Наиболее совершенными являются позиционные системы счисления, т. е. системы, в которых значение каждой цифры зависит от ее положения (позиции) в записи числа. Основанием позиционной системы счисления может быть любое натуральное число, большее или равное двум.

В  $p$ -ичной системе счисления единицами разрядов служат последовательные степени числа  $p$ , иначе говоря,  $p$  единиц какого-либо разряда образуют единицу следующего старшего разряда.

Натуральное число  $a$  в системе счисления можно записать как

$$a = a_n p^n + a_{n-1} p^{n-1} + \dots + a_1 p + a_0 p^0, \quad \text{где } 0 \leq a_i < p, i = 0, 1, \dots, n. \quad (1)$$

Правильную конечную  $p$ -ичную дробь  $b$  можно записать в виде

$$b = b_{-1} p^{-1} + b_{-2} p^{-2} + \dots + b_{-k} p^{-k}, 0 \leq b_i < p, i = -1, -2, \dots, -k. \quad (2)$$

В формулах коэффициенты  $a_i, b_i$  при степенях основания  $p$  являются цифрами в данной позиционной системе счисления. В общем виде любое действительное число  $X$  в позиционной системе счисления с основанием  $p$  можно представить формулой

$$X = a_n p^n + a_{n-1} p^{n-1} + \dots + a_1 p + a_0 + b_{-1} p^{-1} + b_{-2} p^{-2} + \dots + b_{-k} p^{-k} \dots \quad (3)$$

С другой стороны, любое число  $p$ -ичной системы счисления можно записать в виде последовательного перечисления его цифр, начиная со старшей, т. е.

$$X = a_n a_{n-1} \dots a_1 a_0 b_{-1} b_{-2} \dots b_{-k} \dots \quad (4)$$

Представление числа в  $p$ -ичной системе счисления в виде (3) называется **развернутой формой** записи числа.

Представление числа в  $p$ -ичной системе счисления в виде (4) называется **свернутой формой** записи числа.

Арифметические действия над числами в любой позиционной системе счисления производятся по тем же правилам, что и в десятичной системе счисления, так как все они основываются на правилах выполнения действий над соответствующими полиномами. При этом нужно только пользоваться теми таблицами сложения и умножения, которые имеют место при данном основании  $p$  системы счисления.

### Двоичная арифметика

Для того, чтобы производить арифметические операции над двоичными числами, надо знать таблицы сложения и умножения в двоичной системе счисления.

Таблица сложения:

|   |   |    |
|---|---|----|
| + | 0 | 1  |
| 0 | 0 | 1  |
| 1 | 1 | 10 |

Таблица умножения:

|   |   |   |
|---|---|---|
| × | 0 | 1 |
| 0 | 0 | 0 |
| 1 | 0 | 1 |

### Запись двоичного числа в системе счисления с основанием $p = 2^n$

Для записи целого двоичного числа в системе счисления  $p = 2^n$  достаточно данное двоичное число разбить на группы справа налево (т. е. от младших разрядов к старшим) по  $n$  цифр в каждой группе. Затем каждую такую группу следует рассмотреть как  $n$ -разрядное двоичное число и записать его как цифру в системе счисления с основанием  $p = 2^n$ .

Для записи правильной двоичной дроби в системе счисления  $p = 2^n$  достаточно данное двоичное число разбить на группы слева направо (т. е. от старших разрядов к младшим) по  $n$  цифр в каждой группе. Затем каждую такую группу следует рассмотреть как  $n$ -разрядное двоичное число и записать его как цифру в системе счисления с основанием  $p = 2^n$ .

Для замены числа, записанного в системе счисления с основанием  $p = 2^n$ , равным ему числом в двоичной системе счисления достаточно каждую цифру данного числа заменить  $n$ -разрядным двоичным числом.

**Пример 1.** Число  $10001010101,011010111_2$  заменить равным ему числом шестнадцатеричной системы счисления.

Решение.

Так как  $p = 2^n = 16$ , следовательно,  $n=4$ .

| Двоичное число | Цифра шестнадцатеричной системы счисления | Двоичное число | Цифра шестнадцатеричной системы счисления |
|----------------|---|----------------|---|
| 0000           | 0   | 1000           | 8   |
| 0001           | 1   | 1001           | 9   |
| 0010           | 2   | 1010           | A   |
| 0011           | 3   | 1011           | B   |
| 0100           | 4   | 1100           | C   |
| 0101           | 5   | 1101           | D   |
| 0110           | 6   | 1110           | E   |
| 0111           | 7   | 1111           | F   |

Разбиваем двоичное число на группу по 4 цифры в каждой

100 0101 0101 , 0110 1011 1<sub>2</sub>

Недостающие позиции слева и справа дополняем соответствующим числом нулей

0100 0101 0101 , 0110 1011 1000<sub>2</sub>

4 5 5 , 6 B 8<sub>16</sub>

Каждую группу рассматриваем как 4-х разрядное двоичное число и записываем его как цифру в системе счисления с основанием  $p = 16$ .

Ответ:  $455,6B8_{16}$ .

**Пример 2.** Число  $1234,56_8$  преобразовать в шестнадцатеричное.

Решение.

Преобразование выполняем через двоичную систему счисления.

Преобразуем  $1234,56_8$  в соответствующее ему двоичное число.

Так как основание системы счисления равно 8, а  $8 = 2^3$ , то каждую цифру данного числа заменим 3-х разрядным двоичным числом (для этого можно воспользоваться приведенной выше таблицей, отбросив «лишний» старший разряд двоичного числа).

1 2 3 4 , 5 6<sub>8</sub>

001 010 011 100 , 101 110<sub>2</sub>

Далее двоичное число записываем в шестнадцатеричной системе счисления (пример 1).

001010011100,101110<sub>2</sub>

0010 1001 1100, 1011 1000<sub>2</sub>  
 2 9 C B 8<sub>16</sub>  
 Ответ: 29C, B8<sub>16</sub>.

### Преобразование $p$ -ичного числа в десятичное

Замена  $p$ -ичного числа соответствующим ему десятичным, производится путем вычисления в десятичной системе счисления выражения в развернутой форме записи  $p$ -ичного числа.

**Пример 3. Преобразовать 10111,011<sub>2</sub> в десятичное.**

Решение.

Запишем развернутую форму заданного двоичного числа и вычислим выражение

$$10111,011_2 = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} + 1 \times 2^{-3} = 16 + 4 + 2 + 1 +$$

$$\frac{1}{4} + \frac{1}{8} = 23,375$$

Вычисления можно производить по следующей схеме (схема Горнера):

$$10111,011_2 = (((((1 \times 2 + 0) \times 2) + 1) \times 2 + 1) \times 2) + 1) + (((0 \times 1/2 + 1) \times 1/2 + 1) \times 1/2) = 23,375$$

### Преобразование десятичного числа в $p$ -ичное

Преобразование целого десятичного числа в  $p$ -ичное производится путем выполнения цепочки делений с остатком на  $p$ . Деление продолжается до тех пор, пока в частном не получим 0, на каждом новом шаге новое делимое есть частное, полученное на предыдущем шаге. Полученные остатки выписываем в обратном порядке (остаток, полученный на первом шаге – младший разряд  $p$ -ичного числа и т. д.).

**Пример 4. Преобразовать 456 в равное ему восьмеричное число.**

Решение.

$$\begin{array}{r}
 456 \overline{) 8} \\
 - \quad \underline{57} \overline{) 8} \\
 456 - \quad \underline{7} \overline{) 8} \\
 \textcircled{0} \quad 56 - \quad \underline{\quad} \\
 \quad \textcircled{1} \quad \underline{\quad} \\
 \quad \quad \textcircled{7} \quad \underline{\quad} \\
 \quad \quad \quad 0
 \end{array}$$

$$456 = 710_8$$

Ответ: 710<sub>8</sub>.

Описанный выше алгоритм преобразования можно заменить альтернативной схемой: представить десятичное число степенями основания  $p$  с соответствующими коэффициентами (цифрами  $p$ -ичной системы счисления), записать полученный результат в свернутой форме.

**Пример 5. Преобразовать 456<sub>8</sub> в равное ему десятичное число.**

Решение.

$$\begin{array}{ll}
 7 \times 8^2 = 7 \times 64 = 448 & \text{ост. } 456 - 448 = 8 \\
 1 \times 8 = 8 & \text{ост. } 8 - 8 = 0
 \end{array}$$

Получено 3-разрядное восьмеричное число и его свернутая форма 710<sub>8</sub>.

Преобразование правильной десятичной дроби в  $p$ -ичную производится путем последовательных умножений на  $p$  с выделением цифр произведений в качестве искомым.

**Пример 6. Перевести число 0,46875 из десятичной системы счисления в шестнадцатеричную.**

Решение.

$$\begin{array}{r}
 \textcircled{0} \quad \overline{) 46875 \times 16} \\
 \textcircled{7} \quad \overline{) 5 \times 16}
 \end{array}$$

В свернутой форме  $0,46875 = 0,78_{16}$ .

**Пример 7. В системе счисления с некоторым основанием десятичное число 132 записано в виде 110. Укажите это основание.**

Решение.

$$132 = 110_p$$

Разложим правую часть по степеням основания  $p$  неизвестной системы счисления

$$132 = 1 \times p^2 + 1 \times p^1 + 0 \times p^0$$

$$p^2 + p - 132 = 0$$

Решения данного квадратного уравнения

$$p_{1,2} = \frac{-1 \pm \sqrt{1 + 4 \times 132}}{2} = \frac{-1 \pm 23}{2}$$

$$p_1 = 11$$

$$p_2 = -12, \text{ что невозможно.}$$

Следовательно,  $p = 11$ .

**Пример 8. Один школьный учитель на вопрос, много ли у него в классе учеников, ответил: «У меня в классе 100 детей, из них 24 мальчика и 32 девочки». В какой системе счисления производит вычисления учитель?**

Решение.

1 способ. См. решение примера 1, т. е. требуется записать разложение по степеням основания  $p$  неизвестной системы счисления левой и правой части,

$$100_p = 24_p + 32_p, \text{ и далее найти решения квадратного уравнения.}$$

2 способ. Сложить два числа. Очевидно, что счет ведется «шестерками», следовательно,

$$\begin{array}{r} 24 \\ + \\ 32 \\ \hline 100 \end{array}$$

Следовательно, учитель использовал систему счисления с основанием 6.

**Пример 9. Преобразовать десятичную дробь 634,34375 в шестнадцатеричную.**

Решение.

Запишем десятичную дробь в виде целой и дробной части  $634,34375 = 634 + 0,34375$ .

Преобразуем целую часть число 634 в шестнадцатеричное

$$2 \times 16^2 = 2 \times 256 = 512 \quad \text{ост. } 634 - 512 = 122$$

$$7 \times 16^1 = 7 \times 16 = 112 \quad \text{ост. } 122 - 112 = 10$$

$$10 \times 16^0 = 10 \quad \text{ост. } 10 - 10 = 0$$

Выписываем полученный результат  $634 = 27A_{16}$

Преобразуем дробную часть число 0,34375 в шестнадцатеричное число

$$\begin{array}{r|l} 1 & 34375 \times 16 \\ 5 & 5 \times 16 \\ 8 & 0 \end{array}$$

Выписываем полученный результат  $0,34375 = 0,58_{16}$ .

Объединяем целую и дробную части  $634,34375 = 27A,58_{16}$ .

Ответ:  $27A,58_{16}$ .

**Пример 10. Выполнить действия  $(101,11_2 + B5,4C_{16}) \times 0,11_2 - 26,72_8$  (ответ записать в восьмеричной системе счисления).**

Решение.

1 способ. Все вычисления производим в двоичной системе счисления, а полученный результат представим в восьмеричной системе счисления.

а)  $B5,4C_{16} = 10110101,01001100_2$

б)  $26,72_8 = 10110,111010_2$

$$\begin{array}{r} \text{в) } 10110101,01001100_2 \\ + \quad 101,11_2 \\ \hline 10111011,00001100_2 \end{array}$$

$$\begin{array}{r} \text{г) } 10111011,000011_2 \\ \times \quad 0,11_2 \\ \hline 10111011000011 \\ + \quad 10111011000011 \\ \hline 10001100,01001001_2 \end{array}$$

$$\begin{array}{r} \text{д) } 10001100,01001001_2 \\ - \quad 10110,11101_2 \\ \hline 1110101,01100001_2 \end{array}$$

е)  $\underbrace{1\ 110\ 101}_1, \underbrace{011\ 000\ 010}_3_2$

$1\ 6\ 5, 3\ 0\ 2_8$

Ответ:  $165,302_8$ .

2 способ. Вычисления производим в двоичной и восьмеричной системах счисления.

а) см. 1 способ пункт а)

б) см. 1 способ пункт в)

в)  $\underbrace{010\ 111\ 011}_2, \underbrace{000\ 011\ 000}_3_2$

$2\ 7\ 3, 0\ 3_8$ , т. е. результат сложения двоичных чисел преобразовали в восьмеричное число.

г)  $0, \underbrace{110}_2$  преобразование в в восьмеричное число

$0, 6_8$

г) пользуйсь таблицей умножения восьмеричных чисел,

| × | 1 | 2  | 3  | 4  | 5  | 6  | 7  |
|---|---|----|----|----|----|----|----|
| 1 | 1 | 2  | 3  | 4  | 5  | 6  | 7  |
| 2 | 2 | 4  | 6  | 10 | 12 | 14 | 16 |
| 3 | 3 | 6  | 11 | 14 | 17 | 22 | 25 |
| 4 | 4 | 10 | 14 | 20 | 24 | 30 | 34 |
| 5 | 5 | 12 | 17 | 24 | 31 | 36 | 43 |
| 6 | 6 | 14 | 22 | 30 | 36 | 44 | 52 |
| 7 | 7 | 16 | 25 | 34 | 43 | 52 | 61 |

найдем произведение

$273,03_8$

$\times 0,6_8$

$214,222_8$

д)  $214,222_8$

$- 26,72_8$

$\hline 165,302_8$

## Кодирование информации

Формирование представления информации называется ее кодированием. В более узком смысле под **кодированием** понимается переход от исходного представления информации, удобного для восприятия человеком, к представлению, удобному для хранения, передачи и обработки. Обратный переход к исходному представлению называется декодированием.

Для представления **информации** в ЭВМ (как числовой, так и нечисловой) используется двоичный цифровой код.

### Представление символьной (текстовой) информации в ЭВМ

В отличие от обычной словесной формы, принятой в письменном виде, символьная информация **хранится** и обрабатывается в памяти ЭВМ в форме цифрового двоичного кода. При преобразовании символов (знаков) в цифровой код между множествами символов и кодов должно иметь место взаимнооднозначное соответствие, т. е. разным символам должны быть назначены разные цифровые коды, и наоборот. Кроме того, из соображений наглядности и легкости запоминания целесообразно множество символов компьютерного алфавита, упорядоченных по какому-либо признаку (например, лексикографическому), кодировать также с помощью упорядоченной последовательности чисел. Мощность компьютерного алфавита – 256 символов. Таблица, в которой всем символам компьютерного алфавита поставлены в соответствие порядковые номера, называется **таблицей кодировки**. Для того, чтобы представить все номера двоичным цифровым кодом требуется 8 бит (1 байт).

В настоящее время **существует** несколько широко распространенных схем кодирования (таблиц кодировки): Windows CP-1251...

Международным **стандартом** является код ASCII – американский стандартный код информационного обмена (порядковые номера символов от 0 до 127, для чего требуется 7 бит). Остальные 128 кодов, используются в разных вариантах (добавляются символы национальных алфавитов: русского, немецкого, французского и др.)

Кодировка Unicode – мощность алфавита – 65536 символов, и для представления двоичных кодов требуется 16 бит.

### Числовая система ЭВМ

#### Представление целых чисел без знака

В некоторых случаях при представлении в памяти ЭВМ чисел для хранения каждого десятичного знака используется полубайт (4 бита) и десятичные цифры от 0 до 9 представляются соответствующими двоичными числами от 0000 до 1001 (в Паскале тип данных COMP – 10 байтов (18 значащих цифр + 1 разряд знаковый)).

Целые числа без знака могут занимать в памяти 1 байт (в Паскале тип данных BYTE) или 2 байта (в Паскале тип данных WORD) (количество двоичных разрядов определяет множество значений, которые могут быть записаны (или тип (формат) данных)).

Пусть  $I$  – количество бит, которые компьютер предоставляет для записи целого неотрицательного числа. Сколько разных чисел можно записать в  $I$  бит? Минимальное число? Максимальное число?

Формула Хартли  $I = \log_2 N$ , где  $N$ , в данном случае, количество разных чисел.

Следовательно,  $N = 2^I$ .

Минимальное число - 0 (рассматриваются числа неотрицательные)

Максимальное число  $2^I - 1$ .

Пусть  $I = 8$  бит, тогда  $N = 2^8 = 256$ , и в данном количестве бит можно представить числа из отрезка  $[0; 2^I - 1]$ , т. е.  $[0; 255]$ .

Как записать некоторое неотрицательное десятичное число  $M \in [0; 2^I - 1]$  в заданном количестве бит  $I$ .

Способ представления целых неотрицательных чисел – **прямой код**, который может быть получен по следующему алгоритму:

- 1) число  $M$  переводится в двоичную систему счисления:

- 2) двоичную запись числа М слева дополняют таким количеством незначащих нулей, сколько требует тип данных, к которому принадлежит число.

Пример. *Записать в 2 байтах неотрицательное десятичное число 1330.*

Решение.

Можно ли 1330 представить в 2 байт(ах) = 16 бит(ах)?

Максимальное число –  $2^{16} - 1 = 65535$ ,  $1330 < 65535$ .

- 1) число 1330 перевести в двоичную систему счисления

$$2^{10} = 1024 \text{ ост. } 1330 - 1024 = 306$$

$$2^8 = 256 \text{ ост. } 306 - 256 = 50$$

$$2^5 = 32 \text{ ост. } 50 - 32 = 18$$

$$2^4 = 16 \text{ ост. } 18 - 16 = 2$$

$$2^1 = 2 \text{ ост. } 2 - 2 = 0$$

$$1330 = 10100110010_2$$

- 2) двоичную запись числа 1330 слева дополнить таким количеством незначащих нулей, сколько требует тип данных, к которому принадлежит число

| номера битов |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
|--------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 15           | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0            | 0  | 0  | 0  | 0  | 1  | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 |

### Представление целых чисел со знаком

Диапазон значений зависит от количества битов памяти, отведенных для их хранения.

Пусть I – количество бит, которые компьютер предоставляет для записи целого числа со знаком. Старший разряд заданного количества бит называется знаковым: если в этом разряде 1 – число отрицательное, если 0 – неотрицательное.

Сколько разных чисел можно записать в I бит?  $N = 2^I$ .

Весь диапазон значений делится пополам: половина значений – отрицательные числа, другая половина - неотрицательные, т. е. в данном количестве бит можно представить числа из отрезка  $[-2^{I-1}; 2^{I-1} - 1]$ .

Неотрицательные числа записываются в прямом коде (см. выше).

Отрицательные числа записываются в дополнительном коде.

Дополнительный код целого отрицательного числа может быть получен по следующему алгоритму:

- 1) записать прямой код модуля числа;
- 2) инвертировать его (заменить единицы нулями, нули – единицами). Получен обратный (инверсный) код;
- 3) прибавить к инверсному коду единицу.

При получении числа по его дополнительному коду, прежде всего, необходимо определить знак числа (по значению старшего разряда заданного количества бит). Если число окажется неотрицательным, то надо просто перевести его в десятичную систему счисления. В случае отрицательного числа необходимо выполнить следующий алгоритм:

- 1) вычесть из кода число 1;
- 2) инвертировать код;
- 3) перевести двоичный код в десятичную систему счисления. Полученное число записать со знаком минус.

Пример. Записать в 2 байтах положительное десятичное число 555 и отрицательное десятичное число –1000.

Решение.

Можно ли 1330 представить в 2 байт = 16 бит(ах)?

Максимальное число  $2^{16-1} - 1 = 32767$ ,  $555 < 32767$ .



Минимальное число -  $-2^{16-1} = -32768, -1000 > -32768$ .

Число 555 представить в прямом коде.

1) Число 555 перевести в двоичную систему счисления

$$2^9 = 512 \text{ ост. } 555 - 512 = 43$$

$$2^5 = 32 \text{ ост. } 43 - 32 = 11$$

$$2^3 = 8 \text{ ост. } 11 - 8 = 3$$

$$2^1 = 2 \text{ ост. } 3 - 2 = 1$$

$$2^0 = 1 \text{ ост. } 1 - 1 = 0$$

$$555 = 1000101011_2$$

2) двоичную запись числа 555 слева дополнить таким количеством незначащих нулей, сколько требует тип данных, к которому принадлежит число

| номера битов |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
|--------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 15           | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0            | 0  | 0  | 0  | 0  | 0  | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |

Число -1000 представить в дополнительном коде.

1) Число 1000 представить в прямом коде.

а) Число 1000 перевести в двоичную систему счисления

$$2^9 = 512 \text{ ост. } 1000 - 512 = 488$$

$$2^8 = 256 \text{ ост. } 488 - 256 = 232$$

$$2^7 = 128 \text{ ост. } 232 - 128 = 104$$

$$2^6 = 64 \text{ ост. } 104 - 64 = 40$$

$$2^5 = 32 \text{ ост. } 40 - 32 = 8$$

$$2^3 = 8 \text{ ост. } 8 - 8 = 0$$

$$1000 = 1111101000_2$$

б) двоичную запись числа 1000 слева дополнить таким количеством незначащих нулей, сколько требует тип данных, к которому принадлежит число

| номера битов |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
|--------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 15           | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0            | 0  | 0  | 0  | 0  | 0  | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |

2) инвертировать прямой код (заменить единицы нулями, нули – единицами). Получен обратный (инверсный) код

| номера битов |    |    |    |    |    |   |   |   |   |   |   |   |   |   |   |
|--------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 15           | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 1            | 1  | 1  | 1  | 1  | 1  | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 |

3) прибавить к инверсному коду единицу

| номера битов |    |    |    |    |    |   |   |   |   |   |   |   |   |   |          |
|--------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|----------|
| 15           | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0        |
| 1            | 1  | 1  | 1  | 1  | 1  | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1        |
| +            |    |    |    |    |    |   |   |   |   |   |   |   |   |   | <b>1</b> |
| 1            | 1  | 1  | 1  | 1  | 1  | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0        |

## Элементы математической логики

В школьном курсе информатики изучают элементы алгебры логики. Определяются следующие понятия: логическое высказывание, логическая операция, логическая формула.

### Логическое высказывание

**Логическое высказывание** – это любое повествовательное предложение, в отношении которого можно однозначно сказать, истинно оно или ложно.

Выделим существенные признаки логического высказывания:

1. Повествовательное предложение;
2. По содержанию предложения однозначно можно определить его логическое значение («истина» или «ложь»).

#### Пример 1.

Пользуясь определением логического высказывания, установите, какие из следующих предложений являются логическими высказываниями:

- |   |  |
|---|--|
| <input type="checkbox"/> Город Париж — столица Австрии. | <input type="checkbox"/> Посмотрите налево, потом направо. |
| <input type="checkbox"/> $2 + 6 < 10$ .                 | <input type="checkbox"/> Число $X$ не превосходит десяти.  |

Запишем ход рассуждения в виде таблицы.

| Предложение                       | Переформулировка в полное предложение (если надо) | Признаки логического высказывания |  |  | Вывод                       |
|-----------------------------------|---|-----------------------------------|--|--|-----------------------------|
|                                   |   | Предложение повествовательное     | Логическое значение утверждения в предложении        | Наличие обоих признаков логического высказывания |                             |
| Город Париж — столица Австрии     | Город Париж является столицей Австрии             | Да                                | ложь   | +  | Логическое высказывание     |
| $2 + 6 < 10$                      | Сумма двух и шести меньше десяти.                 | Да                                | истина   | +  | Логическое высказывание     |
| Посмотрите налево, потом направо. |   | Нет (предложение побудительное)   | Не определяется                                      | -  | Не подходит под определение |
| Число $X$ не превосходит десяти   |   | Да                                | Не определяется, т.к. не известно значение числа $X$ | -  | Не подходит под определение |

В листе ответов решение будет выглядеть так:

- |  |  |
|--|--|
| <input checked="" type="checkbox"/> Город Париж — столица Австрии. | <input type="checkbox"/> Посмотрите налево, потом направо. |
| <input checked="" type="checkbox"/> $2 + 6 < 10$ .                 | <input type="checkbox"/> Число $X$ не превосходит десяти.  |

#### Пример 2.

Установите, какие из следующих предложений являются логическими высказываниями:

- |  |  |
|--|--|
| <input type="checkbox"/> Вы катались на лыжах? | <input type="checkbox"/> Прямые $X$ и $Y$ параллельны                    |
| <input type="checkbox"/> Число 8 делится на 5. | <input type="checkbox"/> При изменении мест слагаемых сумма не меняется. |

Запишем ход рассуждения в виде таблицы.

| Предложение                                    | Переформулировка в полное предложение (если надо) | Признаки логического высказывания |   |  | Вывод                       |
|--|---|-----------------------------------|---|--|-----------------------------|
|  |   | Предложение повествовательное     | Логическое значение утверждения в предложении                   | Наличие обоих признаков логического высказывания |                             |
| Вы катались на лыжах?                          |   | Нет (предложение вопросительное)  | Не определяется   | -  | Не подходит под определение |
| Число 8 делится на 5                           |   | Да                                | ложь  | +  | Логическое высказывание     |
| Прямые $X$ и $Y$ параллельны                   |   | да                                | Не определяется, т.к. не известно расположение прямых $X$ и $Y$ | -  | Не подходит под определение |
| При изменении мест слагаемых сумма не меняется |   | Да                                | истина  | +  | Логическое высказывание     |

В листе ответов решение будет выглядеть так:

Вы катались на лыжах?

Прямые X и Y параллельны

Число 8 делится на 5.

При изменении мест слагаемых сумма не меняется.

## Логические операции (связки)

Над одним или над несколькими логическими высказываниями можно совершать **операции**. Логическое значение результата операции зависит от значений высказываний, над которыми она совершалась. Значение логических операций удобно запоминать с помощью таблиц истинности или через словесное правило.

Логические высказывания обозначают латинскими буквами. Например, A= «Число 8 делится на 5», z= «Сегодня хорошая погода».

Чаще всего, в качестве значения «истина» используют обозначение 1, в качестве значения «ложь» используют обозначение 0.

| Логическое значение | Числовое обозначение |
|---------------------|----------------------|
| Истина              | 1                    |
| Ложь                | 0                    |

## Отрицание (обозначения «не», not, $\bar{\phantom{A}}$ )

Отрицание высказывания истинно только тогда, когда высказывание ложно, и ложно, когда высказывание истинно.

| A | Не A |
|---|------|
| 1 | 0    |
| 0 | 1    |

Высказывания A и (не A) называются противоположными.

### Пример 3.

Построить противоположное высказывание к A= « $3+5<12$ » и определить его логическое значение.

Из курса математики средней школы известно следующее правило построения отрицаний, которое можно записать в виде таблицы:

| № Строки | Знак отношения между числовыми выражениями | Знак отношения для противоположного высказывания | Пример логического высказывания и его значение A | Отрицание логического высказывания и его значение Не A |
|----------|--|--|--|--|
| 1        | $>$<br>«больше»                            | $\leq$<br>«не больше»                            | $1+2>0$<br>Истина<br>$5-4>9$<br>Ложь             | $1+2\leq 0$<br>Ложь<br>$5-4\leq 9$<br>Истина           |
| 2        | $<$<br>«меньше»                            | $\geq$<br>«не меньше»                            | $1-6<5+1$<br>Истина<br>$2*10<15$<br>Ложь         | $1-6\geq 5+1$<br>Ложь<br>$2*10\geq 15$<br>Истина       |
| 3        | $\geq$<br>«не меньше»                      | $<$<br>«меньше»                                  | $2+3\geq 5$<br>Истина<br>$7\geq 4+5,1$<br>Ложь   | $2+3< 5$<br>Ложь<br>$7< 4+5,1$<br>Истина               |

|   |                       |                      |   |   |
|---|-----------------------|----------------------|---|---|
| 4 | $\leq$<br>«не больше» | $>$<br>«больше»      | $\sin(30^\circ) \leq 1$<br>Истина<br>$2^3 \leq 2^2$<br>Ложь | $\sin(30^\circ) > 1$<br>Ложь<br>$2^3 > 2^2$<br>Истина |
| 5 | $=$<br>«равно»        | $\neq$<br>«не равно» | $\lg 10 = 1$<br>Истина<br>$1 + 1 = 3$<br>Ложь               | $\lg 10 \neq 1$<br>Ложь<br>$1 + 1 \neq 3$<br>Истина   |
| 6 | $\neq$<br>«не равно»  | $=$<br>«равно»       | $\cos(60^\circ) \neq 1$<br>Истина<br>$7^2 \neq 49$<br>Ложь  | $\cos(60^\circ) = 1$<br>Ложь<br>$7^2 = 49$<br>Истина  |

Решение задачи из примера 3: высказывание  $A = \langle 3+5 < 12 \rangle$  содержит отношение «меньше» (строка 2 таблицы выше). При построении отрицания знак меньше меняется на « $\geq$ ».

|                    |                     |
|--------------------|---------------------|
| A                  | Не A                |
| $3+5 < 12$         | $3+5 \geq 12$       |
| $8 < 12$<br>истина | $8 \geq 12$<br>ложь |

Ответ: не A =  $\langle 3+5 \geq 12 \rangle$ . Значение высказывания - ложь.

При формулировании отрицаний к логическим высказываниям нематематического содержания, как правило кроме слова «нет» вставляют сочетания «неверно, что», «никогда не выполняется», «не бывает так, что» и т.п., но часто используют и антонимы (тепло - холодно, ясно - пасмурно, глянцевый - матовый)

Например:

| Высказывание           | Отрицание высказывания |
|------------------------|------------------------|
| A = «Идет дождь»       | Не A = «дождя нет»     |
| B = «Пасмурная погода» | Не B = «Погода ясная»  |
| C = «Прибор исправен»  | Не C = «Прибор сломан» |

Для алгебры логики не важен бытовой или научный смысл сложных высказываний, получаемых из простых с помощью логических связей, рассматриваемых ниже.

### Конъюнкция (логическое умножение) (обозначения «и», and, $\wedge$ , &)

Конъюнкция двух высказываний **истинна тогда и только тогда, когда оба высказывания истинны**. Таблицу истинности для конъюнкции легко получить «перемножая» логические значения высказываний A и B.

| A | B | $A \wedge B$ |
|---|---|--------------|
| 1 | 1 | 1            |
| 1 | 0 | 0            |
| 0 | 1 | 0            |
| 0 | 0 | 0            |

#### Пример 4.

Построить конъюнкцию двух высказываний и определить ее логическое значение.

Дано: A =  $\langle 2+3 < 10 \rangle$

B =  $\langle 2 * 3 = 6 \rangle$

Построить: а)  $A \wedge B$

б)  $\text{not}(A) \wedge B$

Замечание: так как в данном задании операция отрицания применяется только к одному высказыванию, то она выполняется раньше конъюнкции.

Решение:

а)  $A \wedge B = \langle 2+3 < 10 \text{ и } 2*3=6 \rangle$

Логическое значение высказывания А-истина, т.к.  $5 < 10$ .

Логическое значение высказывания В-истина, т.к.  $6=6$ .

Построим таблицу истинности для сложного высказывания  $A \wedge B$ .

| A | B | $A \wedge B$ |
|---|---|--------------|
| 1 | 1 | 1            |

Ответ:  $A \wedge B = \langle 2+3 < 10 \text{ и } 2*3=6 \rangle$ . Высказывание истинно.

б)  $\text{not}(A) = \langle 2+3 \geq 10 \rangle$  (смотри пример 3).

$\text{not}(A) \wedge B = \langle 2+3 \geq 10 \text{ и } 2*3=6 \rangle$ .

Логическое значение высказывания А - истина, т.к.  $5 < 10$ .

Логическое значение высказывания  $\text{not}(A)$  - ложь, т.к. А - истина.

Логическое значение высказывания В - истина, т.к.  $6=6$ .

Построим таблицу истинности для сложного высказывания  $\text{not}(A) \wedge B$ .

| A | $\text{not}(A)$ | B | $\text{not}(A) \wedge B$ |
|---|-----------------|---|--------------------------|
| 1 | 0               | 1 | 0                        |

Ответ:  $A \wedge B = \langle 2+3 \geq 10 \text{ и } 2*3=6 \rangle$ . Высказывание ложно.

### Дизъюнкция (логическое сложение) (обозначения «или», $\vee$ )

Дизъюнкция двух высказываний **ложна тогда и только тогда, когда оба высказывания ложны**. Таблицу истинности для дизъюнкции легко получить «складывая» логические значения высказываний А и В.

| A | B | $A \vee B$ | замечание   |
|---|---|------------|---|
| 1 | 1 | 1          | $1_2 + 1_2 = 10_2$<br>В таблице оставляют только старший разряд числа |
| 1 | 0 | 1          |   |
| 0 | 1 | 1          |   |
| 0 | 0 | 0          |   |

### Пример 5.

Переведите на язык алгебры логики следующее сложное высказывание  $C = \langle \text{В учениях принимают участие пехотинцы или артиллеристы} \rangle$ . Сформулируйте простые логические высказывания, входящие в состав сложного. Найдите значения высказывания С, если известно, что в учениях принимали участие только летчики.

Решение:

Переформулируем высказывание С, не меняя его смысла: «В учениях принимают участие пехотинцы **или** в учениях принимают участие артиллеристы». Можно записать, что

$$C = A \vee B, \text{ где}$$

A = «В учениях принимают участие пехотинцы».

B = «В учениях принимают участие артиллеристы».

Так как в по условию в учениях принимали участие только летчики, то логические значения высказываний А и В – ложь.

Построим таблицу истинности для высказывания С.

| A | B | $C = A \vee B$ |
|---|---|----------------|
| 0 | 0 | 0              |

Ответ: A = «В учениях принимают участие пехотинцы».

B = «В учениях принимают участие артиллеристы».

Высказывание  $C = A \vee B$  имеет значение ложь.

## Импликация (логическое следование) (обозначения $\rightarrow, \Rightarrow$ )

### ПОСЫЛКА $\rightarrow$ ЗАКЛЮЧЕНИЕ

Для запоминания правила заполнения таблицы истинности для импликации имеет смысл выучить следующее правило: **из истины следует только истина, из лжи – все что угодно**. Эту фразу следует понимать так – если посылка истинна, то для истинности импликации заключение тоже должно быть истинно; если же посылка ложна, то импликация истинна независимо от значения заключения.

Таблица истинности для импликации.

| A | B | $A \rightarrow B$ | замечание                 |
|---|---|-------------------|---------------------------|
| 1 | 1 | 1                 | Из истины - только истина |
| 1 | 0 | 0                 |                           |
| 0 | 1 | 1                 | Из лжи - все что угодно   |
| 0 | 0 | 1                 |                           |

### Пример 6.

Даны следующие высказывания  $A = \langle 20 \text{ делится на } 2 \rangle$ ,  
 $B = \langle 70 \text{ делится на } 7 \rangle$ ,  
 $C = \langle 30 \text{ делится на } 4 \rangle$ .

Сформулируйте следующие сложные высказывания и определите их истинность:  $A \rightarrow C$ ,  
 $C \rightarrow B$ ,  
 $A \rightarrow B$ .

Решение:

$A \rightarrow C = \langle \text{Если } 20 \text{ делится на } 2, \text{ то } 30 \text{ делится на } 4 \rangle$

| A                         | C | $A \rightarrow C$ |
|---------------------------|---|-------------------|
| 1                         | 0 | 0                 |
| Из истины – только истина |   |                   |

$C \rightarrow B = \langle \text{Из того, что } 30 \text{ делится на } 4, \text{ следует, что } 70 \text{ делится на } 7 \rangle$

| C                       | B | $C \rightarrow B$ |
|-------------------------|---|-------------------|
| 0                       | 1 | 1                 |
| Из лжи – все что угодно |   |                   |

$A \rightarrow B = \langle \text{То, что } 20 \text{ делится на } 2, \text{ влечет за собой, что } 70 \text{ делится на } 7 \rangle$

| A                         | B | $A \rightarrow B$ |
|---------------------------|---|-------------------|
| 1                         | 1 | 1                 |
| Из истины – только истина |   |                   |

## Эквиваленция (обозначения $\leftrightarrow, \Leftrightarrow$ )

Эквиваленция двух логических высказываний **истинна тогда и только тогда, когда значения высказываний одинаковые**.

| A | B | $A \leftrightarrow B$ |
|---|---|-----------------------|
| 1 | 1 | 1                     |
| 1 | 0 | 0                     |
| 0 | 1 | 0                     |
| 0 | 0 | 1                     |

### Пример 7.

Даны следующие высказывания  $A = \langle 3 < 5 \rangle$ ,  
 $B = \langle 5 > 7 \rangle$ ,  
 $C = \langle 7 < 3 \rangle$ .

Сформулируйте следующие сложные высказывания и определите их истинность:  $A \leftrightarrow C$ ,  
 $C \leftrightarrow B$ ,  
 $A \leftrightarrow \text{not } B$ .

Решение:

$A \leftrightarrow C = \langle\langle 3 < 5 \text{ тогда и только тогда, когда } 7 < 3 \rangle\rangle$

| A | C | $A \leftrightarrow C$ |
|---|---|-----------------------|
| 1 | 0 | 0                     |

$C \leftrightarrow B = \langle\langle 7 < 3 \text{ равносильно } 5 > 7 \rangle\rangle$

| C | B | $C \leftrightarrow B$ |
|---|---|-----------------------|
| 0 | 0 | 1                     |

$\text{not } B = \langle\langle 5 \leq 7 \rangle\rangle$

$A \leftrightarrow \text{not } B = \langle\langle \text{Для того, чтобы было } 3 < 5 \text{ необходимо и достаточно, чтобы было } 5 \leq 7 \rangle\rangle$

| A | B | not B | $A \leftrightarrow \text{not } B$ |
|---|---|-------|-----------------------------------|
| 1 | 0 | 1     | 1                                 |

## Логические формулы

Определение **логической формулы** дается рекурсивно:

1. Любая логическая переменная или константа («истина» или «ложь») является формулой.
2. Если A и B – формулы, то  $(\text{not } A)$ ,  $(A \wedge B)$ ,  $(A \vee B)$ ,  $(A \rightarrow B)$ ,  $(A \leftrightarrow B)$  – формулы.
3. Других формул нет.

Примеры записи формул:  $F = (a \leftrightarrow \bar{b}) \wedge (b \Rightarrow a)$ ;  $F = a \wedge \bar{b} \Rightarrow (b \leftrightarrow a)$ ;  $F = a \wedge \bar{b} \vee (b \Rightarrow \bar{a})$ .

**Тавтологично – истинными** называются формулы, которые имеют значения «истина» при любых значениях в них входящих переменных.

Например,  $F = A \vee (\text{not } A)$  – тавтологично-истинная формула

| A | not A | $A \vee (\text{not } A)$ |
|---|-------|--------------------------|
| 1 | 0     | 1                        |
| 0 | 1     | 1                        |

**Тавтологично – ложными** называются формулы, которые имеют значения «ложь» при любых значениях в них входящих переменных.

Например,  $F = A \wedge (\text{not } A)$  – тавтологично-ложная формула

| A | not A | $A \wedge (\text{not } A)$ |
|---|-------|----------------------------|
| 1 | 0     | 0                          |
| 0 | 1     | 0                          |

Все прочие формулы называются **выполнимыми**.

Формулы называются **равносильными**, если они принимают одинаковые значения при одинаковых значениях входящих в них переменных.

### Пример 8.

Определите, какая из таблиц истинности соответствует формуле  $F = (a \leftrightarrow \bar{b}) \wedge (b \Rightarrow a)$

| <input type="checkbox"/> | <table border="1" style="width: 100%; text-align: center;"> <thead> <tr><th>A</th><th>b</th><th>F</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table> | A | b | F | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | <input type="checkbox"/> | <table border="1" style="width: 100%; text-align: center;"> <thead> <tr><th>a</th><th>b</th><th>F</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table> | a | b | F | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | <input type="checkbox"/> | <table border="1" style="width: 100%; text-align: center;"> <thead> <tr><th>a</th><th>b</th><th>F</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table> | a | b | F | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | <input type="checkbox"/> | <table border="1" style="width: 100%; text-align: center;"> <thead> <tr><th>a</th><th>b</th><th>F</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table> | a | b | F | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |
|--------------------------|--|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--------------------------|--|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--------------------------|--|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--------------------------|--|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A                        | b  | F |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0                        | 0  | 0 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0                        | 1  | 0 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1                        | 0  | 0 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1                        | 1  | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| a                        | b  | F |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0                        | 0  | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0                        | 1  | 0 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1                        | 0  | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1                        | 1  | 0 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| a                        | b  | F |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0                        | 0  | 0 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0                        | 1  | 0 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1                        | 0  | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1                        | 1  | 0 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| a                        | b  | F |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0                        | 0  | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0                        | 1  | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1                        | 0  | 0 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1                        | 1  | 0 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |                          |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |

Решение:

Построим таблицу истинности для формулы  $F = (a \leftrightarrow \bar{b}) \wedge (b \Rightarrow a)$ . Для этого определим какие логические операции использованы при ее построении:

1. not b
2.  $a \leftrightarrow \text{not } b$
3.  $b \rightarrow a$

4.  $F = (a \leftrightarrow \text{not } b) \wedge (b \leftrightarrow a)$

| a | b | not b | $a \leftrightarrow \text{not } b$ | $b \rightarrow a$ | F |
|---|---|-------|-----------------------------------|-------------------|---|
| 0 | 0 | 1     | 0                                 | 1                 | 0 |
| 0 | 1 | 0     | 1                                 | 0                 | 0 |
| 1 | 0 | 1     | 1                                 | 1                 | 1 |
| 1 | 1 | 0     | 0                                 | 1                 | 0 |

Ответ:

| <input type="checkbox"/> <table border="1" style="width: 100%; text-align: center;"> <thead> <tr><th>A</th><th>b</th><th>F</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table> | A | b | F | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | <input type="checkbox"/> <table border="1" style="width: 100%; text-align: center;"> <thead> <tr><th>a</th><th>b</th><th>F</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table> | a | b | F | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | <input checked="" type="checkbox"/> <table border="1" style="width: 100%; text-align: center;"> <thead> <tr><th>a</th><th>b</th><th>F</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table> | a | b | F | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | <input type="checkbox"/> <table border="1" style="width: 100%; text-align: center;"> <thead> <tr><th>a</th><th>b</th><th>F</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table> | a | b | F | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A   | b | F |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 0 | 0 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 1 | 0 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 0 | 0 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 1 | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| a   | b | F |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 0 | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 1 | 0 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 0 | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 1 | 0 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| a   | b | F |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 0 | 0 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 1 | 0 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 0 | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 1 | 0 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| a   | b | F |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 0 | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 1 | 1 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 0 | 0 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 1 | 0 |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |

**Пример 9.**

Истинность двух высказываний: «неверно, что если полк В участвует в учениях, то полк С участвует в учениях» и «если полк А участвует в учениях, то полк В не участвует» означает учения полков

С       А       В       А,С

Решение:

1. Сформулируем простейшие высказывания.

A = «в учениях принимает участие полк А»

B = «в учениях принимает участие полк В»

C = «в учениях принимает участие полк С»

2. Логическая связка «полк В участвует в учениях, то полк С участвует в учениях» запишется так  $B \rightarrow C$ .

Её отрицание - «неверно, что если полк В участвует в учениях, то полк С участвует в учениях» будет  $\text{not } (B \rightarrow C)$ . По условию задачи значение формулы есть 1.

Заполним таблицу истинности для полученной формулы.

| B        | C        | $B \rightarrow C$ | $\text{not } (B \rightarrow C)$ |
|----------|----------|-------------------|---------------------------------|
| 0        | 0        | 1                 | 0                               |
| 0        | 1        | 1                 | 0                               |
| <b>1</b> | <b>0</b> | <b>0</b>          | <b>1</b>                        |
| 1        | 1        | 1                 | 0                               |

Вывод: формула истинна только если истинно высказывание В, значит полк **В участвует в учениях**.

3. Проанализируем второе сложное высказывание из условия задачи. Отрицание высказывания В есть «полк В не участвует», то есть  $(\text{not } B)$ .

Логическая связка «если полк А участвует в учениях, то полк В не участвует» запишется формулой  $A \rightarrow \text{not } B$ . По условию задачи значение формулы есть 1.

Заполним таблицу истинности для полученной формулы, учитывая вывод, что высказывание В принимает значение 1.

| A        | B        | not B    | $A \rightarrow \text{not } B$ |
|----------|----------|----------|-------------------------------|
| <b>0</b> | <b>1</b> | <b>0</b> | <b>1</b>                      |
| <b>1</b> | <b>1</b> | <b>0</b> | <b>1</b>                      |

Вывод: формула  $A \rightarrow \text{not } B$  истинна при истинном заключении В.

Значит, из истинности двух данных в условии задачи высказываний следует, что в учениях **обязательно принимает участие полк В**.

Ответ:  С       А       В       А,С



### Пример 10.

Истинность двух высказываний: «школьник В пойдет в кино, а школьник С не пойдет» и «из двух школьников А и С пойдет в кино только один» означает посещение кинотеатра школьниками

А, В, С     В     А, В     В, С

Решение:

1. Сформулируем простейшие высказывания.

| Высказывание                   | Отрицание высказывания                |
|--------------------------------|---------------------------------------|
| А = «школьник А пойдет в кино» | not А = «школьник А не пойдет в кино» |
| В = «школьник В пойдет в кино» | not В = «школьник В не пойдет в кино» |
| С = «школьник С пойдет в кино» | not С = «школьник С не пойдет в кино» |

2. Запишем логическую формулу для первого высказывания «школьник В пойдет в кино, а школьник С не пойдет»  $V \wedge \text{not } C$ . По условию задачи его значение истина (1).

Составим таблицу истинности и определим, при каких значения В и С оно истинно.

| В        | С        | not C    | $V \wedge \text{not } C$ |
|----------|----------|----------|--------------------------|
| 0        | 0        | 1        | 0                        |
| 0        | 1        | 0        | 0                        |
| <b>1</b> | <b>0</b> | <b>1</b> | <b>1</b>                 |
| 1        | 1        | 0        | 0                        |

Интерпретируем результат третьей строки, так как только в ней значение формулы есть 1: истинно высказывание В = «школьник В пойдет в кино» и ложно высказывание С = «школьник С пойдет в кино». Значит, **школьник С не пойдет в кино**.

3. Проанализируем второе сложное высказывание из условия задачи «из двух школьников А и С пойдет в кино только один». Переформулируем его: «в кино пойдет школьник А и не пойдет школьник С, или в кино пойдет школьник С и не пойдет школьник А». Запишем логическую формулу  $(A \wedge \text{not } C) \vee (C \wedge \text{not } A)$ .

Построим таблицу истинности, зная что высказывание С имеет значение ложь (0).

| А        | С        | not A    | not C    | $A \wedge \text{not } C$ | $C \wedge \text{not } A$ | $(A \wedge \text{not } C) \vee (C \wedge \text{not } A)$ |
|----------|----------|----------|----------|--------------------------|--------------------------|--|
| 0        | 0        | 1        | 1        | 0                        | 0                        | 0  |
| <b>1</b> | <b>0</b> | <b>0</b> | <b>1</b> | <b>1</b>                 | <b>0</b>                 | <b>1</b>   |

Из последней строки таблицы видно, что высказывание А истинно, значит **школьник А пойдет в кино**.

Соединим выводы, полученные выше в пунктах 2 и 3 и получим, что в **кино пойдут школьники А и В**.

Ответ:  А, В, С     В     А, В     В, С

### Пример 11.

Даны высказывания: А={Идет дождь}, В={Повышенная влажность}, С={Пасмурная погода}. Переведите на язык алгебры логики следующее высказывание: «Если будет пасмурно, то будет повышенная влажность, и если будет дождь, то будет пасмурно».

Решение.

1. Переведем на язык алгебры логики первую часть сложного высказывания «Если будет пасмурно, то будет повышенная влажность» ( $C \rightarrow B$ )
2. «Если будет дождь, то будет пасмурно» соответствует формуле ( $A \rightarrow C$ ).
3. Значит данное в условии утверждение запишется как  $(C \rightarrow B) \wedge (A \rightarrow C)$ .

Ответ:  $(C \rightarrow B) \wedge (A \rightarrow C)$ .

## Основные законы алгебры логики

Тождественные преобразования формул алгебры логики производятся на основе основных равенств (законов). Равносильность логических функций, стоящих слева и справа от равенства легко определяется по таблицам истинности.

| Название закона                           | Для конъюнкции   | Для дизъюнкции  |
|---|--|---|
| Отсутствия степеней и коэффициентов       | $A \wedge A = A$   | $A \vee A = A$  |
| Коммутативность<br>(переместительность)   | $A \wedge B = B \wedge A$  | $A \vee B = B \vee A$   |
| Ассоциативность<br>(сочетательность)      | $(A \wedge B) \wedge C = A \wedge (B \wedge C)$                                | $(A \vee B) \vee C = A \vee (B \vee C)$                       |
| Дистрибутивность<br>(Распределительность) | $(A \vee B) \wedge C = (A \wedge C) \vee (B \wedge C)$                         | $(A \wedge B) \vee C = (A \vee C) \wedge (B \vee C)$          |
| Правила Де Моргана                        | $\text{not } (A \wedge B) = \text{not } A \vee \text{not } B$                  | $\text{not } (A \vee B) = \text{not } A \wedge \text{not } B$ |
| Поглощения                                | $A \vee (A \wedge B) = A$  | $A \wedge (A \vee B) = A$                                     |
| Склеивания                                | $(A \wedge B) \vee (\text{not } A \wedge B) = B$                               | $(A \vee B) \wedge (\text{not } A \vee B) = B$                |
| Противоречия                              | $A \wedge \text{not } A = 0$   |   |
| Исключения третьего                       |  | $A \vee \text{not } A = 1$                                    |
| Свойства логических констант              | $A \wedge 1 = A$<br>$A \wedge 0 = 0$   | $A \vee 1 = 1$<br>$A \vee 0 = 0$                              |
| Двойного отрицания                        | $\text{not}(\text{not } A) = A$  |   |
| Отрицание логических констант             | $\text{not}(0) = 1$  | $\text{not}(1) = 0$   |
| Представления импликации                  | $A \rightarrow B = \text{not } A \vee B$                                       |   |
| Представления эквиваленции                | $A \leftrightarrow B = (A \wedge B) \vee (\text{not } A \wedge \text{not } B)$ |   |

### Построение логических формул

В задания уровня В теста по информатике включено построение логической формулы по заданной таблице истинности. Возможны различные варианты решения таких задач. Мы рассмотрим общий способ на примере задания 6 уровня В (ЯГПУ, 2005г. Вариант 4).

Переход от табличного задания формулы к аналитическому возможен всегда. Метод перехода заключается в следующем:

1. Выделяются строки таблицы, в которых значения функции равно 1;
2. Для каждой из таких строк выписывается конъюнкция переменных. Над теми переменными, значения которых в этой строке равны 0, ставятся отрицания.
3. Все полученные конъюнкции соединяются знаками дизъюнкции.

Обычно полученная формула очень громоздкая. Ее упрощают на основе эквивалентных преобразований, опирающихся на основные законы алгебры логики.

**Пример 12.**

Символом F обозначено логическое выражение от двух аргументов: X, Y. Для данной таблицы истинности постройте одно из возможных выражений F.

| X | Y | F |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Решение.

1. Составим конъюнкции переменных в строках, где F=1.

| X | Y | F | Конъюнкция                           |
|---|---|---|--------------------------------------|
| 0 | 0 | 1 | $\text{not } X \wedge \text{not } Y$ |
| 0 | 1 | 1 | $\text{not } X \wedge Y$             |
| 1 | 0 | 1 | $X \wedge \text{not } Y$             |
| 1 | 1 | 0 |                                      |

1. Соединим полученные конъюнкции знаками дизъюнкции:

$$F = (\text{not } X \wedge \text{not } Y) \vee (\text{not } X \wedge Y) \vee (X \wedge \text{not } Y)$$

Чтобы не переписывать длинную формулу, распишем преобразования по действиям.

- $(\text{not } X \wedge \text{not } Y) \vee (\text{not } X \wedge Y) = \text{not } X \wedge (\text{not } Y \vee Y)$  по распределительному закону.
- $\text{not } Y \vee Y = 1$  по закону исключения третьего.
- $\text{not } X \wedge 1 = \text{not } X$  по свойству логической константы.

Имеем  $F = \text{not } X \vee (X \wedge \text{not } Y)$ . Можно упростить еще.

- $\text{not } X \vee (X \wedge \text{not } Y) = (\text{not } X \vee X) \wedge (\text{not } X \vee \text{not } Y)$  по распределительному закону.
- $\text{not } X \vee X = 1$  по закону исключения третьего.
- $1 \wedge (\text{not } X \vee \text{not } Y) = \text{not } X \vee \text{not } Y$  по свойству логической константы.

Получили  $F = \text{not } X \vee \text{not } Y$ . Далее можно не упрощать, но можно и воспользоваться правилом де Моргана ( $F = \text{not } (X \wedge Y)$ ). Любой из этих ответов будет признан как правильный.

**Ответ:**  $F = \text{not } (X \wedge Y)$ .

**Пример 13.**

Упростите логическую функцию:  $f(a, b, c) = \bar{c} \Rightarrow ((a \vee \bar{b}) \wedge b \vee \bar{a} \wedge (b \vee c))$

Решение (по шагам).

- $(a \vee \text{not } b) \wedge b = (a \wedge b) \vee (\text{not } b \wedge b)$  по распределительному закону.
- $\text{not } b \wedge b = 0$  по закону противоречия.
- $(a \wedge b) \vee 0 = a \wedge b$  по свойству логической константы.
- $(a \wedge b) \vee \text{not } a = (a \vee \text{not } a) \wedge (b \vee \text{not } a)$  по распределительному закону.
- $a \vee \text{not } a = 1$  по закону исключения третьего.
- $1 \wedge (b \vee \text{not } a) = b \vee \text{not } a$  по свойству логической константы.
- $(b \vee \text{not } a) \wedge (b \vee c) = b \vee (\text{not } a \wedge c)$  по распределительному закону.
- $(\text{not } c) \rightarrow (b \vee (\text{not } a \wedge c)) = c \vee b \vee (\text{not } a \wedge c)$  по закону представления импликации.
- $c \vee b \vee (\text{not } a \wedge c) = b \vee c \vee (\text{not } a \wedge c)$  по переместительному закону.
- $c \vee (\text{not } a \wedge c) = c$  по закону поглощения.

11.  $F = b \vee c$ .

**Ответ:**  $F = b \vee c$ .

## Логические задачи

Применение логических законов наглядно при решении задач, требующих знания алгебры логики. В различных школьных учебниках по информатике рассматриваются разные подходы к их решению. На наш взгляд самый удобный метод – табличный, однако учащимся доступны и метод логических рассуждений и метод формального преобразования логических формул. Методы решения логических задач хорошо изложены в книге Л.З. Шауцковой.

### Этапы решения логической задачи:

1. Формализация условия:
  - 1.1 выделение простейших логических высказываний;
  - 1.2 определение логических операций над полученными высказываниями;
2. Конструирование логической формулы фабулы задачи;
3. Определение значений логической формулы;
4. Интерпретация результата: по истинным значениям формулы определяются логические значения

### Пример 14.

Решите логическую задачу:

Из Москвы в Санкт-Петербург едут Сидоров, Иванов и Петров. Фамилии у этих пассажиров настолько распространенные, что оказалось, что так же зовут трех человек из поездной бригады – кочегара, кондуктора и машиниста.

Известно, что

- 1) все пассажиры живут в разных местах по Октябрьской железной дороге;
- 2) все члены поездной бригады живут в одном городе;
- 3) пассажир Иванов живет в Москве;
- 4) кондуктор живет на полпути между Москвой и Санкт-Петербургом, в городе Бологое;
- 5) пассажир – однофамилец кондуктора живет в Санкт-Петербурге;
- 6) пассажир Петров не проживает там же, где его однофамилец;
- 7) Сидоров из поездной бригады выиграл у кочегара партию в бильярд.

Какая фамилия у каждого из членов поездной бригады и где живут они и пассажиры?

Решение.

1. Создадим таблицу, в которую будем заносить результаты логических рассуждений.
2. Проанализируем пункты 2 и 4 условия задачи. Из них следует, что вся бригада поезда живет в г.Бологое. Заполним нижнюю часть столбца «Бологое», относящуюся к бригаде поезда, знаком «+». Чтобы не закралась ошибка, нижнюю часть столбцов «Москва» и «С-Петербург» знаками «-».
3. Занесем в таблицу данные пункта 3 условия задачи.
4. Из условия в пункте 6 условия следует, что пассажир Петров не живет в г. Бологое, т.к. его однофамилец есть только в поездной бригаде, но он не живет и в Москве. Значит он из с-Петербурга.
5. Из пунктов 1 условия и пунктов 3 и 4 решения следует, что пассажир Сидоров живет в г.Бологое.
6. Так как по условию задачи (пункт 5) кондуктор однофамилец жителя С-Петербурга, то его фамилия Петров (см.таблицу).
7. Член бригады Сидоров не кочегар, т.к. он играл с ним в бильярд, но он и не кондуктор, значит он – машинист.
8. Из предыдущих рассуждений следует, что Иванов из бригады поезда может быть только кочегаром.

| Категория | Фамилия | Москва | Бологое | С-Петербург | Кочегар | Машинист | Кондуктор |
|-----------|---------|--------|---------|-------------|---------|----------|-----------|
| Пассажиры | Иванов  | +      |         |             |         |          |           |
|           | Петров  |        |         | +           |         |          |           |
|           | Сидоров |        | +       |             |         |          |           |
| Бригада   | Иванов  | -      | +       | -           | +       |          |           |
|           | Петров  | -      | +       | -           |         |          | +         |
|           | Сидоров | -      | +       | -           |         | +        |           |

Ответ: Все члены бригады поезда живут в г.Бологое. Иванов – кочегар;  
Петров – кондуктор;  
Сидоров – машинист.

Пассажиры: Иванов – из Москвы;  
Петров – из С-Петербурга;  
Сидоров – из г. Бологое.

## Рекуррентные соотношения

Рассмотрение понятия рекуррентное соотношение начнем с решения задачи.

*Задача.*

Сколькими способами можно замостить прямоугольную доску размером  $2 \times 7$  костями домино, если все кости считать одинаковыми и учитывать только положение кости: горизонтальное или вертикальное?

*Решение.*

Обозначим через  $\Phi(k)$  количество способов замостить костями домино прямоугольную доску размером  $2 \times k$ . Угловая клетка может быть закрыта одним из двух способов: либо костью, которая лежит вертикально, тогда оставшуюся  $k-1$  кость можно положить  $\Phi(k-1)$  способами, либо костью, которая лежит горизонтально, тогда еще одну кость можем положить только горизонтально, а оставшиеся  $k-2$  кости можно уложить  $\Phi(k-2)$  способами. Используя правило суммы, приходим к соотношению  $\Phi(k)=\Phi(k-1)+\Phi(k-2)$ . Учитывая, что  $\Phi(0)=\Phi(1)=1$ , можем для любого значения  $k$  найти ответ:  $\Phi(2)=2$ ,  $\Phi(3)=3$ ,  $\Phi(4)=5$ ,  $\Phi(5)=8$ ,  $\Phi(6)=13$ ,  $\Phi(7)=21$ . Имеем 21 способ замостить костями домино прямоугольную доску размером  $2 \times 7$ .

При решении многих комбинаторных (и не только) задач часто встречается способ, когда задачу с заданными значениями параметров сводят к аналогичной задаче, но уже с меньшими значениями параметров. Таким образом, можно довести задачу до простой. Данный метод решения задач носит название **метода рекуррентных соотношений**. (От латинского слова *recurrere* – возвращаться).

### Пример.

Используя метод рекуррентных соотношений, можно по-новому вывести формулу для вычисления числа перестановок из  $k$  предметов.

Обозначим за  $P(k)$  количество перестановок из элементов  $k$  типов. В перестановке на первом месте может быть любой из  $k$  предметов, а оставшиеся  $k-1$  предмет можно в каждом из этих случаев переставить  $P(k-1)$  способами. По правилу произведения получаем формулу  $P(k)=k \cdot P(k-1)$ . Далее замечаем, что  $P(1)=1$ , и получаем, что  $P(k)=k!$

## Числа Фибоначчи

Формула, которую мы получили при решении задачи о домино, впервые была опубликована в книге “*Liber Abaci*”, появившейся в 1202 году, где итальянский математик Фибоначчи среди многих других задач привел следующую:

Пара кроликов приносит раз в месяц приплод из двух крольчат (самки и самца), причем новорожденные крольчата через два месяца после рождения уже приносят приплод. Сколько кроликов появится через год, если в начале года была одна пара кроликов?

Из условия задачи следует, что через месяц будет две пары кроликов. Через два месяца приплод даст только первая пара кроликов, и получится 3 пары. А еще через месяц приплод дадут и исходная пара кроликов, и пара кроликов, появившаяся два месяца тому назад. Поэтому всего будет 5 пар кроликов.

Обозначим через  $F(n)$  количество пар кроликов по истечении  $n$  месяцев с начала года. Мы видим, что через  $n+1$  месяцев будут эти  $F(n)$  пар и еще столько новорожденных пар кроликов, сколько было в конце месяца  $n-1$ , то есть еще  $F(n-1)$  пар кроликов. Иными словами, имеет место рекуррентное соотношение

$$F(n+1)=F(n)+F(n-1).$$

Так как, по условию,  $F(0)=1$  и  $F(1)=2$ , то последовательно находим

$$F(2)=3, F(3)=5, F(4)=8 \text{ и т. д.}$$

Полученные числа называют числами Фибоначчи.

Чтобы найти  $F(12)$ , нам придется последовательно вычислить  $F(3), F(4), \dots, F(11)$ , что достаточно долго. А если нам необходимо было бы вычислить  $F(100)$ , то это займет еще больше времени. Попробуем выразить закономерность последовательности Фибоначчи с помощью расчетной формулы (вместо неявного рекуррентного соотношения). Для этого присвоим двоичный номер каждой паре кроликов. Единицам соответствуют месяцы появления на свет одной из пар “предков” данной пары (включая и исходную), а нулями – все остальные месяцы. Например, последовательность 010010100010 устанавливает такую “генеалогия” – сама пара появилась в конце 11-го месяца, ее родители – в конце 7-го месяца, “дед” – в конце 5-го месяца, “прадед” – в конце второго месяца. Исходная пара кроликов зашифровывается при этом последовательностью 000000000000. в последовательности не могут идти подряд две единицы, так как кролики приносят приплод только на второй месяц после рождения.

Тем самым устанавливается связь между числами Фибоначчи и следующей комбинаторной задачей: найти число  $n$ -последовательностей, состоящих из нулей и единиц, в которых никакие две единицы не идут подряд. Число таких последовательностей, в которые входит ровно  $k$  единиц и  $n-k$  нулей, равно  $C_{n-k+1}^k$ . Так как при этом должно выполняться неравенство  $k \leq n-k+1$ , то  $k$  изменяется от 0 до целой части числа  $(n+1)/2$ . Применяя правило суммы, получаем  $F(n) = C_{n+1}^0 + C_n^1 + C_{n-1}^2 + \dots + C_{n-p+1}^p$ , где  $p$  – целая часть числа  $(n+1)/2$ .

К сожалению, задачу нельзя считать решенной, так как, хотя получено выражение, зависящее от  $n$ , его вычисление оказывается даже сложнее рекуррентных расчетов. Желаемую формулу можно получить совсем другим способом.

### Решение рекуррентных соотношений

Рекуррентное соотношение имеет *порядок*  $k$ , если оно позволяет выразить  $f(n+k)$  через  $f(n), f(n+1), \dots, f(n+k-1)$ .

*Пример.*

$f(n+2) = f(n)f(n+1) - 3f^2(n+1) + 1$  – рекуррентное соотношение второго порядка.

$f(n+3) = 6f(n)f(n+2) + f(n+1)$  – рекуррентное соотношение третьего порядка.

Если задано рекуррентное соотношение  $k$ -го порядка, то ему могут удовлетворять бесконечно много последовательностей, так как первые  $k$  элементов последовательности можно задать произвольно – между ними нет никаких соотношений. Но если первые  $k$  членов заданы, то все остальные элементы определяются однозначно.

Пользуясь рекуррентным соотношением и начальными членами, можно один за другим выписывать члены последовательности, при этом рано или поздно мы получим любой её член. Однако если необходимо узнать только один определенный член последовательности, то нерационально вычислять все предыдущие. В этом случае удобнее иметь формулу для вычисления  $n$ -го члена.

*Решением рекуррентного соотношения* называется любая последовательность, для которой данное соотношение выполнено тождественно.

*Пример.*

Последовательность  $2, 4, 8, \dots, 2^n$  является решением для соотношения

$$f(n+2) = 3 \cdot f(n+1) - 2 \cdot f(n).$$

*Доказательство.*

Общий член последовательности имеет вид  $f(n) = 2^n$ . Значит,  $f(n+2) = 2^{n+2}$ ,  $f(n+1) = 2^{n+1}$ . При любом  $n$  имеет место тождество  $2^{n+2} = 3 \cdot 2^{n+1} - 2 \cdot 2^n$ . Следовательно, при подстановке последовательности  $2^n$  в формулу  $f(n+2) = 3f(n+1) - 2f(n)$  соотношение выполняется тождественно. Значит,  $2^n$  является решением указанного соотношения.

**Решение рекуррентного соотношения**  $k$ -го порядка называется **общим**, если оно зависит от  $k$  произвольных постоянных  $\alpha_1, \alpha_2, \dots, \alpha_k$  и путем подбора этих постоянных можно получить любое решение данного соотношения.

*Пример.*

Дано рекуррентное соотношение:  $f(n+2)=5f(n+1)-6f(n)$ . Докажем, что его общее решение имеет вид:  $f(n)=\alpha 2^n + \beta 3^n$ .

1. Сначала докажем, что последовательность  $f(n)=\alpha 2^n + \beta 3^n$  является решением рекуррентного соотношения. Подставим данную последовательность в рекуррентное соотношение.

$f(n)=\alpha 2^n + \beta 3^n$ , значит,  $f(n+1)=(\alpha 2^{n+1} + \beta 3^{n+1})$ ,  $f(n+2)=\alpha 2^{n+2} + \beta 3^{n+2}$ , тогда  $5f(n+1)-6f(n)=5(\alpha 2^{n+1} + \beta 3^{n+1})-6(\alpha 2^n + \beta 3^n)=\alpha (5 \cdot 2^{n+1} - 6 \cdot 2^n) + \beta (5 \cdot 3^{n+1} - 6 \cdot 3^n) = \alpha 2^n \cdot (10-6) + \beta 3^n \cdot (15-6) = \alpha 2^{n+2} + \beta 3^{n+2} = f(n+2)$ .

Рекуррентное соотношение выполняется, следовательно,  $\alpha 2^n + \beta 3^n$  является решением данного рекуррентного соотношения.

2. Докажем, что любое решение соотношения  $f(n+2)=5f(n+1)-6f(n)$  можно записать в виде  $f(n)=\alpha 2^n + \beta 3^n$ . Но любое решение рекуррентного соотношения второго порядка однозначно определяется значениями первых двух членов последовательности. Поэтому достаточно показать, что для любых  $a=f(1)$  и  $b=f(2)$  найдутся  $\alpha$  и  $\beta$  такие, что  $2\alpha + 3\beta = a$  и  $4\alpha + 9\beta = b$ . Легко видеть, что система уравнений имеет решение для любых значений  $a$  и  $b$ .

Таким образом,  $f(n)=\alpha 2^n + \beta 3^n$  является общим решением рекуррентного соотношения  $f(n+2)=5f(n+1)-6f(n)$ .

### **Линейные рекуррентные соотношения с постоянными коэффициентами**

Для решения рекуррентных соотношений общих правил нет, но существует часто встречающийся класс рекуррентных соотношений, для которых известен алгоритм их решения. Это – линейные рекуррентные соотношения с постоянными коэффициентами, т.е. соотношения вида:

$$f(n+k)=c_1f(n+k-1)+c_2f(n+k-2)+\dots+c_kf(n).$$

Найдем решение общего линейного рекуррентного соотношения с постоянными коэффициентами первого порядка.

Линейное рекуррентное соотношение с постоянными коэффициентами первого порядка имеет вид:  $f(n+1)=c f(n)$ .

Пусть  $f(1)=a$ , тогда  $f(2)=c \cdot f(1)=c \cdot a$ ,  $f(3)=c \cdot f(2)=c^2 \cdot a$ , аналогично  $f(4)=c^3 \cdot a$  и так далее, заметим, что  $f(n)=c^{n-1} \cdot f(1)$ .

Докажем, что последовательность  $c^{n-1} \cdot f(1)$  является решением рекуррентного соотношения первого порядка.  $f(n)=c^{n-1} \cdot f(1)$ , значит,  $f(n+1)=c^n f(1)$ . Подставляя это выражение в соотношение, получим тождество  $c^n f(1)=c \cdot c^{n-1} \cdot f(1)$ .

Рассмотрим теперь подробнее **линейные рекуррентные соотношения с постоянными коэффициентами второго порядка**, то есть соотношения вида

$$f(n+2)=C_1 \cdot f(n+1)+C_2 \cdot f(n). (*)$$

Заметим, что все рассуждения сохраняются и для соотношений более высокого порядка.

**Свойства решений:**

- 1) Если последовательность  $x_n$  является решением (\*), то и последовательность  $\alpha \cdot x_n$  тоже является решением.

*Доказательство.*



$x_n$  является решением (\*), следовательно, выполняется тождество  $x_{n+2}=C_1x_{n+1}+C_2x_n$ . Домножим обе части равенства на  $\alpha$ . Получим  $\alpha \cdot x_{n+2} = \alpha \cdot (C_1 \cdot x_{n+1} + C_2 \cdot x_n) = C_1 \cdot \alpha \cdot x_{n+1} + C_2 \cdot \alpha \cdot x_n$ . Это означает, что  $\alpha x_n$  является решением (\*).

- 2) Если последовательности  $x_n$  и  $y_n$  являются решениями (\*), то и последовательность  $x_n + y_n$  тоже является решением.

*Доказательство.*

$x_n$  и  $y_n$  являются решениями, следовательно, выполняются следующие тождества:

$$x_{n+2} = C_1 x_{n+1} + C_2 x_n.$$

$$y_{n+2} = C_1 y_{n+1} + C_2 y_n.$$

Выполним почленное сложение двух равенств:

$x_{n+2} + y_{n+2} = C_1 \cdot x_{n+1} + C_2 \cdot x_n + C_1 \cdot y_{n+1} + C_2 \cdot y_n = C_1 \cdot (x_{n+1} + y_{n+1}) + C_2 \cdot (x_n + y_n)$ . Это означает, что  $x_n + y_n$  является решением (\*).

- 3) Если  $r_1$  является решением квадратного уравнения  $r^2 = C_1 r + C_2$ , то последовательность  $(r_1)^n$  является решением для соотношения (\*).

$r_1$  является решением квадратного уравнения  $r^2 = C_1 r + C_2$ , значит,  $(r_1)^2 = C_1 r_1 + C_2$ . Помножим обе части равенства на  $(r_1)^n$ . Получим

$$r_1^2 \cdot r_1^n = (C_1 r_1 + C_2) r_1^n.$$

$$r_1^{n+2} = C_1 r_1^{n+1} + C_2 r_1^n.$$

Это означает, что последовательность  $(r_1)^n$  является решением (\*).

Из этих свойств вытекает **способ решения** линейных рекуррентных соотношений с постоянными коэффициентами второго порядка:

1. Составим характеристическое (квадратное) уравнение  $r^2 = C_1 r + C_2$ . Найдём его корни  $r_1, r_2$ . Если корни различны, то общее решение имеет вид  $f(n) = \alpha r_1^n + \beta r_2^n$ .
2. Найдём коэффициенты  $\alpha$  и  $\beta$ . Пусть  $f(0) = a, f(1) = b$ . Система уравнений

$$\begin{cases} \alpha + \beta = a \\ \alpha \cdot r_1 + \beta \cdot r_2 = b \end{cases}$$

имеет решение при любых  $a$  и  $b$ . Этими решениями являются

$$\alpha = (b - a \cdot r_2) / (r_1 - r_2).$$

$$\beta = (a \cdot r_1 - b) / (r_1 - r_2).$$

*Задача.*

Найдем формулу для общего члена последовательности Фибоначчи.

*Решение.*

Характеристическое уравнение имеет вид  $x^2 = x + 1$  или  $x^2 - x - 1 = 0$ , его корнями являются

числа  $\frac{1 \pm \sqrt{5}}{2}$ , значит, общее решение имеет вид  $f(n) = \alpha \left( \frac{1 + \sqrt{5}}{2} \right)^n + \beta \left( \frac{1 - \sqrt{5}}{2} \right)^n$ . Как

нетрудно видеть, из начальных условий  $f(0) = 0, f(1) = 1$  вытекает, что  $\alpha = -\beta = 1/\sqrt{5}$ , и, следовательно, общее решение последовательности Фибоначчи имеет вид:

$$f_n = \frac{1}{\sqrt{5}} \left[ \left( \frac{1 + \sqrt{5}}{2} \right)^n - \left( \frac{1 - \sqrt{5}}{2} \right)^n \right].$$

Что удивительно, это выражение при всех натуральных значениях  $n$  принимает целые значения.

### Случай равных корней характеристического многочлена.

Если характеристическое уравнение имеет два равных корня, то общее решение имеет вид:

$$f(n) = \alpha \cdot r_1^n + \beta \cdot n \cdot r_1^n.$$

*Доказательство.*

В случае равных корней характеристического уравнения выражение  $f(n) = \alpha \cdot r_1^n + \beta \cdot r_1^n$  нельзя считать общим решением, так как в формуле  $f(n) = \alpha \cdot r_1^n + \beta \cdot r_1^n = r_1^n \cdot (\alpha + \beta) = \delta \cdot r_1^n$  останется только одна постоянная и выбрать её так, чтобы удовлетворить двум начальным  $f(0)=a$ ,  $f(1)=b$  условиям, невозможно. Найдем второе решение отличное от  $r_1^n$ .

Если квадратное уравнение  $r^2 = C_1 r + C_2$  имеет два совпадающих корня  $r_1 = r_2$ , то по теореме Виета  $C_1 = 2 r_1$ , а  $C_2 = -r_1^2$ . Поэтому уравнение записывается так:

$$r^2 = 2 \cdot r_1 \cdot r - r_1^2.$$

Тогда рекуррентное соотношение имеет вид:

$$f(n+2) = 2 r_1 f(n+1) - r_1^2 f(n).$$

Докажем, что  $n r_1^n$  является решением данного рекуррентного соотношения.

Подставим  $n r_1^n$  в полученное рекуррентное соотношение, получим верное равенство:  $(n+2)r_1^{n+2} = 2r_1(n+1)r_1^{n+1} - r_1^2 n r_1^n = 2n r_1^{n+2} + 2r_1^{n+2} - n r_1^{n+2} = (n+2)r_1^{n+2}$ , значит,  $n r_1^n$  является решением данного рекуррентного соотношения.

Таким образом, имеем два решения рекуррентного соотношения:  $r_1^n$  и  $n r_1^n$ . Общее решение будет иметь вид:  $f(n) = \alpha r_1^n + \beta n r_1^n$ .

*Задача.*

Для последовательности с  $f(1)=0$  и  $f(2)=8$ , удовлетворяющей рекуррентному соотношению  $f(k+2)=4f(k+1)-4f(k)$ , выписать формулу общего члена.

*Решение.*

Характеристическое уравнение имеет вид:  $r^2 - 4r + 4 = 0$ , его корни  $r_1 = r_2 = 2$ . Общее решение рекуррентного соотношения:  $f(k) = \alpha \cdot 2^k + \beta \cdot k \cdot 2^k$ . Найдем коэффициенты  $\alpha$  и  $\beta$ , пользуясь тем, что  $f(1)=0$  и  $f(2)=8$ . Решим систему уравнений:

$$\begin{cases} 2 \cdot \alpha + 2 \cdot \beta = 0 \\ 4 \cdot \alpha + 8 \cdot \beta = 8 \end{cases}$$

Получим  $\alpha = -2$  и  $\beta = 2$ . Формула общего члена:  $f(n) = -2^{(n+1)} + n \cdot 2^{(n+1)}$ .

Линейные рекуррентные соотношения, порядок которых больше 2, решаются аналогично.

*Задача.*

Найдем общее решение рекуррентного соотношения:

$$f(n+4) = 5f(n+3) - 6f(n+2) - 4f(n+1) + 8f(n).$$

*Решение.*

Характеристическое уравнение имеет вид:  $r^4 - 5r^3 + 6r^2 + 4r - 8 = 0$

Решая его, получаем корни:  $r_1 = 2$ ,  $r_2 = 2$ ,  $r_3 = 2$ ,  $r_4 = -1$ .

Значит, общее решение имеет вид:  $f(n) = 2^{n-1}(\alpha + \beta \cdot n + \gamma \cdot n^2) + \delta \cdot (-1)^{n-1}$ .

## Асимптотики

Асимптотики – это искусство оценивания и сравнения скоростей роста функций.

Нередко в задачах на нахождение общего решения рекуррентного соотношения не требуется точное значение  $n$ -го члена, чаще бывает достаточной оценка порядка, а иногда требуется только оценка скорости роста функции  $f(n)$ . Для описания роста функции используется  $O$ -символика. (ввел Поль Бахман в 1894 г.)

Запись  $f(n)=O(g(n))$  означает, что существуют положительные константы  $M$  и  $n_0$ , такие, что  $|f(n)| \leq M \cdot |g(n)|$ , для всех целых  $n \geq n_0$ .

*Пример1.*

Для рекуррентного соотношения  $f(n)=5f(n-1)-6f(n-2)$ , с начальными членами  $f(1)=0$  и  $f(2)=13$  общее решение имеет вид  $f(n)=2^n+3^n$ . Можно сказать, что  $f(n)=O(3^n)$ .

Докажем это. Здесь  $M=2$ , а  $n_0=1$ .

Неравенство  $2^n+3^n \leq 3^n+3^n$  верно для всех  $n \geq 1$

$3^n+3^n=2 \cdot 3^n$ , значит,  $2^n+3^n \leq 2 \cdot 3^n$ , следовательно,  $f(n)=O(3^n)$

*Пример2.*

Для последовательности Фибоначчи, как было доказано выше, формула общего члена

имеет вид:  $f_n = \frac{1}{\sqrt{5}} \left[ \left( \frac{1+\sqrt{5}}{2} \right)^n - \left( \frac{1-\sqrt{5}}{2} \right)^n \right]$ . Легко доказать, что  $f_n=O(2^n)$ .

Соотношение  $f(n)=O(g(n))$  называют асимптотическим соотношением, оно означает, что функция  $f(n)$  растет не быстрее, чем функция  $g(n)$ . Определены еще два асимптотических отношения.

**Определение 1.** Функция  $f(x)$  асимптотически равна (эквивалентна)  $g(x)$  (обозначается  $f(x) \sim g(x)$ ) при  $x \rightarrow x_0$ , если  $\lim (f(x)/g(x))=1$ .

Если функция  $f(x)$  эквивалентна  $g(x)$ , то это означает, что  $f(x)$  растет с такой же скоростью, как и  $g(x)$ .

**Определение 2.**  $f(x)=o(g(x))$  при  $x \rightarrow x_0$ , если  $\lim (f(x)/g(x))=0$ .

Если  $f(x)=o(g(x))$ , то это означает, что функция  $f(x)$  растет медленнее, чем  $g(x)$ .

*Замечание.*

В соотношениях, использующих  $O$ -символику, левые и правые части не симметричны: правая часть всегда содержит меньше информации, чем левая, и поэтому нельзя в любом контексте заменять левую часть выражения правой. Например, из двух корректных асимптотических равенств  $x=O(x^2)$  и  $x^2=O(x^2)$  не следует, что  $x=x^2$ .

*Примеры.*

1. Полином асимптотически равен своему старшему члену:

$$\sum_{i=0}^n a_i x^i = O(x^n) \quad \text{при } x \rightarrow \infty.$$

2. Полином  $f(n)=2n^5+6n^4+6n^2+18$  есть  $O(n^5)$ .

3. Функция  $f(n)=2^n$  есть  $O(2^{n+1})$  и  $o(5^{n+1})$ .

4. Для линейного рекуррентного соотношения общее решение имеет вид:

$f(n)=C_1r_1^n+C_2r_2^n+\dots+C_kr_k^n$ . Если нас интересует только асимптотическое поведение последовательности, то достаточно рассмотреть лишь члены  $C_i(r_i)^n$ , у которых  $r_i$  имеет максимальное абсолютное значение среди тех членов, у которых  $C_i \neq 0$ .

Существуют оценки и асимптотики для комбинаторных чисел. Наиболее известна асимптотика для чисел  $n!$ , называемая формулой Стирлинга:  $n! \sim \sqrt{2\pi n} n^n e^{-n}$ .

$O$ -символику используют также для оценки сложности алгоритма. Одной из важнейших характеристик алгоритма является его временная сложность в худшем случае.

Пусть некоторая задача имеет размерность  $n$  (например, длина массива при сортировке). Обозначим через  $t(n)$  максимальное число действий, необходимое для решения задачи. Под действием понимают выполнение «простой» операции – любой арифметической операции, операции сравнения, присваивания и т. п. При этом сложность алгоритма зависит от конкретного вида команд. Поэтому при оценке интересует лишь асимптотическая сложность алгоритма, то есть порядок роста сложности при условии, что размер задачи неограниченно возрастает.

Алгоритм считается достаточно хорошим, если сложность этого алгоритма есть  $O(n^k)$  при некотором  $k > 0$ . В таком случае говорят, что задача может быть решена за полиномиальное время, а сам алгоритм называется полиномиальным.

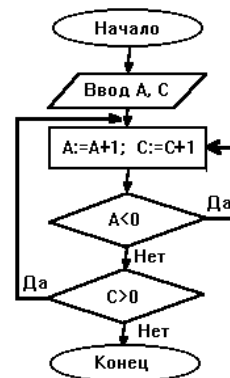
## Основы программирования

Из школьного курса информатики студент должен знать основные алгоритмические конструкции, уметь читать, исполнять на бумаге и тестировать алгоритм (блок-схему), писать несложные программы на одном из языков высокого уровня (Школьный алгоритмический, Паскаль, Бейсик) или уметь составлять блок-схемы.

### Пример 1.

При каких начальных значениях переменных алгоритм закончит работу:

- $A=-2; C=-1$
- $A=-2; C=-3$
- $A=-3; C=3$
- $A=-3; C=-2$
- $A=-4; C=-3$



### Решение.

Эта задача может быть несколькими способами: можно протестировать алгоритм, поочередно подставляя предлагаемые наборы значений переменных и, получая заикливание или окончание работы алгоритма.

Мы покажем более сложный аналитический способ решения, так как считаем, что для студента полезнее проанализировать конструкцию алгоритма и сформулировать ограничения на набор значений аргументов и потом уже проверить все наборы данных на соответствие этим требованиям.

Проанализируем конструкцию блок-схемы, представленной на рисунке. Так как в условии указаны только целые значения  $A$  и  $C$  будем вести рассуждения для целого типа данных. Обозначим  $C_{нач}$  и  $A_{нач}$  – начальные значения переменных  $C$  и  $A$ .

| Элемент блок-схемы | Анализ   |
|--------------------|--|
|                    | <p>Представленная конструкция есть цикл с постусловием. Выход из цикла осуществляется при ложном значении условия <math>A &lt; 0</math> (переменная <math>A</math> – параметр цикла). Т.е. исполнение цикла прервется если <math>A</math> станет неотрицательным.</p> <p>До проверки условия выхода команда изменения параметра цикла <math>A:=A+1</math> исполняется хотя бы один раз. Команда <math>A:=A+1</math> показывает, что значение величины <math>A</math> растет. Имеем <math>A_{нач} &lt; A</math>.</p> <p>Значит выход из цикла возможен, если растущий параметр будет иметь <b>наибольшее</b> значение <math>A+1=0</math>, т.е. <math>A=-1</math>,</p> <p>Отсюда имеем ограничения на значения отрицательной величины <math>A</math>: <math>A_{нач} &lt; A \leq -1</math>.</p> <p style="text-align: center;"><math>A_{нач} \leq -1</math></p>   |
|                    | <p>Предыдущая конструкция есть вложенный цикл. Внутри него величины <math>A</math> и <math>C</math> возрастут одинаково, за одно исполнение цикла каждая увеличится на шаг <math>= (+1)</math>.</p> <p>Вычислим, на сколько увеличится <math>C</math>, после исполнения вложенного цикла:</p> <p>Пусть цикл исполнялся <math>n</math> раз. Тогда значение величин <math>A</math> и <math>C</math> увеличится по сравнению с <math>A_{нач}</math> и <math>C_{нач}</math> на <math>n \cdot \text{шаг} = n \cdot 1 = n</math>.</p> <p>Если произошел выход из вложенного цикла, значит целое <math>A</math> стало равно 0.</p> <p>Имеем <math>A_{нач} + n = 0</math>. Значит <math>n = -A_{нач}</math>.</p> <p>Величина <math>C</math> до исполнения команд второй конструкции имеет значение <math>(C_{нач} + n = C_{нач} - A_{нач})</math>.</p> <p>Представленная в левом столбце конструкция есть цикл с постусловием.</p> <p>До проверки условия выхода команда изменения параметра цикла <math>C:=C+1</math> исполняется хотя бы один раз. Значит <math>C</math> примет значение</p> |

|  |
|--|
| $C_{нач}-A_{нач}+1$<br>Выход из цикла будет возможен, если выполнится условие $not(C>0) = (C \leq 0)$ .<br>Отсюда имеем ограничения на значения величины $C$ :<br>$C_{нач}-A_{нач}+1 \leq 0$ . |
|--|

Так как полученные условия должны выполняться одновременно, имеем следующие ограничения на начальные значения величин  $A$  и  $C$ :

$$A_{нач} \leq -1 \text{ и } C_{нач}-A_{нач}+1 \leq 0$$

Истинность конъюнкции («и») рассмотрена в разделе «Элементы алгебры логики».

| Данные условия | Значение условия $A_{нач} \leq -1$ | Значение условия $C_{нач}-A_{нач}+1 \leq 0$ | Значение условия $A_{нач} \leq -1 \text{ и } C_{нач}-A_{нач}+1 \leq 0$ | Вывод                    |
|----------------|------------------------------------|---|--|--------------------------|
| $A=-2; C=-1$   | $-2 \leq -1$<br>истина             | $-1-(-2)+1 \leq 0$<br>ложь                  | ложь   | Зацикливание             |
| $A=-2; C=-3$   | $-2 \leq -1$<br>истина             | $-3-(-2)+1 \leq 0$<br>истина                | истина   | Алгоритм закончит работу |
| $A=-3, C=3$    | $3 \leq -1$<br>ложь                | $3-3+1 \leq -1$<br>ложь                     | ложь   | Зацикливание             |
| $A=-3; C=-2$   | $-3 \leq -1$<br>истина             | $-2-(-3)+1 \leq 0$<br>ложь                  | ложь   | Зацикливание             |
| $A=-4, C=-3$   | $-4 \leq -1$<br>истина             | $-3-(-4)+1 \leq 0$<br>ложь                  | ложь   | Зацикливание             |

**Ответ:**       $A=-2; C=-1$

$A=-2; C=-3$

$A=-3, C=3$

$A=-3; C=-2$

$A=-4, C=-3$

Для решения задач необходимо усвоить алгоритмы решения основных задач, связанных с обработкой элементов одномерных массивов:

нахождение суммы элементов, обладающих определенным признаком;

нахождение произведения элементов, обладающих определенным признаком;

нахождение количества элементов, обладающих определенным признаком;

определение, есть или нет в массиве элемент с заданным свойством;

нахождение номера (индекса) крайнего (первого или последнего) элемента, обладающего определенным признаком;

нахождение максимального (минимального) элемента массива;

нахождение номера (индекса) максимального (минимального) элемента массива.

Приведем основную часть самых распространенных алгоритмов решения перечисленных выше задач без исполнения и подробных комментариев. В экзаменационном тесте элементы программы приведены на языке программирования Паскаль и на школьном алгоритмическом.

| № | Задача   | Система команд   |
|---|--|--|
| 1 | Нахождение суммы элементов, обладающих определенным признаком<br>$s$ – сумма;<br>$a[1:n]$ – массив;<br>$X(a[i])$ – признак элемента. | $s:=0$<br><u>нц для i от 1 до n</u><br>если $X(a[i])$<br>то $s:=s+a[i]$<br>все<br>кц |
| 2 | Нахождение произведения элементов, обладающих определенным признаком   | $p:=1$<br><u>нц для i от 1 до n</u>  |

|   |  |   |
|---|--|---|
|   | <p><math>p</math> – произведение;<br/> <math>a[1:n]</math> – массив;<br/> <math>X(a[i])</math> – признак сомножителя.</p>  | <p>если <math>X(a[i])</math><br/> то <math>p:=p*a[i]</math><br/> все<br/> кц</p>  |
| 3 | <p>Нахождение количества элементов, обладающих определенным признаком</p> <p><math>k</math> – сумма;<br/> <math>a[1:n]</math> – массив;<br/> <math>X(a[i])</math> – признак элемента.</p>  | <p><math>k:=0</math><br/> <u>нц для <math>i</math> от 1 до <math>n</math></u><br/> если <math>X(a[i])</math><br/> то <math>k:=k+1</math><br/> все<br/> кц</p>   |
| 4 | <p>Определение, есть или нет в массиве элемент с заданным свойством</p> <p>Soob-сообщение о наличии искомого элемента;<br/> <math>a[1:n]</math> – массив;<br/> <math>X(a[i])</math> – признак элемента.</p>                                | <p>Soob:=’нет’<br/> <u>нц для <math>i</math> от 1 до <math>n</math></u><br/> если <math>X(a[i])</math><br/> то Soob:=’да’<br/> все<br/> кц<br/> вывод (Soob);</p>   |
| 5 | <p>Нахождение номера крайнего (первого или последнего) элемента, обладающего определенным признаком</p> <p><math>np</math> – номер искомого элемента;<br/> <math>a[1:n]</math> – массив;<br/> <math>X(a[i])</math> – признак элемента.</p> | <p>Номер последнего элемента<br/> <math>np:=0</math><br/> <u>нц для <math>i</math> от 1 до <math>n</math></u><br/> если <math>X(a[i])</math><br/> то <math>np:=i</math><br/> все<br/> кц<br/> Номер первого элемента<br/> <math>np:=0</math><br/> <u>нц для <math>i</math> от <math>n</math> до 1 шаг (-1)</u><br/> если <math>X(a[i])</math><br/> то <math>np:=i</math><br/> все<br/> кц</p> |
| 6 | <p>Нахождение максимального элемента массива<br/> <math>max</math> – максимальный элемент;<br/> <math>a[1:n]</math> – массив.</p>  | <p><math>max:=a[1]</math><br/> <u>нц для <math>i</math> от 2 до <math>n</math></u><br/> если <math>a[i]&gt;max</math><br/> то <math>max:=a[i]</math><br/> все<br/> кц</p>   |
| 7 | <p>Нахождение номера минимального элемента</p> <p><math>np</math> – номер минимального элемента;<br/> <math>min</math> – минимальный элемент;<br/> <math>a[1:n]</math> – массив.</p>   | <p><math>min:=a[1]</math><br/> <math>np:=1</math>;<br/> <u>нц для <math>i</math> от 2 до <math>n</math></u><br/> если <math>a[i]&lt;min</math><br/> то <math>min:=a[i]; np:=i</math><br/> все<br/> кц</p>   |

**Пример 2.**

Задан одномерный массив  $x[1..N]$ . Фрагмент алгоритма

|   |   |
|---|---|
| <p><math>s:=0</math>;<br/> for <math>k:=1</math> to <math>N</math> do<br/> if (<math>0&lt;x[k]</math>)<br/> then <math>s:=s+x[k]</math></p> | <p><math>s:=0</math>; нц для <math>k</math> от 1 до <math>N</math><br/>   если <math>0&lt;x[k]</math><br/>     то <math>s:=s+x[k]</math><br/>   все<br/> кц</p> |
|---|---|

определяет:

- максимальный элемент массива                       сумму положительных элементов  
 количество положительных элементов               индекс последнего положительного элемента

**Решение.**

|  |  |
|--|--|
| Сравним данный алгоритм с имеющимися основными алгоритмами в таблице выше. |  |
|--|--|

|   |  |
|---|--|
| <p>Замечаем, что он содержит конструкцию (1) - <u>суммирование</u>.</p> <p>Сформулируем признак, по которому определяются суммируемые элементы:<br/> <math>0 &lt; x[k]</math><br/>         «k-ый элемент массива x - <b>положительный</b> (больше нуля)»</p> <p>Значит фрагмент алгоритма определяет <b>сумму положительных элементов</b></p> | <pre>s:=0; нц для k от 1 до N   если 0&lt;x[k]     то s:=s+x[k]   все кц</pre> |
|---|--|

**Ответ:**

- максимальный элемент массива                       сумму положительных элементов  
 количество положительных элементов             индекс последнего положительного элемента

Задачи могут содержать фрагменты линейных и разветвляющихся алгоритмов которые необходимо исполнить, или решить обратную задачу: найти исходные значения аргументов для которых получился заданный результат.

**Пример 3.**

Найдите значение переменной y после исполнения серии команд:

```
x:= -1;
y:=2;
y:=x*y;
y:=y*y;
x:=y-1;
y:=x-y;
```

**Решение:**

Наиболее удобно табличная запись пошагового исполнения алгоритма. Напомним, что после выполнения команды присваивания предыдущее значение величины, стоявшей слева от знака присваивания (:=) теряется. Таким образом, последнее (нижнее) неизменное значение величины является искомым.

| Команда                       | Вычисления      | Значение X    | Значение Y    |
|-------------------------------|-----------------|---------------|---------------|
| x:= -1;                       |                 | -1            | Не определено |
| y:=2;                         |                 | Не изменилось | 2             |
| y:=x*y;                       | $y=(-1)*2=-1$   | Не изменилось | -1            |
| y:=y*y;                       | $y=(-1)*(-1)=1$ | Не изменилось | 1             |
| x:=y-1;                       | $x=1-1=0$       | 0             | Не изменилось |
| y:=x-y;                       | $y=0-1=-1$      | Не изменилось | <b>-1</b>     |
| Искомое значение y= <b>-1</b> |                 |               |               |

**Ответ:**        y=-1.

**Пример 4.**

Найдите значение переменной y после исполнения серии команд:

```
y:=4;
y:=y+1;
y:=y*4;
y:=20-y;
y:=12-y;
y:=y*y-100;
```

**Решение:**

Заполним таблицу исполнения фрагмента алгоритма.

| Команда | Вычисления | Значение Y |
|---------|------------|------------|
| y:=4    |            | 4          |
| y:=y+1  | $4+1=5$    | 5          |



|                         |                |    |
|-------------------------|----------------|----|
| $y:=y*4$                | $5*4=20$       | 20 |
| $y:=20-y$               | $20-20=0$      | 0  |
| $y:=12-y$               | $12-0=12$      | 12 |
| $y:=y*y-100$            | $12*12-100=44$ | 44 |
| Искомое значение $y=44$ |                |    |

**Ответ:**  $y=44$ .

Обратной задачей к двум решенным выше является поиск начальных значений аргументов по известным значениям результатов после исполнения фрагмента алгоритма. Для линейных фрагментов решение заключается в рассмотрении результатов команд в направлении обратном исполнению алгоритма, то есть **снизу вверх**. В качестве результата исполнения оператора берется значение аргумента из последующей команды. Рассмотрим сказанное на примере.

### Пример 5.

Определите, каково было значение величины А, если после исполнения серии команд получилось  $D=15$ .

$$V:=A*4-2*V$$

$$D:=26-V$$

$$D:=5+D$$

### Решение.

1. В таблицу исполнения алгоритма запишем команды алгоритма в обратном порядке.

| Команда                 | Вычисления                          | Значение D   | Значение V    | Значение A   |
|-------------------------|-------------------------------------|--|---------------|--------------|
|                         |                                     | 15<br>Подставляем 15 только в левую часть команды $D:=5+D$ | Не вычислено  | Не вычислено |
| $D:=5+D$                | $15=5+D$<br>$D=10$                  | 10   | Не вычислено  | Не вычислено |
| $D:=26-V$               | $10=26-V$<br>$V=16$                 | Не изменилось  | 16            | Не вычислено |
| $V:=A*4-2*V$            | $16=A*4-2*16$<br>$48=4*A$<br>$A=12$ | Не изменилось  | Не изменилось | 12           |
| Искомое значение $A=12$ |                                     |  |               |              |

**Ответ:**  $A=12$ .

В случае, если фрагмент алгоритма содержит команду ветвления, решение будет легче, если сначала создать наглядную модель задачи.

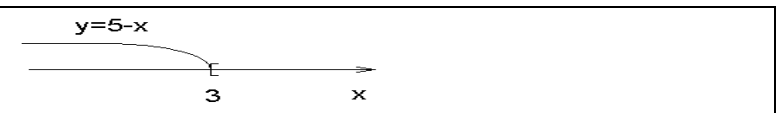

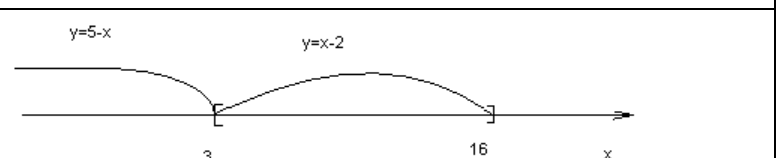
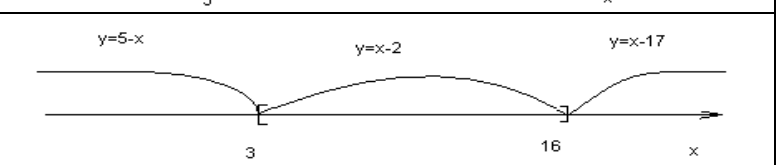
### Пример 6.

Укажите все значения величины  $x$ , при которых после исполнения фрагмента программы получится ответ  $y=3$

| Pascal  | Школьный алгоритмический язык  |
|---|--|
| <pre>If x&lt;3 then y:=5-x else if x&lt;=16 then y:=x-2 else y:=x-17;</pre> | <pre>если x&lt;3 то y:=5-x иначе если x&lt;=16 то y:=x-2 иначе y:=x-17 все все</pre> |

### Решение.

1. Создадим математическую модель условия задачи. Изобразим числовую прямую, отметим на ней точки – границы числовых промежутков, над которыми надпишем уравнения для вычисления значения переменной  $y$ . Символ [ на числовой прямой будет означать, что точка принадлежит числовому промежутку справа, ] - промежутку слева.

|                                  |  |
|----------------------------------|--|
| если $x < 3$ то $y := 5 - x$     |  |
| иначе                            |  |
| если $x \leq 16$ то $y := x - 2$ |  |
| иначе $y := x - 17$              |  |

2. На интервале  $x \in (-\infty; 3)$  имеем  $y = 5 - x$  и по условию  $y = 3$

$$3 = 5 - x$$

$$x = 2,$$

Проверим, принадлежит ли полученное значение переменной  $x$  рассматриваемому интервалу:

$$2 \in (-\infty; 3).$$

Значит, при  $x = 2$  получим  $y = 3$ .

3. На отрезке  $x \in [3; 16]$  имеем  $y = x - 2$  и по условию  $y = 3$

$$3 = x - 2$$

$$x = 5$$

$$5 \in [3; 16]$$

Значит, при  $x = 5$  получим  $y = 3$ .

4. На интервале  $x \in (16; +\infty)$  имеем  $y = x - 17$  и по условию  $y = 3$

$$3 = x - 17$$

$$x = 20$$

$$20 \in (16; +\infty)$$

Значит, при  $x = 20$  получим  $y = 3$ .

**Ответ:** при  $x = 2$  или  $x = 5$  или  $x = 20$  переменная  $y$  примет значение 3.

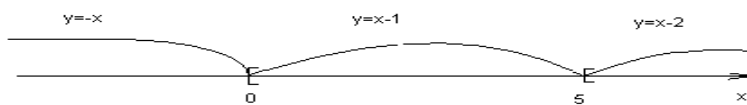
### Пример 7.

Укажите все значения  $x$ , при которых после исполнения фрагмента программы получится ответ  $y = 5$

|  |   |
|--|---|
| <pre>If x &lt; 0 then y := -x else if x &gt;= 5 then y := x - 2 else y := x - 1;</pre> | <pre>если <math>x &lt; 0</math> то <math>y := -x</math> иначе если <math>x \geq 5</math> то <math>y := x - 2</math> иначе <math>y := x - 1</math> все все</pre> |
|--|---|

Решение.

По аналогии с решенной выше задачей создадим математическую модель решения задачи.



На интервале  $x \in (-\infty; 0)$  имеем  $y = -x$  и по условию  $y = 5$

$$5 = -x$$

$$x = -5,$$

Проверим, принадлежит ли полученное значение переменной  $x$  рассматриваемому интервалу:

$$5 \in (-\infty; 0).$$

Значит при  $x = -5$  получим  $y = 5$ .

На луче  $x \in [0; 5)$  имеем  $y = x - 1$  и по условию  $y = -5$

$$5 = x - 1$$

$$x = 6$$

$$6 \notin [0; 5)$$

Значит на этом интервале нет значений  $x$ , при которых  $y = 5$ .

На луче  $x \in [5; +\infty)$  имеем  $y = x - 2$  и по условию  $y = 5$

$$5 = x - 2$$

$$x = 7,$$

$$7 \in [5; +\infty)$$

Значит при  $x = 7$  получим  $y = 5$ .

**Ответ:** при  $x = -5$  или  $x = 7$  переменная  $y$  примет значение 5.

### Пример 8.

Определите количество выполнений тела цикла в программе

|   |  |
|---|--|
| $C := \text{true}; x := 64;$<br>while C do<br>begin $C := (x >= 4);$<br>$x := x \text{ div } 6 + 2;$ end; | $V := \text{да}; x := 64$<br>нц пока V<br>  $V := (x >= 4); x := \text{div}(x, 6) + 2$<br>кц |
|---|--|

### Решение.

*Справка 1:* правила исполнения функций  $\text{div}$  и  $\text{mod}$ .

Функция  $\text{div}(a, b)$  вычисляет частное от деления с остатком целого числа  $a$  на целое число  $b$ .

Функция  $\text{mod}(a, b)$  вычисляет остаток при целочисленном делении целого числа  $a$  на целое число  $b$ . Значение функции всегда принадлежит промежутку  $0 \leq \text{mod}(a, b) < b$

$$a = \text{div}(a, b) * b + \text{mod}(a, b)$$

число = частное \* делитель + остаток

| a  | b | $a = \text{div}(a, b) * b + \text{mod}(a, b)$ | $\text{div}(a, b)$ | $\text{mod}(a, b)$ | $0 \leq \text{mod}(a, b) < b$ |
|----|---|---|--------------------|--------------------|-------------------------------|
| 33 | 7 | $33 = 4 * 7 + 5$                              | 4                  | 5                  | $0 \leq 5 < 7$                |
| 6  | 7 | $6 = 0 * 7 + 6$                               | 0                  | 6                  | $0 \leq 6 < 7$                |
| 56 | 7 | $56 = 8 * 7 + 0$                              | 8                  | 0                  | $0 \leq 0 < 7$                |

*Справка 2:* в приведенном примере условие в заголовке цикла «пока» представляет собой имя переменной логического типа (C может принимать значения только «да» или «нет»). Команда присваивания  $C := (x >= 4);$  возможна для величин этого типа. Например, при  $x = 64$  условие  $x >= 4$  примет значение «да», так как  $64 >= 4$ . Следовательно, C тоже примет значение «да».

Таблица исполнения фрагмента алгоритма

| Команда, условие                                 |   |   |   |   |                                   |
|--|---|---|---|---|-----------------------------------|
| $C := \text{да}$                                 |   |   |   |   |                                   |
| $x := 64$  |   |   |   |   |                                   |
| нц пока V  | пока да<br>входим в тело<br>цикла             | пока да<br>входим в тело<br>цикла           | пока да<br>входим в тело<br>цикла         | пока да<br>входим в тело<br>цикла         | пока нет<br><b>выход из цикла</b> |
| $x >= 4$   | $64 >= 4$ да                                  | $12 >= 4$ да                                | $4 >= 4$ да                               | $2 >= 4$ нет                              |                                   |
| $C := (x >= 4);$                                 | да  | да  | да  | нет                                       |                                   |
| $\text{div}(x, 6)$                               | $64 = 10 * 6 + 4$<br>$\text{div}(64, 6) = 10$ | $12 = 2 * 6 + 0$<br>$\text{div}(12, 6) = 2$ | $4 = 0 * 6 + 4$<br>$\text{div}(4, 6) = 0$ | $2 = 0 * 6 + 2$<br>$\text{div}(4, 6) = 0$ |                                   |
| $x := \text{div}(x, 6) + 2$                      | 12  | 4   | 2   | 2   |                                   |
| кц   | Возвращает на<br>нц                           | Возвращает на<br>нц                         | Возвращает на<br>нц                       | Возвращает на<br>нц                       |                                   |
| 1 - исполнение<br>цикла<br>0 - выход из<br>цикла | 1   | 1   | 1   | 1   | 0<br>Цикл исполнился 4 раза       |

Чтобы узнать количество исполнений цикла подсчитаем количество (сумму) единиц в последней строке.

**Ответ:** цикл исполнится 4 раза.

Задачи на нахождение значения какой-либо величины решаются аналогично. Составляется таблица исполнения алгоритма. Последнее значение величины в таблице будет искомым.

### Пример 9.

Укажите значение переменной k после исполнения алгоритма

|   |   |
|---|---|
| <pre>PROGRAM UPR12; VAR I,K,N: INTEGER; A: ARRAY[1..10] OF INTEGER;  BEGIN K:=0; FOR I:=1 TO 10 DO   A[I]:=I+I MOD 3; FOR I:=1 TO 10 DO BEGIN   N:=A[I] MOD 3;   IF TRUNC(A[I]/4)=N   THEN K:=K+I MOD 3 END; WRITE(K); END.</pre> | <pre>алг таблица(рез цел K) нач цел I, цел таб A[1:10]  K:=0 нц для I от 1 до 10   A[I]:=I+mod(I,3) кц нц для I от 1 до 10   если int(A[I]/4)=mod(A[I],3)   то K:=K+mod(I,3)   все кц вывод K кон</pre> |
|---|---|

Решение.

Анализируя текст программы, видим, что первый цикл с параметром заполняет значениями целочисленный массив A.

Составим таблицу для фрагмента заполнения массива A. Так как в массиве 10 элементов, то удобно расположить имена величин и значения функций в столбцах. Одно исполнение тела цикла будет соответствовать одной строке таблицы.

| Команда  | I  | нц для I от 1 до 10 | mod(I,3)                | A[i]:= I+mod(I,3) | кц (шаг 1)                 |
|----------|----|---------------------|-------------------------|-------------------|----------------------------|
| Значения | 1  | входим в тело цикла | 1=0*3+1<br>mod(1,3)=1   | A[1]=2            | Возвращает на нц, i:=i+шаг |
|          | 2  | входим в тело цикла | 2=0*3+2<br>mod(2,3)=2   | A[2]=4            | Возвращает на нц, i:=i+шаг |
|          | 3  | входим в тело цикла | 3=1*3+0<br>mod(3,3)=0   | A[3]=3            | Возвращает на нц, i:=i+шаг |
|          | 4  | входим в тело цикла | 4=0*3+1<br>mod(4,3)=1   | A[4]=5            | Возвращает на нц, i:=i+шаг |
|          | 5  | входим в тело цикла | 5=1*3+2<br>mod(5,3)=2   | A[5]=7            | Возвращает на нц, i:=i+шаг |
|          | 6  | входим в тело цикла | 6=2*3+0<br>mod(6,3)=0   | A[6]=6            | Возвращает на нц, i:=i+шаг |
|          | 7  | входим в тело цикла | 7=2*3+1<br>mod(7,3)=1   | A[7]=8            | Возвращает на нц, i:=i+шаг |
|          | 8  | входим в тело цикла | 8=2*3+2<br>mod(8,3)=2   | A[8]=10           | Возвращает на нц, i:=i+шаг |
|          | 9  | входим в тело цикла | 9=3*3+0<br>mod(9,3)=0   | A[9]=9            | Возвращает на нц, i:=i+шаг |
|          | 10 | входим в тело цикла | 10=3*3+1<br>mod(10,3)=1 | A[10]=11          | <b>выход из цикла</b>      |

Данные этой таблицы из столбца A[i] перенесем в следующую, так как они необходимы для дальнейших вычислений. Для исполнения второго цикла кое-что поясним: функция int(a) равна целой части числа a. Составим таблицу для исполнения второго фрагмента программы.

| Команда  | I | нц для I от 1 до 10 | A[I]   | mod(A[i],3)           | int(A[I]/4) | mod(A[I],3)=<br>int(A[I]/4)= | K     | кц (шаг 1)                 |
|----------|---|---------------------|--------|-----------------------|-------------|------------------------------|-------|----------------------------|
| Значения | 1 | входим в тело цикла | A[1]=2 | 2=0*3+2<br>mod(2,3)=2 | int(2/4)=0  | 2=0<br>ложь                  | 0     | Возвращает на нц, i:=i+шаг |
|          | 2 | входим в тело цикла | A[2]=4 | 4=1*3+1<br>mod(4,3)=1 | int(4/4)=1  | 1=1<br>истина                | 0+1=1 | Возвращает на нц, i:=i+шаг |
|          | 3 | входим в тело цикла | A[3]=3 | 3=1*3+0               | int(3/4)=0  | 0=0                          | 1+0=1 | Возвращает на              |

|    |                     |            |                                    |                      |               |       |                                     |
|----|---------------------|------------|------------------------------------|----------------------|---------------|-------|-------------------------------------|
|    | тело цикла          |            | $\text{mod}(3,3)=0$                |                      | истина        |       | нц, $i:=i+\text{шаг}$               |
| 4  | входим в тело цикла | $A[4]=5$   | $5=1*3+2$<br>$\text{mod}(5,3)=2$   | $\text{int}(5/4)=1$  | 2=1<br>ложь   | 1     | Возвращает на нц, $i:=i+\text{шаг}$ |
| 5  | входим в тело цикла | $A[5]=7$   | $7=2*3+1$<br>$\text{mod}(7,3)=1$   | $\text{int}(7/4)=1$  | 1=1<br>истина | 1+1=2 | Возвращает на нц, $i:=i+\text{шаг}$ |
| 6  | входим в тело цикла | $A[6]=6$   | $6=2*3+0$<br>$\text{mod}(6,3)=0$   | $\text{int}(6/4)=1$  | 0=1<br>ложь   | 2     | Возвращает на нц, $i:=i+\text{шаг}$ |
| 7  | входим в тело цикла | $A[7]=8$   | $8=2*3+2$<br>$\text{mod}(8,3)=2$   | $\text{int}(8/4)=2$  | 2=2<br>истина | 2+2=4 | Возвращает на нц, $i:=i+\text{шаг}$ |
| 8  | входим в тело цикла | $A[8]=10$  | $10=3*3+1$<br>$\text{mod}(10,3)=1$ | $\text{int}(10/4)=2$ | 1=2<br>ложь   | 4     | Возвращает на нц, $i:=i+\text{шаг}$ |
| 9  | входим в тело цикла | $A[9]=9$   | $9=3*3+0$<br>$\text{mod}(9,3)=0$   | $\text{int}(9/4)=2$  | 0=2<br>ложь   | 4     | Возвращает на нц, $i:=i+\text{шаг}$ |
| 10 | входим в тело цикла | $A[10]=11$ | $11=3*3+2$<br>$\text{mod}(11,3)=2$ | $\text{int}(11/4)=2$ | 2=2<br>истина | 4+2=6 | <b>выход из цикла</b>               |

На экран будет выведено значение  $k=6$ .

**Ответ:**  $k=6$ .

Несколько проще задачи на определение результатов величины после исполнения вложенных циклов. Однако эти задания требуют внимательности и аккуратности.

### Пример 10.

Определите значение переменной  $S$  после выполнения операторов:

|  |  |
|--|--|
| for $i:= 1$ to 2 do begin<br>$S:=1$ ;<br>for $j:= 2$ to 3 do<br>$S:=S+i+j$ ;<br>$S:=S+1$ ;<br>end;<br>end; | нц для $i$ от 1 до 2<br>$S:=1$<br>нц для $j$ от 2 до 3<br>$S:=S+i+j$<br>кц<br>$S:=S+1$<br>кц |
|--|--|

При решении этой задачи необходимо помнить, что вложенный цикл выполняется вновь от начального значения до конечного при каждом повторении внешнего цикла. Величина  $S$  накапливает сумму значений параметров циклов  $i$  и  $j$ .

Достаточно опытные в программировании абитуриенты обратят внимание на то, что первой командой тела наружного цикла является присваивание числовой константы переменной  $S$ . Значит, значение величины  $S$  не зависит от накопленных сумм при предыдущих исполнениях цикла. Получается, что для решения задачи достаточно 1 раз исполнить внешний цикл по  $i$  для ее конечного значения.

Однако при выполнении теста указывается окончательный результат, а как он получен – рационально или нет, не учитывается. Поэтому мы приведем полную таблицу исполнения алгоритма, выделив последнее исполнение внешнего цикла как определяющего искомое значение величины  $S$ .

| Величины                       |                                 |           | Пояснения                                  |
|--------------------------------|---------------------------------|-----------|--|
| нц для $i$ от 1 до 2<br>шаг= 1 | нц для $j$ от 2 до 3<br>шаг = 1 | $S$       |  |
| $i$                            | $j$                             |           |  |
| 1                              |                                 | 1         | <b><math>S:=S+i+j</math></b>               |
|                                | 2                               | 4         | $S=1+1+2=4$                                |
|                                | 3                               | 8         | $S=4+1+3=8$                                |
|                                |                                 | 9         | Цикл по $j$ закончен<br>$S:=S+1$           |
| $i:=i+\text{шаг}$              |                                 |           |  |
| 2                              |                                 | 1         | Предыдущее значение $S$<br><b>потеряно</b> |
|                                | 2                               | 5         | $S=1+2+2=5$                                |
|                                | 3                               | 10        | $S=5+2+3=10$                               |
|                                |                                 |           | Цикл по $j$ закончен                       |
|                                |                                 | <b>11</b> | $S:=S+1$                                   |
|                                |                                 |           | Цикл по $i$ закончен                       |

Последнее значение  $S=11$  – искомое.

Из таблицы видно, что все исполнения внешнего цикла, кроме последнего – лишние.

**Ответ:**  $S=11$ .

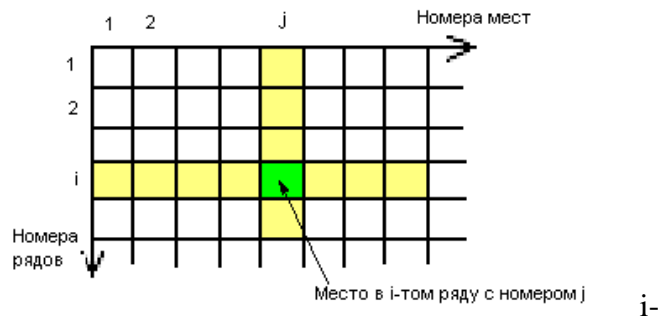
### Пример 11.

Составьте программу на одном из языков программирования для решения следующей задачи: В зрительном зале 23 ряда, в каждом из которых 40 мест. Информация о проданных билетах хранится в двумерном массиве, номера строк которых соответствуют номерам рядов, а номера столбцов – номерам мест. Если билет на то или иное место продан, то соответствующий элемент массива имеет значение 1, в противном случае – 0. Определить количество рядов, в которых есть свободные места.

#### Решение.

На каждом билете в зрительный зал содержит информацию о номере ряда и о номере места в этом ряду. Изобразим графически расположение мест в зрительном зале и пронумеруем ряды и места.

По условию, данные о продаже билетов занесены в массив. Размер массива целесообразно выбрать 23×40 по числу рядов и кресел в ряду. Тип элементов массива – целый, так как в него заносятся только 0 или 1. Дадим массиву имя ZZ (от «Зрительный Зал»). Тогда ZZ[i,j] – информация о месте в зрительном зале: i-тый ряд, кресло с номером j.



Обратите внимание, что в задаче требуется определить не количество пустых мест, а количество рядов, в которых есть непроданные места. Поэтому для каждого ряда из 23 надо проверить, есть ли в нем хотя бы одно не проданное место. Есть смысл ввести вспомогательную величину (soob), которая, например, равна «нет», если в ряду все места проданы, и принимает значение «да», если встретится хотя бы одно непроданное место. Если такое место есть (есть 0), то искомое количество рядов надо увеличить на 1. Величине «КОЛИЧЕСТВО РЯДОВ» дадим имя k – целое число.

#### Таблица величин

| Величина  | Имя  | Тип           | Назначение               | Ограничения на принимаемые значения       |
|---|------|---------------|--------------------------|---|
| Массив данных   | ZZ   | <u>целтаб</u> | <u>арг</u>               | <u>целтаб</u> [1:23,1:40]                 |
| Номер ряда  | i    | <u>цел</u>    | вспомогательная величина | от 1 до 23                                |
| Номер кресла  | j    | <u>цел</u>    | вспомогательная величина | от 1 до 40                                |
| ZZ[i,j] - информация о месте в зрительном зале: i-тый ряд, кресло с номером j |      |               |                          |   |
| ZZ[i,j]=0 – условие того, что в ряду i место j не продано                     |      |               |                          |   |
| Количество рядов, в которых есть непроданное место                            | k    | <u>цел</u>    | <u>рез</u>               | 0 ≤ k ≤ 23<br>k=0, если искомым рядов нет |
| Сообщение о наличии непроданного места  | Soob | <u>лит</u>    | вспомогательная величина | «нет», «да»                               |

#### Напишем фрагмент алгоритма поиска непроданного места в i-том ряду

|  |  |
|--|--|
| 1 способ<br>проверка всех элементов в i-той строке   | 2 способ<br>поиск до первого элемента, удовлетворяющего признаку                   |
| Soob:= 'нет'<br><u>нц</u> для j от 1 до 40<br><u>если</u> ZZ[i,j]=0<br><u>то</u> Soob:= 'да' | Soob:= 'нет'<br>j:=1<br><u>нц</u> пока Soob='нет' и j<=40<br><u>если</u> ZZ[i,j]=0 |

|                         |   |
|-------------------------|---|
| <u>все</u><br><u>кц</u> | <u>то</u> Soob:='да'<br><u>все</u><br>j:=j+1<br><u>кц</u> |
|-------------------------|---|

Проверить, как работает каждый фрагмент, предоставляем читателю. Отметим только, что второй способ рациональнее.

Фрагмент проверки того, есть ли в ряду непроданное место, и подсчета количества рядов.

если Soob:='да'

то k:=k+1

все

Для рационального способа команды ветвления лучше объединить в одну. Отметим, что до того, как проверять ряды на наличие непроданных мест, значение величины k должно быть присвоено 0.

Соединим все фрагменты в один алгоритм.

| 1 способ  | 2 способ (рациональный)   |
|---|---|
| <u>алг</u> Кинотеатр ( <u>целтаб</u> [1:23,1:40] ZZ, цел k)<br><u>арг</u> ZZ<br><u>рез</u> k<br><u>нач цел</u> i,j; <u>лит</u> Soob<br>k:=0<br><u>нц для</u> i <u>от</u> 1 <u>до</u> 23<br>Soob:='нет'<br><u>нц для</u> j <u>от</u> 1 <u>до</u> 40<br><u>если</u> ZZ[i,j]=0<br><u>то</u> Soob:='да'<br><u>все</u><br><u>кц</u><br><u>если</u> Soob:='да'<br><u>то</u> k:=k+1<br><u>все</u><br><u>кц</u><br><u>вывод</u> (k)<br><u>кон</u> | <u>алг</u> Кинотеатр ( <u>целтаб</u> [1:23,1:40] ZZ, цел k)<br><u>арг</u> ZZ<br><u>рез</u> k<br><u>нач цел</u> i,j; <u>лит</u> Soob<br>k:=0<br><u>нц для</u> i <u>от</u> 1 <u>до</u> 23<br>Soob:='нет'<br>j:=1<br><u>нц пока</u> Soob='нет' и j<=40<br><u>если</u> ZZ[i,j]=0<br><u>то</u> Soob:='да'<br>k:=k+1<br><u>все</u><br>j:=j+1<br><u>кц</u><br><u>вывод</u> (k)<br><u>кон</u> |

## **Лабораторные работы по основам программирования**

### **Лабораторная работа №1**

#### **Знакомство с клавиатурой. Организация работы в среде Турбопаскаль**

Цель работы: научиться запускать среду Турбопаскаль, загружать, создавать, записывать файлы, компилировать и исполнять программы, ознакомиться с наиболее распространенными операциями обработки текстов в редакторе среды Турбопаскаль.

Замечание: перед работой должна быть выполнена контролирующая программа по теме "Величина".

#### **Задание 1**

- 1.1 С разрешения преподавателя включите компьютер.
- 1.2 Войдите в Турбопаскаль способом, рассказанным на лекции.

#### **Задание 2**

- 2.1 Загрузите программу с именем PRIMER1.pas. Для этого:
  - выйдите в верхнее меню (F10),
  - поместите курсор на команду FILE,
  - в выпадающем меню установите курсор на команду Load,
  - нажмите ввод (↵),
  - после появления сообщения \*.pas,
  - нажмите еще раз ввод (↵)
  - установите курсор на имя PRIMER1.PAS и ведите программу (↵) в редактор Pascal-я.
- 2.2 Откомпилируйте программу: (признак окончания компиляции - мерцающая фраза "Press any key" ("нажми любую клавишу")):
  - 2.2.1 Через верхнее меню (COMPILE).
  - 2.2.2 Через Alt+F9.
- 2.3 Исполните программу (признак окончания исполнения - выход в окно редактора):
  - 2.3.1 Через верхнее меню (RUN).
  - 2.3.2 Через Ctrl+F9.
- 2.4 Просмотрите результаты исполнения (возврат - нажатие любой клавиши):
  - 2.4.1 Через верхнее меню (RUN, USER SCREEN).
  - 2.4.2 Через Alt+F5.
- 2.5 Удалите символы ремарки ( { } ) и вновь исполните программу.
- 2.6 Удалите программу из памяти редактора Паскаля, не записывая изменения (на вопрос "Save... (Y/N)?" дать ответ "N").

#### **Задание 3**

Загрузите файл с именем PRIMER2.pas (повторив по аналогии серию команд, описанную в задании 2). Восстановите заголовок, откомпилируйте и исполните программу, просмотрите результат исполнения. Подготовьте набор тестовых значений для аргументов и результатов, объясните причину выбора именно этих чисел.

Удалите программу из памяти редактора Pascal-я, не записывая изменения (на вопрос "Save... (Y/N)?" дать ответ "N").



## Задание 4

4.1 Загрузите файл с именем PRIMER3.pas. Скопируйте строку с запросом значения величины длины катета  $a$ , исправьте имя величины так, чтобы запрашивалась длина катета  $b$ . Подберите удобные для устных вычислений тестовые данные. Снимите выделение блока.

4.2 Опишите величины:

тангенс угла  $A$  - имя tga;

тангенс угла  $B$  - имя tgb;

площадь треугольника - имя S;

радиус описанной окружности - имя R;

длину медианы СК - имя sk;

организуя вычисление и вывод на экран, скопировав нужное количество раз и исправив оператор вывода длины гипотенузы. При исправлении операторов перейдите в режим заборя (клавиши Ins или Insert). Откомпилируйте и исполните программу, просмотрите результат исполнения. Снимите выделение блока.

Ctrl+k, b - отмечает начало блока;

Ctrl+k, k - отмечает конец блока (при этом блок выделяется инверсной строкой);

Ctrl+k, c - копирование блока (до выполнения этой команды переместите курсор на строку, перед которой должен будет встать блок);

Ctrl+k, h - снятие выделения с блока.

4.3 Вставьте пустую строку между операторами нахождения площади и радиуса описанной окружности. Организуйте в этой строке нахождение и вывод значения периметра треугольника (имя - p) любым способом набора операторов (копирование или набор с клавиатуры). Выделите блок строк нахождения радиуса описанной окружности.

Перенесите его в строки ниже вычисления периметра. Исправьте формулу так, чтобы вычислялся радиус вписанной окружности. Откомпилируйте и исполните программу, просмотрите результат исполнения.

Ctrl+n - вставка пустой строки.

Ctrl+k, v - перенос блока.

4.4 Удалите строку вычисления значения медианы СК. Отметьте блок строк вычислений, организованных в пунктах 4.2-4.3 и разом удалите его.

Ctrl+y, - удаление строки, в которой находится курсор.

Ctrl+k, y - удаление выделенного блока.

4.5 Очистите окно редактора (FILE, NEW).

4.6 Изучите требования к отчету. Спишите необходимые данные из окна помощи, нажав F1 находясь в окне редактора.

4.6 Выйдите из Паскаля через верхнее меню. Войдите в Паскаль (смотрите задание 1). Выйдите из Паскаля через Alt+x.

## Задание 5

Выключите компьютер.

Требования к отчету: отчет должен содержать список из 27 команд редактора Паскаля, относящихся к перемещению, выделению, вставке и удалению (символов, слов строк и блоков). Данные взять из трех первых страниц помощи в окне редактирования.

## **Лабораторная работа №2.**

### **Команды присваивания, ввода и вывода. Составление простейших программ на языке Турбопаскаль**

Цель работы: научиться составлять простейшие программы на языке Турбопаскаль, содержащие команды ввода, вывода и их модификаций, команд указания позиции курсора на экране, очистки экрана. Повторить правила запуска среды Турбопаскаль, загрузки, создания, записи файлов, редактирования, компиляции и исполнения программ.

Замечание: перед работой должна быть выполнена контролирующая программа по теме "Команда присваивания".

#### **Задание 1**

- Загрузите программу с именем PRIMER1.pas.
- Откомпилируйте и исполните программу.
- Вставьте оператор очистки экрана так, чтобы при просмотре результатов исполнения программы на экране остались только ваши анкетные данные (не было бы строк, начинающихся со слова input).
- Организуйте вывод данных в столбик в центре экрана.
- Сохраните программу под своим именем и удалите ее из редактора Паскаля.

#### **Задание 2**

Создайте свою программу так, чтобы решалась одна из перечисленных ниже задач (выбор задачи осуществляется преподавателем). Откомпилируйте программу и исполните ее, подобрав тестовые данные. Протестированную программу сохраните.

- 2.1. Вычислить площадь треугольника по трем сторонам.
- 2.2. Вычислить площадь треугольника по двум сторонам и углу между ними, заданному в градусах.
- 2.3. Вычислить площадь параллелограмма по двум сторонам и углу между ними, заданному в градусах.
- 2.4. Вычислить площадь четырехугольника по двум диагоналям и углу между ними, заданному в градусах.
- 2.5. Вычислить площадь равнобокой трапеции по большему основанию, боковой стороне и углу между ними, заданному в градусах.
- 2.6. Вычислить площадь равнобокой трапеции по длинам оснований и боковой стороне.
- 2.7. Вычислить площадь правильного  $n$ -угольника со стороной произвольной длины, вписанного в окружность радиуса  $R$ .
- 2.8. Вычислить длину окружности радиуса  $R$ .
- 2.9. Вычислить площади круга ( $R$ ) и эллипса с полуосями  $a$  и  $b$ .

#### **Задание 3**

- Очистите окно редактирования (FILE, NEW).
- Введите и исполните программу PRIMER4.PAS.
- Исправьте программу так, чтобы на рисунке изображалась одна из букв ваших инициалов (по выбору преподавателя).

#### Задание 4

Исправьте предыдущую программу так, чтобы созданный вами рисунок выводился в указанном ниже месте экрана.

- 4.1. В верхнем правом углу.
- 4.2. В правом нижнем углу.
- 4.3. В левом нижнем углу.
- 4.4. По центру экрана.
- 4.5. На средней вертикальной линии экрана сверху.
- 4.6. На средней вертикальной линии экрана снизу.
- 4.7. На средней горизонтальной линии экрана справа.
- 4.8. На средней горизонтальной линии экрана слева.

#### Задание 5

- 5.1. Заданы три корня кубического уравнения:  $x_1$ ,  $x_2$ ,  $x_3$ . Найти коэффициенты этого уравнения.
- 5.2. Заданы координаты точки подвески математического маятника  $A(x_0, y_0, z_0)$  и координаты одной из точек его наивысшего подъема  $B(x_1, y_1, z_1)$ . Найти координаты самой низкой точки траектории и другой наивысшей точки подъема.
- 5.3. У квадрата ABCD на плоскости известны координаты двух противоположных вершин – точек A и C. Найти координаты точек B и D.
- 5.4. Русские неметрические единицы длины: 1 верста = 500 сажений; 1 сажень = 3 аршина; 1 аршин = 16 вершков; 1 вершок = 44,45 мм. Длина некоторого отрезка составляет  $p$  метров. Перевести ее в русскую неметрическую систему.
- 5.5. Найти координаты вершины параболы  $y = ax^2 + bx + c$
- 5.6. Треугольник ABC задан длинами своих сторон. Найти длину высоты, опущенной из вершины A.

#### Лабораторная работа №3

##### Команды ветвления и выбора на языке Турбопascal

Цель работы: Ознакомиться с командами ветвления и выбора, научиться составлять программы с использованием этих команд на языке Паскаль.

Замечание: перед работой должна быть выполнена контролирующая программа по теме "Команда ветвления".

#### Задание 1

Составьте программу с использованием команды ветвления для решения одной из следующих задач:

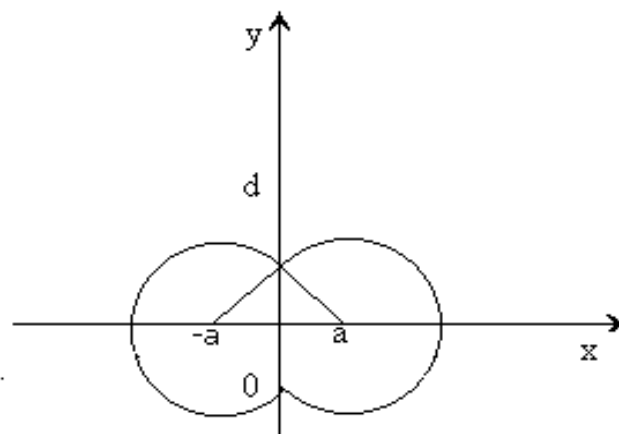
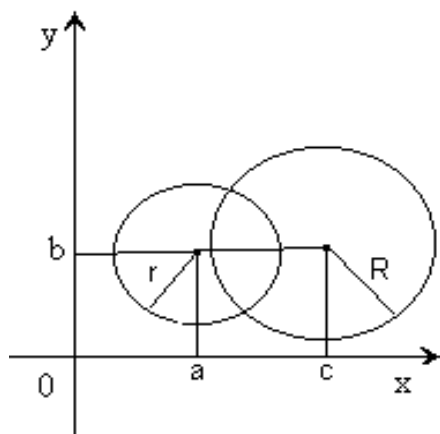
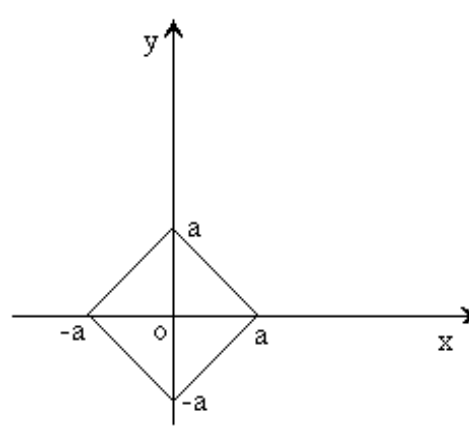
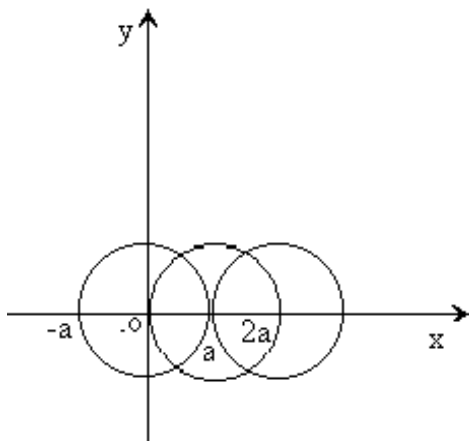
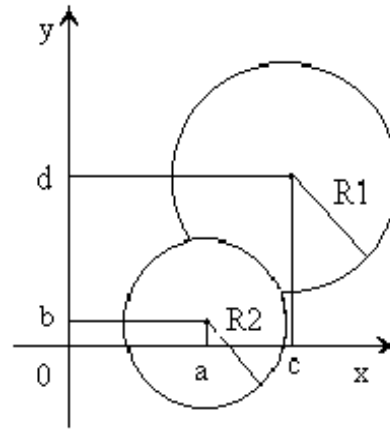
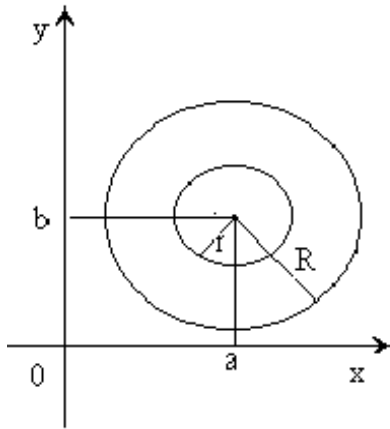
- 1.1. По номеру месяца определить количество дней в нем (в невисокосном году).
- 1.2. По номеру месяца определить время года.
- 1.3. По номеру дня недели определить количество занятий.
- 1.4. По числу лет определить является ли человек школьником, пенсионером и т.п.
- 1.5. По номеру класса определить в начальной школе, среднем звене или в старших классах обучается ученик.
- 1.6. По номеру группы определить на каком курсе учится студент.
- 1.7. По числу текущего месяца определить день недели.

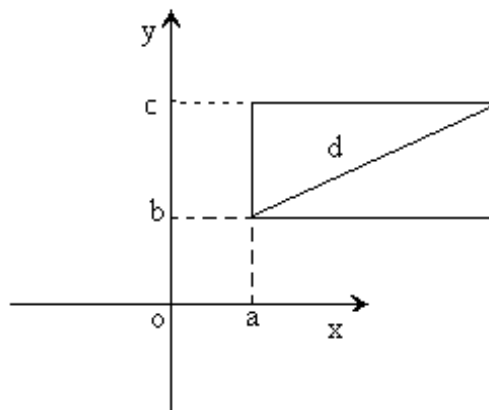
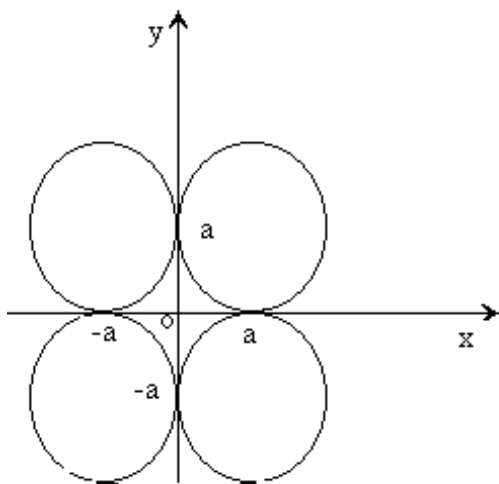
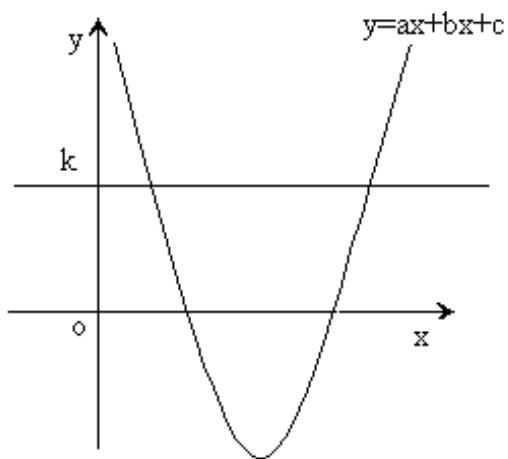
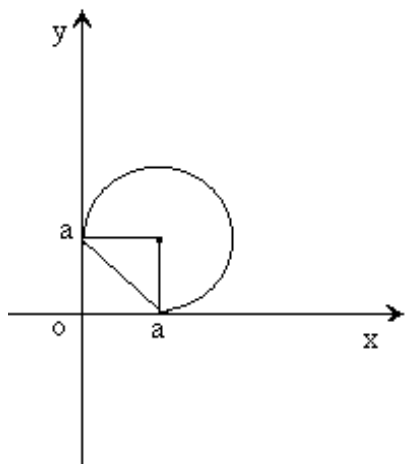
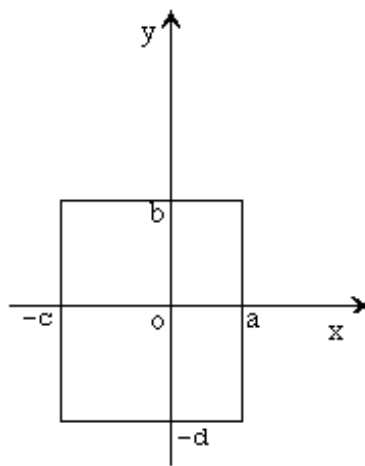
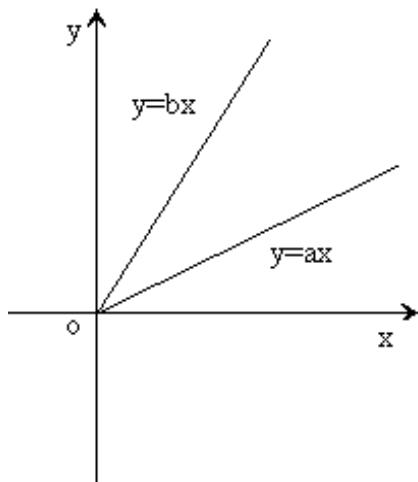
## Задание 2

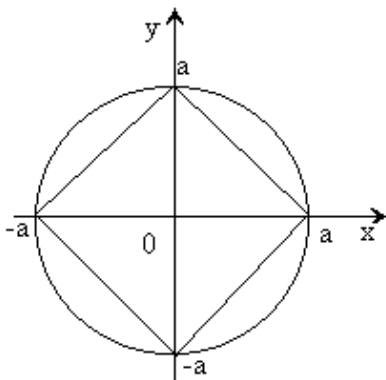
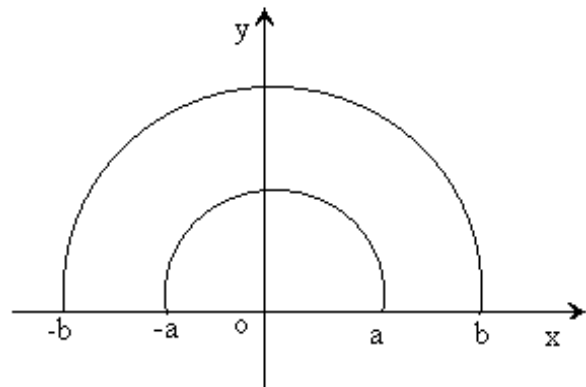
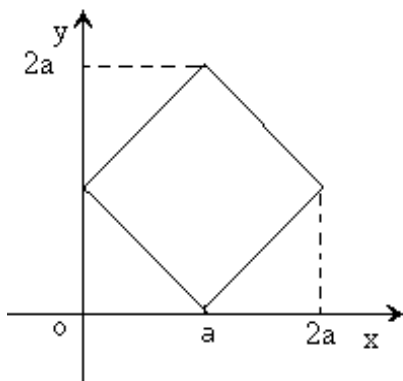
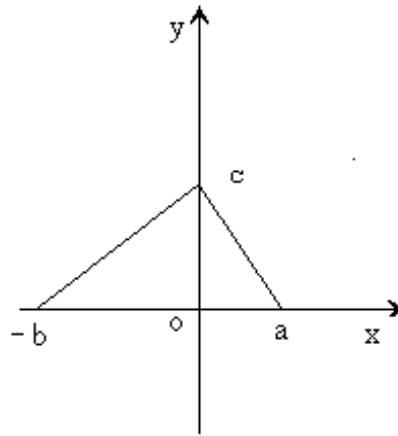
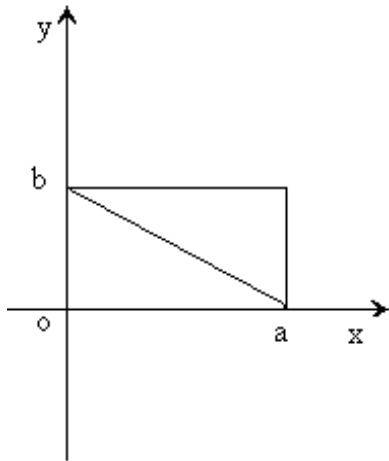
Исправьте программу из задания 1, чтобы для решения задачи использовалась команда выбора.

## Задание 3

Составьте программу, которая определяет, принадлежит ли точка с координатами  $(x, y)$  одной из указанных областей.







#### Задание 4

Составьте программу для решения одной из следующих задач.

- 4.1. Определить, существует ли треугольник с заданными сторонами  $a$ ,  $b$ ,  $c$ . Отдельно указать случай, когда треугольник является вырожденным.
- 4.2. По заданным длинам сторон определить вид треугольника (равносторонний, равнобедренный, общего вида).
- 4.3. По заданным длинам сторон определить вид треугольника (тупоугольный, остроугольный, прямоугольный).
- 4.4. Два угла заданы в градусах, минутах, секундах. Определить их разность в градусах, минутах, секундах.
- 4.5. По координатам трех точек на плоскости (в пространстве) определить, лежат ли они на одной прямой.

- 4.6. Определить вид треугольника по трем углам (остроугольный, прямоугольный, тупоугольный, вырожденный, не существует).
- 4.7. Определить количество решений у системы двух линейных уравнений с двумя переменными.
- 4.8. Можно ли вписать круг в заданную равнобедренную трапецию, если известны длины оснований и высота?

### Задание 5

Составьте программу для решения одного из уравнений или неравенств:

- |                                    |                      |
|------------------------------------|----------------------|
| 5.1. $AX^2+BX+C=0$ ( $A \neq 0$ ); | 5.5. $AX^2+BX=0$ ;   |
| 5.2. $AX+B>0$ ;                    | 5.6. $AX^2 \leq B$ ; |
| 5.3. $AX+B<0$ ;                    | 5.7. $AX^3+BX=0$ ;   |
| 5.4. $AX^2+B=0$ ;                  | 5.8. $AX^3+BX^2=0$ . |

### Лабораторная работа №4

#### Команды ветвления и повторения на языке Паскаль

Цель работы: Повторить правила синтаксиса команд ветвления и выбора. Ознакомиться с видами команды повторения, научиться составлять программы с использованием этих команд на языке Паскаль.

Замечание: перед работой должна быть выполнена контролирующая программа по теме "Команда повторения с параметром (FOR)".

#### Задание 1

Составьте программу для решения одной из следующих задач:

- 1.1. Для данного натурального числа проверить, делится ли оно на числа: 2, 3, 5, 6, 9.
- 1.2. По введенному номеру группы выдать сообщение: на каком факультете и на каком курсе учится студент.
- 1.3. Пройдет ли кирпич со сторонами  $a$ ,  $b$  и  $c$  сквозь прямоугольное отверстие со сторонами  $r$  и  $s$ ? Стороны отверстия должны быть параллельны граням кирпича.
- 1.4. Может ли шар радиуса  $r$  пройти через ромбообразное отверстие с диагоналями  $p$  и  $q$ ?
- 1.5. Можно ли коробку размером  $a \times b \times c$  упаковать в посылку размером  $r \times s \times t$ ? «Углом» укладывать нельзя.
- 1.6. Можно ли на прямоугольном участке застройки размером  $a \times b$  разместить два дома размерами  $r \times q$  и  $r \times s$  метров? Дома можно располагать только параллельно сторонам участка.

#### Задание 2

Составьте программу для решения одной из следующих задач. Программу запишите на диск под своей фамилией.

- 2.1. Решить невырожденное (т.е.  $a \neq 0$ ) биквадратное уравнение  $ax^4+bx^2+c=0$ .
- 2.2. Решить невырожденное (т.е.  $a \neq 0$ ) квадратное неравенство  $ax^2+bx+c>0$ .
- 2.3. Дано натуральное число ( $n \leq 100$ ), определяющее возраст человека (в годах). Дать для этого числа наименования "год", "года", "лет".
- 2.4. Дано время (часы, минуты, секунды)-три натуральных числа. Определить время через 10 секунд.
- 2.5. Определить дату следующего дня. Например:

31.12.1985

01.01.1986

29.04.1985

30.04.1985

- 2.6. Определить, каким днем недели является дата, заданная в виде число, месяц (в текущем году).
- 2.7. Определить число полных лет на текущий момент по введенной с клавиатуры дате.
- 2.8. По введенной дате рождения определить, является ли на сегодняшний день совершеннолетним пользователь программы.
- 2.9. Даны целые числа  $m, n$  ( $0 < m \leq 12, 0 \leq n < 60$ ), указывающие момент времени: "  $m$  часов,  $n$  минут". Определить наименьшее время (число полных минут), которое должно пройти до того момента, когда часовая и минутная стрелки на циферблате:
  - 1) совпадут;
  - 2) расположатся перпендикулярно друг другу.
- 2.10. Определить число полных лет, месяцев и дней на текущий момент по введенной с клавиатуры дате.

### **Задание 3**

Составьте программу, которая рисует во всю высоту экрана один из ваших инициалов, используя введенный символ (букву согласуйте с преподавателем).

### **Задание 4**

Вызовите с диска программу, составленную Вами в задании 2, и добавьте в нее защиту от ввода некорректных данных, т.е. при вводе данных организуйте цикл до тех пор, пока не будут выполнены ограничения, накладываемые на значения аргументов.

## **Лабораторная работа №5**

### **Циклические вычисления**

Цель работы: Ознакомиться с командой повторения, научиться составлять программы с использованием команды повторения.

Замечание: перед работой должна быть выполнена контролирующая программа по теме "Команда повторения WHILE".

### **Задание 1**

Составьте программу для решения одной из следующих задач.

- 1.1. Вычислить  $n!$ .
- 1.2. Вычислить произведение  $(a-n)(a-2n)\dots(a-kn)$ .
- 1.3. Вычислить  $\frac{(-1)^n}{n!}$
- 1.4. Вычислить сумму квадратов чисел от 1 до  $n$ .
- 1.5. Вычислить  $(2n+1)!!$ .
- 1.6. Вычислить  $(2n)!!$

### **Задание 2**

Составьте программу для решения одной из следующих задач.

- 2.1. Предприниматель, начав дело, взял кредит размером  $k$  рублей под  $p$  процентов годовых и вложив его в свое дело. По прогнозам его дело должно давать прибыль  $r$  рублей в год. Сможет ли он накопить сумму, достаточную для погашения кредита, и если да, то через сколько лет?



- 2.2. Для каждого посетителя парикмахерской (с одним мастером) известны следующие величины:  $t$  – момент его прихода и  $p$  – продолжительность его обслуживания. Сколько клиентов обслужит мастер за смену продолжительностью  $T$ ?
- 2.3. Известно время начала и окончания работы некоторого пригородного автобусного маршрута с одним автобусом на линии, а также протяженность маршрута в минутах (в один конец) и время отдыха на конечных остановках. Составить суточное расписание этого маршрута (моменты отправления с конечных пунктов) без учета времени на обед и пересменку.
- 2.4. В учебном заведении задается начало учебного дня, продолжительность пары, продолжительность обычного и большого перерывов (и место большого перерыва в расписании). Получить расписание звонков до седьмой пары.
- 2.5. Леспромхоз ведет заготовку древесины. Первоначальный объем ее на территории леспромхоза составлял  $p$  кубометров. Годовой план заготовки –  $t$  кубометров. Через сколько лет в бывшем лесу будут расти одни опята?
- 2.6. У гусей и кроликов вместе  $2n$  лап. Сколько может быть гусей и кроликов (вывести все возможные сочетания)?

### Задание 3

Составить программу решения одной из следующих задач.

- 3.1. Даны положительные действительные числа  $a, x, \varepsilon$  (в программе дать имя  $\text{eps}$ ). В последовательности  $y_0, y_1, y_2, \dots$ , образованной по закону:

$$y_0 = a; \dots$$

$$y_i = \frac{y_{i-1} + \frac{x}{y_{i-1}}}{2}, \text{ при } i=1, 2, \dots$$

найти первый член  $y_n$ , для которого выполнено неравенство  $|y_n - y_{n-1}| < \varepsilon$ . Сравнить с результатом использования встроенной функции  $y = \sqrt{x}$

- 3.2. Пусть  $x_0 = 1; \dots$

$$x_k = \frac{2 - x_{k-1}^3}{5}, k=1, 2, 3, \dots$$

Найти первый член  $x_n$ , для которого выполнено неравенство  $|x_n - x_{n-1}| < \varepsilon$ .

- 3.3. Вычислить  $x = \sqrt[3]{a}$  для заданного значения  $a$ , используя рекуррентное соотношение:

$$x_{n+1} = \frac{1}{3} \left( x_n + 2 \sqrt{\frac{a}{x_n}} \right); x_0 = a$$

Процесс вычислений выполнять до тех пор пока не будет выполнено неравенство  $|x_{n+1} - x_n| < \varepsilon$ .

- 3.4. Для заданных чисел  $m$  и  $n$  вычислить число сочетаний  $C_m^n$  непосредственно:

$$C_m^n = \frac{m!}{n!(m-n)!}$$

и по рекуррентной формуле:  $C_m^n = \frac{m-n+1}{n} C_m^{n-1}, C_m^1 = m$

- 3.5. Дано целое число  $m > 1$ . Получить наибольшее целое  $n$ , при котором  $4^n < m$

- 3.6. Найти сумму первых  $N$  чисел Фибоначчи.

- 3.7. Найти  $N$ -ое число Фибоначчи.

### Задание 4

Составьте программу для решения одной из следующих задач.

- 4.1. Определить, является ли число  $n$  простым.

- 4.2. Найти сумму делителей числа  $n$ .

- 4.3. Определить, является ли число  $n$  совершенным.
- 4.4. Найти НОК двух натуральных чисел.
- 4.5. Найти все общие делители чисел  $n$ ,  $m$ .
- 4.6. Найти все общие кратные чисел  $n$  и  $m$ , меньшие  $mn$ .
- 4.7. Даны натуральные числа  $n$ ,  $m$ . Сократите дробь  $m/n$ , то есть найдите такие натуральные  $p$  и  $q$ , не имеющие общих делителей, что  $m/n = p/q$ .
- 4.8. Дано натуральное число  $n$ . Получить все такие натуральные  $q$ , что  $n$  делится на  $q$  в квадрате и не делится на третью степень числа  $q$ .
- 4.9. Дано натуральное число  $n$ . Получить все натуральные числа, меньшие  $n$  и взаимно простые с ним.
- 4.10. Назовем шестизначное число счастливым, если сумма первых трех его цифр равна сумме последних трех. Подсчитать количество счастливых шестизначных чисел, у которых сумма первых трех цифр равна 13.
- 4.11. Подсчитать количество "счастливых" шестизначных билетов.

### Задание 5

Составьте программу, которая разбивает число  $n$  на цифры и печатает их в столбик. Дополните программу для решения одной из следующих задач.

- 5.1. Сколько цифр в числе  $N$ ?
- 5.2. Чему равна сумма его цифр?
- 5.3. Получить сумму  $m$  последних цифр числа  $n$ .
- 5.4. Выяснить, входит ли цифра 3 в запись числа  $n$ .
- 5.5. Поменять порядок цифр числа  $n$  на обратный.
- 5.6. Переставить первую и последнюю цифры числа  $n$ .
- 5.7. Приписать по единице в начало и конец записи числа  $n$ .
- 5.8. Дано натуральное число  $n$ . Выбросить из записи числа  $n$  цифры 0 и 5, оставив прежним порядок остальных цифр. Например, из числа 59015509 должно получиться 919.

**Проследите исполнение программы в пошаговом режиме(F7), создав в Watch-окне (Ctrl+F7) список промежуточных величин задачи.**

## Лабораторная работа №6

### Циклы

Цель работы: Закрепить умения составлять алгоритмы с использованием команды повторения в различных видах.

### Задание 1

Составьте программу для решения одной из следующих задач. Вычислить.

- 1.1.  $\frac{1}{a^2} + \frac{1}{a^4} + \dots + \frac{1}{a^{2^n}}$ .
- 1.2.  $a + a(a+1) + a(a+1)(a+2) + \dots + a(a+1)\dots(a+n-1)$ .
- 1.3.  $\frac{1}{a} + \frac{1}{a(a+1)} + \dots + \frac{1}{a(a+1)\dots(a+n)}$ .
- 1.4.  $\frac{1}{\sin 1} + \frac{1}{\sin 1 + \sin 2} + \dots + \frac{1}{\sin 1 + \dots + \sin(n)}$ .
- 1.5.  $1! + 2! + \dots + n!$ .
- 1.6.  $\sin(x) + \sin(\sin(x)) + \dots + \sin(\dots \sin(x))$  ( $n$  слагаемых).

- 1.7.  $\cos(x)+\cos(\cos(x))+\dots +\cos(\dots\cos(x))$  (n слагаемых).  
 1.8.  $1*2+2*3*4+\dots+n*(n+1)\dots(2n)$ .

### Задание 2

Не используя стандартные функции (за исключением модуля), вычислить с точностью  $\varepsilon > 0$  (считать, что требуемая точность достигнута, если модуль очередного слагаемого меньше  $\varepsilon$ ).

$$2.1. y=e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!}$$

$$2.5. y=\text{arctg}(x) = \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n+1}}{2n+1}$$

$$2.2. y=\cos(x) = \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n}}{(2n)!}$$

$$2.6. y=\text{sh}(x) = \frac{e^x - e^{-x}}{2} = \sum_{n=0}^{\infty} \frac{x^{2n+1}}{(2n+1)!}$$

$$2.3. y=\sin(x) = \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n+1}}{(2n+1)!}$$

$$2.7. y=\text{ch}(x) = \frac{e^x + e^{-x}}{2} = \sum_{n=0}^{\infty} \frac{x^{2n}}{(2n)!}$$

$$2.4. y=\ln(x+1) = \sum_{n=1}^{\infty} \frac{(-1)^{n-1} x^n}{n}$$

### Задание 3

Не стирайте предыдущую программу. Дополните ее вычислением номера последнего слагаемого суммы и выведите на экран изображение знака суммы, границ суммирования, числового значения суммы.

### Задание 4

Загрузите программу PRIMER7.pas, определите, какая задача решена в каждом фрагменте, сформулируйте ее и запишите условие в тетради, перепишите указанный Вам фрагмент с помощью цикла другого вида.

### Задание 5

Составьте программу для решения одной из следующих задач.

- 5.1. Даны натуральные числа  $x, y$ . ( $x > 0, y > 1$ ). Получить целое число  $k$  (положительное, отрицательное или равное нулю), удовлетворяющее условию  $y^{k-1} \leq x < y^k$ .
- 5.2. Дано натуральное число  $n$ . Можно ли представить его в виде суммы двух квадратов натуральных чисел? Если можно, то указать все пары  $x, y$  таких натуральных чисел, что  $n = x^2 + y^2$ .
- 5.3. Дано натуральное число  $n$ . Получить все простые делители этого числа.
- 5.4. Натуральное число из  $n$  цифр является числом Армстронга, если сумма его цифр, возведенных в  $n$ -ую степень, равна самому числу (например,  $153 = 1^3 + 5^3 + 3^3$ ). Получить все числа Армстронга, состоящие из трех и четырех цифр.
- 5.5. Дано натуральное число  $n$ . Получить все пифагоровы тройки натуральных чисел, каждое из которых не превосходит  $n$ , т.е. все такие тройки натуральных чисел  $a, b, c$ , что  $a^2 + b^2 = c^2$  ( $a \leq b \leq c \leq n$ ).
- 5.6. Два натуральных числа называются дружественными, если каждое из них равно сумме всех делителей другого, кроме самого этого числа. Найти все пары дружественных чисел, лежащих в диапазоне от 200 до 300.
- 5.7. Составить программу для подсчета количества "счастливых" шестизначных билетов, при исполнении которой работает не более 50000 арифметических команд.

- 5.8. Дано натуральное число  $n$ . Получить и напечатать первые  $n$  строк треугольника Паскаля.
- 5.9. Найти все простые несократимые дроби, заключенные между 0 и 1, знаменатели которых не превышают 7 (дробь задается двумя натуральными числами - числителем и знаменателем).

## **Лабораторная работа №7**

### **Одномерные массивы**

Цель работы: Сформировать понятие одномерного массива числового типа. Овладеть умениями обращения к элементам одномерного массива, ввода и вывода массива, составления алгоритмов обработки массивов.

#### **Задание 1**

Составьте программу, которая запрашивает значение элементов одномерного числового массива  $A$ , формирует и выводит на экран в строку новый числовой массив  $B$ , в котором:

- 1.1. элементы те же, что и в  $A$ , но расположены в обратном порядке.
- 1.2. элементы равны модулю соответствующих элементов  $A$ .
- 1.3. элементы на 1 больше, чем соответствующие элементы  $A$ .
- 1.4. элементы те же, что и в  $A$ , но отрицательные заменяются на 0.

Составьте программу для решения одной из следующих задач.

- 1.5 Подсчета суммы элементов массива, меньших своего номера.
- 1.6 Подсчета суммы элементов массива, стоящих на четных местах.
- 1.7 Замены всех элементов массива соответствующими степенями числа 2.
- 1.8 Замены ненулевых элементов массива на обратные.

#### **Задание 2**

Измените предыдущую программу так, чтобы в массиве  $B$ , элемент с номером  $i$  был равен:

- 2.1. сумме первых  $i$  элементов массива  $A$ .
- 2.2. произведению первых  $i$  элементов массива  $A$ .
- 2.3. количеству нулей среди первых  $i$  элементов массива  $A$ .
- 2.4. максимальному среди первых  $i$  элементов массива  $A$ .
- 2.5. количеству положительных чисел первых  $i$  элементов массива  $A$ .
- 2.6. минимальному среди первых  $i$  элементов массива  $A$ .
- 2.7. номеру максимального элемента среди первых  $i$  элементов  $A$ .
- 2.8. номеру минимального элемента среди первых  $i$  элементов  $A$ .

После исполнения программы элементы массива  $B$  должны быть напечатаны на экране каждый под соответствующим элементом массива  $A$ .

#### **Задание 3**

Составьте программу для решения одной из следующих задач.

- 3.1. Определить, есть ли в массиве положительные числа, отрицательные числа, нули.
- 3.2. Определить, все ли числа из массива  $A$  лежат в указанном диапазоне.
- 3.3. Определить, есть в массиве числа, превосходящие сумму всех элементов массива.
- 3.4. Определить, единственный ли в массиве максимальный элемент.
- 3.5. Определить, является ли массив  $A$  упорядоченным по возрастанию.
- 3.6. Определить, является ли массив  $A$  упорядоченным по убыванию.

- 3.7. Определить номер первого нуля в массиве А или выдать сообщение, что нулей в массиве нет.
- 3.8. Определить, симметричны ли элементы в массиве относительно центрального или центральных.

#### **Задание 4**

Составьте программу для решения одной из следующих задач.

- 4.1. Удалить из массива минимальный элемент
- 4.2. Подсчитать количество рабочих в данной бригаде, работающих хуже, чем в среднем вся бригада.
- 4.3. Обменять значениями максимальный и минимальный элементы.
- 4.4. Подсчитать количество дней, имеющих наибольшую температуру за последнюю декаду марта.
- 4.5. Найти количество максимальных элементов массива, используя только один оператор цикла.
- 4.6. Вычислить произведение элементов массива А до первого отрицательного элемента (при наличии отрицательного элемента).
- 4.7. Вычислить произведение элементов массива А, находящихся между первым максимальным и первым минимальным элементами.
- 4.8. Вычислить сумму положительных элементов массива А после первого нуля (при наличии нуля).
- 4.9. Вычислить сумму отрицательных элементов (если такие есть) массива А до первого максимального элемента.

#### **Задание 5**

Составьте программу для решения одной из следующих задач.

- 5.1. Найти второй по величине элемента массива.
- 5.2. Упорядочить массив А по возрастанию.
- 5.3. Найти наибольшее количество одинаковых элементов массива А, идущих подряд.
- 5.4. Подсчитать К- количество элементов, равных первому отрицательному, используя один оператор цикла; если отрицательных нет, то  $K=-1$ .
- 5.5. Исключить нули со сдвигом элементов таблицы.
- 5.6. По двоичной записи натурального числа М (в виде массива 0 и 1) выдать двоичную запись числа  $M+1$ .
- 5.7. Определить количество разных элементов массива А.
- 5.8. Найти максимальный из отрицательных элементов массива А с четными номерами (с учетом того, что их может не быть).
- 5.9. Найти минимальный из положительных элементов массива А с четными номерами (с учетом того, что их может не быть).
- 5.10. Определить наибольшее количество одинаковых элементов в данном массиве.

### **Лабораторная работа №8**

#### **Двумерные массивы**

Цель работы: Сформировать понятие двумерного массива числового типа. Овладеть умениями обращения к элементам двумерного массива, ввода и вывода массива, составления алгоритмов обработки массивов.

#### **Задание 1**

Составьте программу, которая запрашивает размеры двумерного массива, значения его элементов и выводит на очищенный экран получившийся массив.

### **Задание 2**

Дополните предыдущую программу так, чтобы вычислялось и выводилось на экран значение:

- 2.1 суммы всех элементов массива.
- 2.2 количества 0 в массиве.
- 2.3 произведения ненулевых элементов последней строки.
- 2.4 произведения положительных элементов последнего столбца.
- 2.5 максимального элемента массива.
- 2.6 минимального элемента массива.
- 2.7 суммы положительных элементов массива.
- 2.8 количества отрицательных элементов в массиве.

### **Задание 3**

Измените предыдущую программу так, чтобы вычислялось и выводилось на экран значение:

- 3.1. суммы элементов выше главной диагонали.
- 3.2. количества 0 ниже главной диагонали.
- 3.3. произведения ненулевых элементов ниже побочной диагонали
- 3.4. произведения положительных элементов выше побочной диагонали.
- 3.5. максимального элемента выше побочной диагонали.
- 3.6. минимального элемента ниже побочной диагонали
- 3.7. суммы положительных элементов выше главной диагонали.
- 3.8. количества отрицательных элементов ниже главной диагонали.

**Замечание:** массив должен быть квадратным, т.е. число строк равно числу столбцов.

### **Задание 4**

Составьте программу для нахождения количества:

- 4.1 строк, где все элементы одинаковы.
- 4.2 столбцов, где все элементы положительны.
- 4.3 строк, где знаки элементов чередуются.
- 4.4 столбцов, совпадающих с первым.
- 4.5 нулевых строк.
- 4.6 столбцов, где все элементы одинаковы.
- 4.7 строк, совпадающих с последней.
- 4.8 нулевых столбцов.

### **Задание 5**

Составьте программу для решения одной из следующих задач.

- 5.1. Определить, есть ли в массиве одинаковые строки.
- 5.2. Определить сколько в массиве столбцов, где все элементы различны.
- 5.3. Определить, верно ли, что все строки упорядочены по убыванию.
- 5.4. Определить, верно ли, что максимальные элементы всех строк находятся в разных столбцах.
- 5.5. Определить, верно ли, что минимальные элементы всех столбцов находятся в разных строках.

- 5.6. Определить, является ли массив магическим квадратом, т.е. совпадает ли в нем сумма каждой строки, каждого столбца и двух диагоналей.
- 5.7. Найти минимальное число  $K$ , для которого хотя бы в одной строке все элементы меньше или равны  $K$ .
- 5.8. Найти максимальное число  $K$ , для которого в каждой строке есть хотя бы один элемент, больший или равный  $K$ .

### **Лабораторная работа №9**

#### **Работа со строковыми величинами**

Цель работы: Сформировать понятие величин полусоставного типа. Научиться составлять алгоритмы обработки строковых переменных.

#### **Задание 1**

Составьте программу для решения одной из следующих задач.

- 1.1. Определить, какое из двух слов длиннее и на сколько.
- 1.2. Определить, является ли какое-нибудь из двух слов частью другого.
- 1.3. Определить, есть ли в записи квадрата данного числа цифра 1.
- 1.4. Поменять в слове первую и последнюю буквы.
- 1.5. Если в слове нечетное число букв, то удвоить среднюю.
- 1.6. По последнему символу определить тип предложения (повествовательное, вопросительное, восклицательное).
- 1.7. Определить, является ли данный символ латинской буквой.
- 1.8. Удалить из слова среднюю букву (или две средних).

#### **Задание 2**

Составьте программу для решения одной из следующих задач.

- 2.1. Заменить в арифметическом выражении знаки "+" на знаки "-", а знаки "-" на знаки "+".
- 2.2. Удалить все буквы "я" в данном слове.
- 2.3. Удвоить все четные буквы слова.
- 2.4. Удалить все предлоги "на" в данном предложении.
- 2.5. Вставить после каждой буквы данного слова букву "о".
- 2.6. Удалить лишние пробелы в данном предложении.
- 2.7. Удвоить каждую букву данного слова.
- 2.8. Заменить каждую точку многоточием (т.е. тремя точками).

Программу запишите на диск.

#### **Задание 3**

Вызовите программу, составленную Вами при выполнении задания 2, и переделайте ее так, чтобы в ней использовалась только одна строковая переменная.

#### **Задание 4**

Составьте и исполните программу для решения одной из задач.

- 4.1 Заменить в тексте все маленькие латинские буквы на большие.
- 4.2 Заменить в тексте все большие латинские буквы на маленькие.

#### **Задание 5**

Составьте и исполните программу для решения следующей задачи.

- 5.1. Удвоить все согласные буквы.

- 5.2. Удалить из данного слова все согласные буквы.
- 5.3. Проверить, имеются ли в данном слове одинаковые буквы.
- 5.4. Оставить в данном слове из каждого набора одинаковых букв, идущих подряд, только одну букву.
- 5.5. Определить возможность составления одного данного слова из букв другого данного слова с учетом кратности.
- 5.6. Определить возможность составления одного данного слова из букв другого данного слова без учета кратности.
- 5.7. Для подсчета количества слов в предложении, учитывая что между словами может быть несколько пробелов.
- 5.8. Выяснить, можно ли из символов заданного слова составить слово "море".

### **Лабораторная работа №10**

#### **Обработка литерных величин на языке Турбопascal**

Цель работы: Закрепить умение составлять алгоритмы обработки строковых переменных, научиться составлять вспомогательные алгоритмы.

Загрузите программу lab10.pas, проверьте ее исполнение.

#### **Задание 1**

Добавить программу так, чтобы вводился массив слов, очищался экран и печатался введенный массив в столбик одним из следующих способов:

- 1.1 с общим концом в конце соответствующей строки.
- 1.2 с общим концом в середине соответствующей строки.
- 1.3 с общей серединой в середине соответствующей строки.
- 1.4 по два в строку с выравниванием по концам слов.
- 1.5 по два в строку с выравниванием по серединам слов.

#### **Задание 2**

Дополнить предыдущую программу так, чтобы отдельно печатались:

- 2.1 слова, начинающиеся с буквы "а".
- 2.2 слова, в которых есть буква "а".
- 2.3 слова, которые короче 5 букв.
- 2.4 слова, которые длиннее 8 букв.
- 2.5 слова, которые заканчиваются на "о".
- 2.6 слова, у которых первый и последний символы совпадают.
- 2.7 слова с нечетным количеством букв.
- 2.8 слова, в которые входит буква "в".

#### **Задание 3**

Изменить предыдущую программу так, чтобы:

- 3.1 перевернуть каждое слово.
- 3.2 подсчитать количество шипящих букв в каждом слове.
- 3.3 определить количество слов, состоящих из одного, двух, трех и более слогов.
- 3.4 найти длину самого короткого слова.
- 3.5 печатались слова, из букв которых можно составить слово 'море'.
- 3.6 для каждого из слов указывалось, сколько раз оно встречается (желательно без повторов).
- 3.7 вычислялось количество слогов в каждом слове.



3.8 находилось количество слов-палиндромов.

#### **Задание 4**

Составить процедуру, которая:

- 4.1 в данном слове считает количество букв "а".
- 4.2 удаляет в данном слове четные буквы.
- 4.3 считает, сколько раз данное слово встречается в массиве.
- 4.4 заменяет в данном слове все буквы "n" на "m".
- 4.5 в данном слове считает количество букв "о".
- 4.6 удаляет в данном слове букву "а".
- 4.7 вставляет в данном слове букву "о" после каждой буквы.
- 4.8 удваивает все буквы в данном слове и опробовать ее для каждого слова из введенного массива.

#### **Задание 5**

Составить программу, которая:

- 5.1 проверяет, правильна ли запись числа римскими цифрами.
- 5.2 находит все слова, содержащие наибольшее количество гласных букв.
- 5.3 упорядочивает список фамилий.
- 5.4 разбивает введенные слова на слоги.
- 5.5 находит для каждого слова наиболее часто встречающуюся букву.
- 5.6 переводит число из римской записи в арабскую.
- 5.7 Дан текст-словарь из N слов. Вводятся слова, в которых может быть допущены одна из следующих ошибок:
  - пропущена одна буква;
  - вставлена лишняя буква;
  - искажена одна буква;

Найти в словаре все слова, из которых могло получиться данное слово в результате не более чем одной ошибки.

#### **Дополнительные задания**

1. Определить, сколько разных символов в каждом слове.  
Вывести информацию на экран.
2. Для каждого из слов указать, сколько раз оно встречается среди слов заданного текста.
3. Найти все слова, содержащие наибольшее количество гласных букв.

### **Лабораторная работа №11**

#### **Обработка двумерных массивов на языке Турбопascal**

Цель работы: Закрепить умение составлять вспомогательные процедуры и функции на основе составления алгоритмов обработки двумерных массивов, сформировать умение работать с библиотекой вспомогательных алгоритмов.

Загрузите программу lab11.pas, проверьте ее исполнение, замените вызов процедуры ввода на вызов присваивания и снова исполните программу.

#### **Задание 1**

Изменив текст основной программы, решите одну из следующих задач:

Вычислить количество строк, где:

- 1.1 среднее арифметическое меньше нуля.
- 1.2 все элементы меньше нуля.
- 1.3 только один элемент не равен нулю.
- 1.4 есть элементы разных знаков.
- 1.5 все элементы упорядочены по возрастанию.
- 1.6 сумма отрицательных больше суммы положительных (те и другие присутствуют в строке обязательно).

### **Задание 2**

- 2.1 Составьте и опробуйте процедуру печати одной строки массива.
- 2.2 Используя процедуры смены знака и нахождения минимума, составьте и опробуйте процедуру нахождения минимума в строке с указанным номером.

### **Задание 3**

Решите одну из задач так, чтобы в тексте основной программы не было бы конструкций нахождения минимума (максимума), суммы и т.п.

Выведите на экран строку:

- 3.1 с минимальной суммой модулей.
- 3.2 с максимальной суммой модулей.
- 3.3 с минимальной суммой.
- 3.4 с максимальной суммой.
- 3.5 содержащую максимальный по модулю элемент.
- 3.6 содержащую минимальный из максимальных элементов строк.
- 3.7 содержащую максимальный из минимальных элементов строк.
- 3.8 в которой все элементы равны.

### **Задание 4**

Составить и опробовать процедуру (или несколько), которая:

- 4.1 заменяет все элементы массива на числа с обратным порядком цифр.
- 4.2 заменяет все элементы массива на числа, в которых удалена десятичная точка.
- 4.3 заменяет все элементы массива на числа, у которых целая и дробная части поменялись местами.
- 4.4 каждый элемент массива заменяет на число его цифр.
- 4.5 заменяет все элементы массива на числа, из которых удалена цифра 5.
- 4.6 заменяет все элементы массива на число цифр исходного элемента.

Замечания: 1) исходный и конечный массивы должны быть выведены на экран;

2) при создании процедуры можно пользоваться любыми ранее изученными алгоритмами.

### **Задание 5**

Составить рекурсивный алгоритм нахождения:

- 5.1 суммы элементов одномерного массива А длины N.
- 5.2 максимального элемента.
- 5.3 минимального элемента.
- 5.4 номера максимального элемента.
- 5.5 номера минимального элемента.
- 5.6 количества отрицательных элементов.
- 5.7 номера последнего нуля.

5.8 номера последнего отрицательного элемента.

## **Лабораторная работа №12**

### **Работа с одномерными и двумерными массивами**

Цель работы: Закрепить умение составлять алгоритмы обработки одномерных и двумерных массивов с использованием вспомогательных процедур и функций.

#### **Задание 1**

Составьте программу для решения одной из следующих задач:

- 1.1. Имеются результаты  $n$  ежедневных измерений количества выпавших осадков. За какую из недель, считая с начала периода измерений, выпало наибольшее количество осадков?
- 1.2. Дан массив фамилий студентов 1-го курса и массив их результатов в беге на 100 метров. Составьте команду из четырех лучших бегунов для участия в эстафете 4\*100.
- 1.3. На факультете были проведены соревнования по подтягиванию и все результаты занесены в массив в порядке выступления студентов. Составьте программу, которая распечатывает места, занятые участниками соревнований. Если результаты одинаковые, то лучшее место занимает тот, кто раньше выступал.
- 1.4. Пусть таблица выигрышей лотереи представлена в виде двух массивов: первый - это выигравшие номера, второй - выпавшие выигрыши на эти номера. Ваша группа купила несколько билетов лотереи, номера которых занесены в третий массив. Для каждого номера определите выигрыш, если он есть и распечатайте результат в виде хорошо оформленной таблицы.
- 1.5. После I семестра все итоги по информатике были занесены в две базы, каждая из которых содержит сведения о фамилии студента и о количестве набранных им баллов. В одну из них занесены студенты первой подгруппы вашей группы, в другую - второй. Обе базы упорядочены по убыванию количества набранных баллов. Распечатайте вместе обе подгруппы студентов так, чтобы все результаты были упорядочены по убыванию.
- 1.6. Студенты одной из групп решили съездить на неделю в Париж. Но оказалось, что бесплатных путевок, которые им выделил деканат, на одну меньше, чем студентов в группе. На собрании было решено не брать последнего ученика в списке группы, у которого удовлетворительная оценка по информатике. Составьте программу, которая печатает список тех, кто поедет в Париж на халяву.
- 1.7. Есть база, содержащая итоговые оценки по информатике, полученные каждым студентом группы за I полугодие. Студенты работали за ЭВМ парами, и данные в базе записаны так, что ученики из одной пары стоят рядом. Учитель зачислил учеников, получивших итоговую оценку 5, в отдельную группу для более углубленного изучения информатики. В результате некоторые ученики потеряли свои пары. Распечатайте список фамилий отличников по информатике и список новых пар, который бы вы составили на месте учителя, сохранив не распавшиеся пары.
- 1.8. Дан массив фамилий абитуриентов и массив результатов экзамена по математике. Оставьте в массивах только тех, кто будет допущен к следующему экзамену.

#### **Задание 2**

Составьте программу для решения одной из следующих задач:

- 1.1. Для двумерного массива, содержащего сведения о расписании занятий в вашей группе (<день недели>, <номер пары>, <назв. предмета>), напечатайте, какие предметы изучают студенты (предметы не должны повторяться).
- 1.2. В группе решили участвовать в конкурсе эрудитов. Оказалось, что число студентов на одного больше, чем требуется в команде. Было решено отстранить от участия в конкурсе первого студента в списке, у которого сумма всех оценок за сессию наименьшая. Составьте программу, которая печатает получившийся состав команды.
- 1.3. Есть массивы, содержащие все оценки по информатике, полученные каждым студентом группы за I полугодие. В другом массиве записаны критерии итоговых оценок, определяемых по сумме набранных каждым студентом баллов. Составьте программу, которая печатает список студентов вместе с их итоговыми оценками.
- 1.4. В соревнованиях по прыжкам в длину каждому участнику дается три попытки, и все результаты заносятся в массив. При подведении итогов результатом считается лучшая из попыток. Напечатайте фамилии участников и их результаты в порядке занятых ими мест.
- 1.5. В некоторых видах спортивных состязаний выступление каждого спортсмена независимо оценивается несколькими судьями, затем из всей совокупности оценок удаляется наиболее высокая и наиболее низкая, а для оставшихся оценок вычисляется среднее арифметическое, которое и идет в зачет спортсмену. Если наиболее высокую оценку выставили несколько судей, то из совокупности оценок удаляется только одна такая оценка, аналогично поступают с наиболее низкими оценками. Определите оценки, которые пойдут в зачет каждому спортсмену.
- 1.6. В некоторых видах спортивных состязаний выступление каждого спортсмена независимо оценивается несколькими судьями, расставляющих всех спортсменов по местам. Затем для каждого спортсмена вычисляется сумма набранных им мест у каждого из судей, которая определяет итоговое место спортсмена. Составьте программу, которая вычисляет для каждого спортсмена сумму его мест (и распределяющую спортсменов по занятым ими местам).
- 1.7. Известны результаты вступительных экзаменов на определенную специальность. Определить проходной балл, если из числа абитуриентов необходимо отобрать  $k$  человек.
- 1.8. Клеточное поле размером  $m \times n$  есть результат игры в крестики-нолики. Определить результат игры: выигрыш «крестиков», выигрыш «ноликов», ничья. Считается, что «крестики» выиграли, если найдется по горизонтали, вертикали или диагонали цепочка, состоящая подряд из 5 крестиков.

### **Лабораторная работа №13**

#### **Многочлены**

Цель работы: Закрепить умение работать с библиотекой вспомогательных алгоритмов основе составления алгоритмов обработки многочленов.

Загрузите программу `polinom.pas`, подберите данные, удобные для тестирования и исполните ее.

#### **Задание 1**

Составьте программу для решения одной из предложенных задач.

1.1. Даны действительные числа  $a_0, a_1, a_2, \dots, a_6$ , являющиеся коэффициентами многочлена  $r(x)$  шестой степени.

Получить для  $x=1, 3, 4$  значение  $r(x+1)-r(x)$ .

1.2. Дан многочлен  $P(x)$  степени  $n$ . Получить многочлен  $P^2(x)$ .

1.3. Дан многочлен  $P(x)$  степени  $n$ . Вычислить  $P'(1), P'(2), P(3)$ .

1.4. Найти сумму многочленов  $f(x), g(x)$ .

1.5. Вычислите значение многочлена  $f(x)$ :

$$f(x)=(5, -7, 8, -3, 7), x_0=3,$$

$$f(x)=(2, 2, -3, 4, -6, 5), x_0=-0.5.$$

## Задание 2

Составьте программу для решения одной из предложенных задач.

2.1. Даны действительное число  $a$ , многочлен  $P(x)$  степени  $n$ . Получить:

многочлен  $(x-a)P(x)$ ,

многочлен  $(x^2+2ax+3)P(x)$ ,

многочлен  $(x^2+a^2)P(x)$ .

2.2. Пользуясь схемой Горнера разделите с остатком многочлен  $f(x)$  на многочлен  $(x-x_0)$  и вычислите  $f(x_0)$ .

а)  $f(x)=(1, -3, 6, -10, 16)$ .  $x_0=4$ .

б)  $f(x)=(1, 2, -3, -4, 1)$ .  $x_0=-1$ .

в)  $f(x)=(1, 5, -6, 8, -3)$ .  $x_0=2$ .

г)  $f(x)=(2, 7, -8, 3, -5)$ .  $x_0=-2$ .

д)  $f(x)=(3, -2, 6, -8, 11)$ ;  $x_0=-1.5$ .

2.3. Найдите нормированный многочлен четвертой степени с действительными коэффициентами, имеющий двукратный корень 2 и простые корни 3 и  $(-1)$ .

2.4. Разделите многочлен  $f(x)$  на многочлен  $g(x)$  с остатком.

а)  $f(x)=(4, -2, 1, 1, 2)$ ;  $g(x)=(2, -1, -1, 1)$ .

б)  $f(x)=(2, -3, 4, -5, 6)$ ;  $g(x)=(1, -3, 1)$ .

в)  $f(x)=(1, -3, -1, -1)$ ;  $g(x)=(3, -2, 1)$ .

## Задание 3

Составьте программу для решения одной из предложенных задач:

3.1. Даны действительные числа  $a_0, a_1, a_2, \dots, a_{12}$ , являющиеся коэффициентами многочлена  $p(x)$  степени 5. Получить  $p(1)-p^2(3)-2p(2)$ .

3.2. Даны действительные числа  $s$  и  $t$ , натуральное число  $n$ , действительные числа  $a_0, a_1, a_2, \dots, a_n$ . Среди  $a_0, a_1, a_2, \dots, a_n$  есть как отрицательные, так и неотрицательные числа. Получить значение  $P(s)+Q(t)$ , где в качестве коэффициентов многочлена  $P$  взяты отрицательные члены последовательности  $a_0, a_1, a_2, \dots, a_n$  (с сохранением порядка их следования), а в качестве коэффициентов многочлена  $Q$  — неотрицательные члены (также с сохранением порядка их следования).

3.3. Даны действительные числа  $s, t$ , многочлен  $P(X)$  степени  $n$ . Получить многочлен  $(sx^2+t)P(x)+P'(x)$ .

3.4. С помощью схемы Горнера найдите кратность корня  $x_0$  многочлена  $f(x)$ :

а)  $x_0=1$ ;  $f(x)=(1, -5, -2, 26, -31, 11)$ .

б)  $x_0=2$ ;  $f(x)=(1, -5, 7, -2, 4, -8)$ .

в)  $x_0=-2$ ;  $f(x)=(1, 7, 16, 8, -16, -16)$ .

г)  $x_0=-2$ ;  $f(x)=(1, 3, -4, 6, -5)$ .

д)  $x_0=3$ ;  $f(x)=(2, 0, -3, 6, -8, -4)$ .

е)  $x_0=-4$ ;  $f(x)=(2, 0, 1, 0, -3, 0, 4, -7)$ .

ж)  $x_0=3; f(x)=(1, -6, 10, -6, 9)$ .

з)  $x_0=-2; f(x)=(1, 6, 11, 2, -12, -8)$ .

3.5. Найти сумму коэффициентов многочлена  $f(x)$ , равного  $(2-5x+x^2)^t(3-7x+9x^2)^s$ . Значения  $s$  и  $t$  введите с клавиатуры.

#### Задание 4

Составьте программу для решения одной из предложенных задач:

5.1. Даны целые числа  $n_0, d_0, n_1, d_1, \dots, n_7, d_7, a, b$  ( $d_0d_1\dots d_7b < 0$ ). Вычислить по схеме Горнера  $\frac{n_7}{d_7} \cdot \frac{a^7}{b^7} + \frac{n_6}{d_6} \cdot \frac{a^6}{b^6} + \dots + \frac{n_0}{d_0}$ .

5.2. Даны действительные числа  $a_0, a_1, \dots, a_5$ . Получить многочлен шестой степени  $(x-a_0)(x-a_1)\dots(x-a_5)$ .

5.3. Даны действительные числа  $a_0, \dots, a_5, d_0, \dots, d_5$ . Получить многочлен шестой степени  $d_0+d_1(x-a_0)+d_2(x-a_0)(x-a_1)+\dots+d_5(x-a_0)(x-a_1)\dots(x-a_5)$ .

5.4. Последовательность многочленов  $T_0(x), T_1(x), \dots$  определяется следующим образом:

$$T_0(x)=1,$$

$$T_1(x)=x,$$

$$T_k(x)=2xT_{k-1}(x)-T_{k-2}(x) \quad (k=2, 3, \dots).$$

Получить все многочлены, начиная с  $T_2(x)$  до  $T_8(x)$ .

5.5. Последовательность многочленов  $H_0(x), H_1(x), \dots$  определяется следующим образом:

$$H_0(x)=1,$$

$$H_1(x)=x,$$

$$H_k(x)=xH_{k-1}(x)-(k-1)H_{k-2}(x) \quad (k=2, 3, \dots).$$

Получить:

а)  $H_2(x), H_4(x), H_6(x)$ .

б) Даны действительные числа  $a_0, \dots, a_6$ . Получить многочлен  $a_0H_0(x)+\dots+a_6H_6(x)$ .

в) Дано действительное число  $a$ . Вычислить  $H_0(a)+\dots+H_6(a)$ .

5.6. Последовательность многочленов  $G_0(x), G_1(x), \dots$  определяется следующим образом:

$$G_0(x)=1,$$

$$G_1(x)=x-1,$$

$$G_k(x)=(x-2k+1)G_{k-1}(x)-(k-1)^2G_{k-2}(x) \quad (k=2, 3, \dots).$$

Получить:

а)  $G_3(x), G_5(x), G_7(x)$ .

б) Даны действительные числа  $a_0, \dots, a_6$ . Получить многочлен  $a_0G_0(x)+\dots+a_6G_6(x)$ .

в) Дано действительное число  $a$ . Вычислить  $G_0(a)+\dots+G_6(a)$ .

5.7. Пользуясь схемой Горнера, найти значение многочлена  $f(x)$  и его производных при  $x=a$ .

а)  $f(x)=(4, -2, 5, -1), a=2$ .

б)  $f(x)=(3, 8, -2, 6, -5), a=3$ .

в)  $f(x)=(1, 9, 7, -2, -11, 7), a=-4$ .

5.8. Многочлен  $f(x)$  четвертой степени со старшим коэффициентом, равным 1, имеет число  $(-2)$  трехкратным корнем и при делении на  $(x+3)$  дает остаток, равный  $(-1)$ . Найдите этот многочлен.

## Задание 5

Составьте программу для решения одной из предложенных задач:

- 5.1. Даны целые числа  $f_1, f_2, f_3, \dots, f_{10}$ , являющиеся коэффициентами многочлена  $z(x)$ . Исследовать существование целочисленных корней уравнения  $z(x)=0$ .
- 5.2. Даны действительные числа  $a_0, \dots, a_n, b_0, b_1, \dots, b_n$  ( $a_0, \dots, a_n$  попарно различны). Требуется найти многочлен  $F(x)$  степени не выше  $n$ , такой, что  $F(a_i)=b_i$  ( $i=0, 1, 2, \dots, n$ ).
- 5.3. Найдите наибольший общий делитель многочленов  $f(x), g(x)$ :
- а)  $f(x)=(1, 3, -1, -4, -3), g(x)=(3, 10, 2, -3)$ .
  - б)  $f(x)=(1, 1, -3, -4, -1), g(x)=(1, 1, -1, -1)$ .
  - в)  $f(x)=(1, 2, -4, -3, 8, -5), g(x)=(1, 1, -1, 1)$ .
  - г)  $f(x)=(1, 1, 2, 1, 1), g(x)=(1, -2, 1, -2)$ .
  - д)  $f(x)=(1, 2, 2, 2, 2), g(x)=(1, 0, 3, 2)$ .
  - е)  $f(x)=(1, 6, 17, 24, 12), g(x)=(1, -2, -13, -10)$ .
  - ж)  $f(x)=(1, 1, 3, 4, 4, 2), g(x)=(1, 2, 3, 6, 6, 2)$ .
  - з)  $f(x)=(1, 6, 2, 3, 6, 1), g(x)=(1, 6, 4, 4, 6)$ .
- 5.4. Найдите наименьшее общее кратное многочленов  $f(x), g(x)$ :
- а)  $f(x)=(2, 0, 1, -3), g(x)=(1, 1, -2)$ .
  - б)  $f(x)=(1, -2, 1, 7, -12, 10), g(x)=(3, -6, 5, 2, -2)$ .
  - в)  $f(x)=(1, 0, -10, 0, 1), g(x)=(1, -4, 2, 6, 4, 2, 1)$ .
- 5.5. Даны действительные числа  $a_0, \dots, a_5$ , многочлен  $P(x)$  шестой степени. Получить действительные числа  $d_0, \dots, d_6$  такие, что  $P(x)=d_0+d_1(x-a_0)+d_2(x-a_0)(x-a_1)+\dots+d_6(x-a_0)(x-a_1)\dots(x-a_5)$ .

## Лабораторная работа №14

### Линейная комбинация векторов

Цель работы: Овладеть навыками составления алгоритмов решения геометрических задач по теме "Линейные операции над векторами", используя заданный набор процедур.

Файл LIST.1 содержит заголовки программы, функции `det2`, `det3` и следующие процедуры: `input`, `output`, `sum`, `subtract`, `multiply`, `system2`, `system3`.

#### Порядок решения задачи

1. Внимательно проанализируйте условие задачи и определите, какими процедурами и функциями из файла LIST.1 Вы воспользуетесь для данной задачи.
2. Допишите основную программу.
3. Исполните программу. Проанализируйте ответ, результат покажите преподавателю.

#### Пример решения задачи

Задача : Найти сумму  $k$  векторов размерности  $n$ .

Анализ : для решения данной задачи воспользуемся процедурами `input,output,sum`.

Основная программа имеет вид :

```
begin
write('Введите размерность вектора ');
readln(n); write('Введите число векторов ');readln(k);
```

```

{----- Первоначальное обнуление вектора суммы b -----}
for i:=1 to n do s[i]:=0;
{--- Ввод координат очередного вектора и добавление его к вектору суммы ---}
for i:=1 to k do
begin
  writeln('Введите координаты ',i,'-того вектора ');
  input(n,a);      sum(n,a,s,s);      end;
writeln('Координаты вектора суммы ');output(n,s);
end.

```

## Задания к лабораторной работе

### Задание 1

Исполните программу вычисления суммы  $k$  векторов.

### Задание 2

Переделайте предыдущую программу так, чтобы вычислялась линейная комбинация  $k$  векторов.

### Задание 3

Составьте программу, определяющую:

- а) коллинеарны ли два вектора плоскости;
- б) компланарны ли три вектора.

### Задание 4

Составьте программу, определяющую составляющую  $q$  вектора  $p$  на вектор  $a$  при косом проектировании в направлении вектора  $b$

### Задание 5

Составьте программу для решения одной из следующих задач:

а) Найти составляющую  $q$  вектора  $p$  на плоскость, определяемую векторами  $a$  и  $b$ , при косом проектировании в направлении вектора  $c$  (в случае компланарности ввод векторов  $a, b, c$  повторяется).

б) Произвольная пирамида  $SABCD$ , в основании которой лежит параллелограмм  $ABCD$ , задана векторами  $SA=a$ ,  $SB=b$ ,  $SC=c$ . Вычислить вектор  $MN$ , где  $M$  - середина ребра  $SD$ ,  $N$  - центроид треугольника  $SAC$ .

в) Произвольная пирамида  $SABCD$ , в основании которой лежит трапеция  $ABCD$  ( $CD=L \cdot BA$ ), задана векторами  $SA=a$ ,  $SB=b$ ,  $SC=c$ . Вычислить вектор  $MN$ , где  $M$  - середина ребра  $SD$ ,  $N$  - центроид треугольника  $SAC$ .

г) Наклонная треугольная призма  $ABCA_1B_1C_1$  построена на векторах  $AB=a$ ,  $AC=b$ ,  $AA_1=c$ . Найдите вектор  $MN$ , если  $M$  - центр параллелограмма  $BCC_1B_1$ , а  $N$  - центроид треугольника  $A_1B_1C_1$ .

д) Параллелепипед  $ABCD A_1B_1C_1D_1$  построен на векторах  $AB=a$ ,  $AD=b$ ,  $AA_1=c$ . Найдите вектор  $MN$ , если  $M$  - середина ребра  $CC_1$ , а  $N$  - центроид треугольника  $CB_1D_1$ .



е) Усеченная четырехугольная пирамида  $ABCD A_1 B_1 C_1 D_1$ , в основании которой лежит параллелограмм  $A_1 B_1 C_1 D_1$ , а отношение сходственных ребер равно  $L$ , задана векторами  $AB=a$ ,  $AD=b$ ,  $AA_1=c$ . Найдите вектор  $MN$ , если  $M$  - середина ребра  $CC_1$ , а  $N$  - центр тяжести треугольника  $AB_1 D_1$ .

ж) Усеченная четырехугольная пирамида  $ABCD A_1 B_1 C_1 D_1$ , в основании которой лежит трапеция  $A_1 B_1 C_1 D_1$  ( $A_1 B_1 = L \cdot D_1 C_1$ ), а отношение сходственных ребер равно  $K$ , задана векторами  $AB=a$ ,  $AD=b$ ,  $AA_1=c$ . Найдите вектор  $MN$ , если  $M$  - середина ребра  $CC_1$ , а  $N$  - центр тяжести треугольника  $AB_1 D_1$ .

з) Трапеции  $ABCD$  и  $A_1 B_1 C_1 D_1$ , расположенные в различных плоскостях, имеют общую вершину  $A$  и равные отношения оснований ( $AD/BC = A_1 D_1 / B_1 C_1 = L$ ). Проверьте, что отрезки  $BB_1$ ,  $CC_1$ ,  $DD_1$  параллельны некоторой плоскости (Примем  $AB_1=a$ ,  $AB=b$ ,  $AD=c$ ,  $A_1 D_1=d$ ).

и) На сторонах  $AB, BC, CD, DA$  косоугольного четырехугольника  $ABCD$ , не обязательно лежащего в одной плоскости, взяты точки  $A_1, B_1, C_1, D_1$  соответственно так, что  $AA_1=AB/2$ ,  $BB_1=BC \cdot 2/3$ ,  $CC_1=CD \cdot 3/4$ ,  $DD_1=DA/7$ . Проверьте, что векторы  $A_1 B_1, B_1 C_1, C_1 D_1$  компланарны. Предполагается, что заданы векторы  $AB=a$ ,  $AC=b$ ,  $AD=c$ .

### **Задания повышенной трудности**

к) Трапеции  $ABCD$  и  $A_1 B_1 C_1 D_1$ , расположенные в различных плоскостях, имеют общую вершину  $A$  и равные отношения оснований ( $AD/BC=A_1 D_1 / B_1 C_1=L$ ). Найдите вектор  $OO_1$ , соединяющий точки пересечения диагоналей трапеций. Предполагается, что заданы векторы  $AB=a$ ,  $AD=b$ ,  $AB_1=p$ ,  $AD_1=q$ .

л) Наклонная призма  $ABCD A_1 B_1 C_1 D_1$ , в основании которой лежит трапеция  $ABCD$  ( $BA = L \cdot CD$ ), построена на векторах  $BA=a$ ,  $BC=b$ ,  $BB_1=c$ . Вычислить вектор  $OD_1$ , где  $O$  - точка пересечения диагоналей основания.

м) Найти составляющие  $p_1, p_2, p_3$  вектора  $p$  на плоскости, определяемые векторами  $a$  и  $b$ , векторами  $c$  и  $b$ , векторами  $a$  и  $c$  соответственно при косом проектировании в направлении векторов  $c, b, a$  соответственно. Определить, являются ли вектора  $p$  и  $S=p_1+p_2+p_3$  коллинеарными. Если да, то определить коэффициент пропорциональности.

### **Лабораторная работа №15**

#### **Скалярное произведение векторов.**

Цель работы: Научиться составлять алгоритмы решения геометрических задач по теме "Скалярное произведение векторов", используя заданный набор процедур.

Файл LIST.2 содержит заголовки программ функции `det2`, `modulus`, `scalar` и следующие процедуры: `input`, `output`, `sum`, `subtract`, `multiply`, `angle`, `vectormult`.

#### **Задания к лабораторной работе.**

##### **Задание 1**

Для векторов  $a$  и  $b$  вычислите:

- а) скалярное произведение;
- б) модули векторов;
- в) угол между векторами в градусах;
- г) координаты векторного произведения;
- д) площадь треугольника, построенного на векторах  $a$  и  $b$ .

### **Задание 2**

Ромб задан векторами смежных сторон. Проверьте, то диагонали перпендикулярны.

### **Задание 3**

Тетраэдр  $SABC$  задан векторами трех ребер  $a, b, c$ , выходящих из одной вершины (точка  $S$  совпадает с началом координат). Найдите:

- а) объем тетраэдра;
- б) угол между векторами  $c$  и  $a-b$ ;
- в) площади грани  $ABC$ ;
- г) величину проекции вектора  $a$  на грань  $(ABC)$ ;
- д) центроид  $G$  тетраэдра.

### **Задание 4**

Дано некопланарные вектора  $a, b, c$ . Произведите ортогонализацию данного базиса.

### **Задание 5**

Тетраэдр  $SABC$  задан векторами трех ребер  $a, b, c$ , выходящих из одной вершины (точка  $S$  совпадает с началом координат). Найдите:

- а) центр вписанной сферы;
- б) расстояние между центром вписанной сферы и центроидом;
- в) величину двугранного угла при ребре  $SA$ ;
- г) расстояние между парой противоположных ребер.

## **Лабораторная работа «Точка»**

Цель работы: Научиться составлять алгоритмы решения геометрических задач по теме "Точка", используя заданный набор процедур.

Файл LIST.3 содержит заголовок программы функции  $det2$ ,  $det3$ ,  $modulus$ ,  $scalar$  и следующие процедуры:  $input$ ,  $output$ ,  $sum$ ,  $subtract$ ,  $multiply$ ,  $angle$ ,  $centre$ .

### **Задания к лабораторной работе**

#### **Задание 1**

Составить ПРОЦЕДУРУ нахождения координат точки  $M$ , делящей отрезок  $AB$  в отношении  $AM:MB = t_1:t_2$ . Добавьте ее в файл LIST.3.

#### **Задание 2**

Найдите координаты образа точки  $A$  при:

- a) центральной симметрии с центром  $C$ .
- b) гомотетии с центром  $C$  и коэффициентом  $k$ .

### Задание 3

Составьте программу для решения одной из следующих задач:

- a) проверьте, принадлежат ли три точки одной прямой.
- b) точки  $A$  и  $A_1$  делят отрезки  $BC$  и  $B_1C_1$  соответственно в равных простых отношениях. Проверьте, что середины отрезков  $AA_1$ ,  $BB_1$ ,  $CC_1$  лежат на одной прямой.

### Задание 4

Треугольник задан радиус-векторами вершин  $\vec{a}$ ,  $\vec{b}$ ,  $\vec{c}$ . Найдите:

- a) длины всех сторон.
- b) площадь треугольника.
- c) координаты центроида.
- d) координаты центра вписанной окружности;
- e) величины углов треугольника в градусах.

### Задание 5

Треугольник задан радиус-векторами вершин  $\vec{a}$ ,  $\vec{b}$ ,  $\vec{c}$ . Найдите:

- a) длины всех высот;
- b) длины всех медиан;
- c) длины всех биссектрис.

## Лабораторная работа «Прямая линия на плоскости»

Цель работы: Научиться составлять алгоритмы решения геометрических задач по теме "Прямая линия на плоскости", используя заданный набор процедур.

Файл LIST.4 содержит заголовок программы функции `det2`, `pointline`, `modulus`, `scalar` и следующие процедуры: `input`, `output`, `sum`, `inputline`, `outputline`, `subtract`, `multiply`, `angle`, `system2`.

### Задания к лабораторной работе

#### Задание 1

Для прямой, заданной общим уравнением составить ПРОЦЕДУРЫ нахождения:

- a) координат направляющего вектора.
- b) координат нормального вектора.

#### Задание 2

Составьте уравнение прямой, заданной:

- a) двумя точками;
- b) точкой и направляющим вектором;
- c) точкой и нормальным вектором;
- d) точкой и угловым коэффициентом;
- e) медиатрисой.

### Задание 3

Дана прямая и две точки  $F(f_1, f_2)$ ,  $G(g_1, g_2)$ . Найдите:

- простое отношение, в котором прямая делит отрезок  $FG$ .
- взаимное расположение точек относительно прямой.
- расстояния от точек до прямой.
- координаты образов точек при осевой симметрии относительно заданной прямой.

### Задание 4

Даны две прямые. Определите:

- взаимное расположение двух данных прямых (пересекаются, параллельны, совпадают).
- угол между прямыми.
- лежит ли заданная точка внутри полосы (проверить, параллельны ли прямые).

### Задание 5

Решите одну из следующих задач:

- Даны две пересекающиеся прямые и точка  $F(f_1, f_2)$ , не лежащая ни на одной из этих прямых. Найдите уравнение биссектрисы того угла, в котором лежит точка.
- Даны две пересекающиеся прямые и точка  $F(f_1, f_2)$ , не лежащая ни на одной из этих прямых. Найдите величину угла, в котором лежит точка.  
Даны две пересекающиеся прямые и точка  $F(f_1, f_2)$ , не лежащая ни на одной из этих прямых. Найдите условие, при котором эта точка лежит в остром угле
- Даны уравнения боковых сторон равнобедренного треугольника и точка на его основании. Составить уравнение прямой, содержащей его основание.
- Прямая  $q$  пересекает стороны  $AB$ ,  $BC$ ,  $CA$  треугольника или их продолжения соответственно в точках  $C_1$ ,  $A_1$ ,  $B_1$ . Проверьте, что середины отрезков  $AA_1$ ,  $BB_1$ ,  $CC_1$  принадлежат одной прямой.
- На прямых  $a$  и  $b$  заданы соответственно точки  $A, B, C$  и  $D, E, F$ . Проверьте, что точки пересечения прямых  $AD$  и  $CE$ ,  $BD$  и  $CF$ ,  $BE$  и  $AF$  принадлежат одной прямой.
- Исследовать взаимное расположение трех прямых.
- Определить, лежит ли точка внутри треугольника, заданного уравнениями сторон.
- Дан четырехугольник  $ABCD$ , у которого пары противоположных сторон пересекаются в точках  $S$  и  $T$ . Проверьте, что середины отрезков  $AC$ ,  $BD$  и  $ST$  лежат на одной прямой (теорема Гаусса).

## Лабораторная работа №16

### Простейшие графические операторы

#### Задание 1

Составьте программу, которая рисует на экране две линии по диагоналям экрана. Добавьте окружность диаметром в половину высоты экрана с центром в центре экрана и закрасьте две ее противоположные четверти. Покажите рисунок преподавателю и сотрите.

#### Задание 2

Нарисуйте на экране 16 разноцветных закрашенных фигур:

- 2.1. концентрических прямоугольников;

- 2.2. горизонтальных прямоугольников;
- 2.3. вертикальных прямоугольников;
- 2.4. прямоугольников, расположенных в виде решетки 4x4;
- 2.5. кругов, расположенных в виде решетки 4x4;
- 2.6. концентрических кругов.

Покажите рисунок преподавателю и сотрите.

### **Задание 3**

Используя модификации оператора line, нарисуйте (так, чтобы его было легко передвинуть, задавая начальные координаты рисунка с клавиатуры):

- 3.1. звезду;
- 3.2. кленовый лист;
- 3.3. домик с трубой;
- 3.4. четырехугольную пирамиду (без невидимых линий);
- 3.5. снежинку;
- 3.6. гайку (с отверстием);
- 3.7. каркас параллелепипеда;
- 3.8. что-нибудь еще.

Программу сохраните на диске под своим именем.

### **Задание 4**

Составьте и исполните программу для изображения графика какой-нибудь функции:

- 4.1.  $y=\cos(ax+b)$  (параметры a и b запросите с клавиатуры);
- 4.2.  $y=\text{tg}(x)$
- 4.3.  $y=\text{ctg}(x)$
- 4.4.  $y=x^2+ax+b$  (параметры a и b запросите с клавиатуры);
- 4.5.  $y=\text{exp}(ax)$  (параметр a запросите с клавиатуры);
- 4.6.  $y=\ln(x)$

Обозначьте оси, расставьте масштабные единицы, надпишите график функции.

```
{ОПЕРАТОР ВЫВОДА ТЕКСТА В ГРАФИЧЕСКОМ РЕЖИМЕ outtext('сообщение');}
```

### **Задание 5**

Используя изученные графические операторы, составьте программу построения своего собственного рисунка и сохраните ее на диске.

## **Лабораторная работа №17 Создание движущихся рисунков**

### **Задание 1**

Загрузите созданную Вами программу при выполнении последнего задания предыдущей лабораторной работы, выберите небольшой кусочек созданного Вами шедевра и организуйте его движение по любой траектории экрана с использованием динамической памяти для запоминания данного фрагмента рисунка и вывода его на экран.

### **Задание 2**

Загрузите программу рисования фигуры из задания 3 предыдущей лабораторной работы и организуйте движение Вашей фигуры по контуру экрана способом, аналогичным предыдущему заданию.

### ***Задание 3***

Составьте второй вариант программы так, чтобы движение организовывалось с помощью стирания рисунка (т.е. рисования его цветом фона) и рисования на новом месте (рядом с предыдущим). Сравните визуально результаты работы двух последних программ.

### ***Задание 4***

Создайте несколько фаз движения некоторого простого рисунка (например, ходьбы стилизованного человечка, полета птицы и т.п.). Организуйте их поочередное появления на некотором выбранном Вами месте экрана. Подберите паузы между появлением очередной фазы и стиранием предыдущей так, чтобы было максимально похоже на задуманное Вами движение.

### ***Задание 5***

Совместите чередование фаз движения и перемещение рисунка по экрану (например, человечек ходит по нижней кромке экрана, птица летит по диагонали экрана, ...).

### ***Дополнительное задание***

Разберитесь самостоятельно с организацией графики с помощью нескольких графических страниц и опробуйте их работу. В качестве модельного примера возьмите организацию движения колеса со спицами по горизонтальной прямой без проскальзывания. Реализуйте движение без использования графических страниц (1 балл) и с их использованием (2 балла) и сравните качество изображения при этих двух реализациях.

## **Лабораторная работа №18** **Использование датчика случайных величин**

### ***Задание 1***

Составьте программу, рисующую на экране «звездное небо», т.е. точки со случайными координатами и случайным цветом.

### ***Задание 2***

Составьте программу, имитирующую вращение круга в игре «Поле чудес». На экране должны поочередно зажигаться сектора с цифрами. Вращение должно заканчиваться при повторных запусках программы на разных секторах. Желательно, чтобы с течением времени вращение замедлялось. Программу сохраните.

### ***Задание 3***

Составьте программу, которая проверяет знание таблицы умножения, задавая пять различных вопросов. По итогам ответов опрашиваемому должна выставляться оценка. Программу сохраните.

### ***Задание 4***

Составьте программу, которая задумывает число в диапазоне от 1 до 1000, а затем отвечает «больше», «меньше» или «угадал» на вводимые Вами числа до тех пор, пока число не будет введено Вами правильно. **Дополнительное задание** (до 2 баллов). Научите машину угадывать задуманное Вами число. Затем составьте программу двусторонней игры «угадай число». Игра должна продолжаться до тех пор, пока один из игроков (Вы или машина) не отгадает задуманное противником число, при этом можно ограничить число попыток десятью.

### **Задание 5**

Исправьте программу из задания 3 таким образом, чтобы вопросы выбирались случайным образом из заранее сформированного массива вопросов по какой-либо теме, а ответы, соответственно, – из массива ответов.

### **Задание 6**

Дополните предыдущую программу таким образом, чтобы вопросы гарантированно выбирались без повторений. Массив вопросов должен отражать содержание выбранной Вами темы.

### **Дополнительные задания.**

1. Дополните программу из задания 6 таким образом, чтобы после ответов на вопросы выставлялась итоговая оценка, сопровождаемая, в зависимости от успехов, различными рисунками (**1 балл**).

2. Генерирование бриджевых раскладов

В бридж играют полной карточной колодой из 52 карт (по 13 в каждой масти 2,3,4,5,6,7,8,9,T(ten-10), J(jack-валет), Q(queen-дама), K(king-король), A(ace-туз)). В начале каждой раздачи колода случайным образом делится поровну между четырьмя игроками. Составить программу, которая генерирует случайные расклады и изображает их на экране. (**1 балл**).

3. Разберитесь самостоятельно с воспроизведением звуков средствами языка Паскаль и дополните какую-нибудь из своих программ звуками и мелодиями (**1 балл**).

4. Тест на внимание.

Составить программу, которая рисует таблицу 5x5 (NxN), заполняет ее в случайном порядке числами от 1 до 25, а затем проверяет, за какое время испытуемый сумеет указать все числа в возрастающем (убывающем) порядке (**до 3 баллов**).

5. Составьте программу игры «Пятнадцать» (**до 3 баллов**).

6. Попробуйте, используя программы из заданий 2 и 6, создать некое подобие игры «Поле чудес» для двух или нескольких игроков (**до 5 баллов**).

## **Лабораторная работа N 19. Работа с множествами**

### **Задание 1**

Составьте программу подсчета с помощью множеств в тексте всех

- 1.1. гласных букв
- 1.2. согласных букв
- 1.3. знаков препинания

Составьте с помощью множеств программу, удаляющую из текста

- 1.4. гласные буквы
- 1.5. согласные буквы

### 1.6. знаки препинания

Составьте с помощью множеств программу удваивания в тексте

### 1.7. согласных букв

### 1.8. гласных букв

## **Задание 2-3**

Составьте все нужные Вам в задании 4 процедуры или функции для

- ввода с клавиатуры числового множества
- ввода с клавиатуры числового массива
- вывода числового множества на экран
- вывода числового массива на экран
- вычисления количества элементов множества символов
- вычисления количества элементов числового множества
- ввода с клавиатуры множества символов
- вывода множества символов на экран
- вывода массива символов на экран
- перевода числового массива в числовое множество из тех же элементов
- перевода строки в множество символов, содержащихся в ней (все числа в диапазоне 1-100, а символы - произвольные).

Опробуйте их работу.

## **Задание 4**

Составьте программу для решения одной из следующих задач:

- 4.1. Даны два массива. Сравнить множество их значений.
- 4.2. Дан массив и множество. Составить программу выясняющую, входит ли каждый элемент массива в множество.
- 4.3. Даны две строки. Сравнить множества их символов.
- 4.4. Определить, сколько выходных дней среди чисел мая текущего года, содержащихся в данном числовом массиве.
- 4.5. Дан массив натуральных двузначных чисел. Проверить, все ли его элементы являются простыми числами.
- 4.6. Даны две строки. Из множества символов первой строки удалить символы содержащиеся во второй строке.

## **Задание 5**

Составьте программу для решения одной из следующих задач:

- 5.1. Сформировать множество чисел из первой сотни, которые имеют простыми делителями только числа 2,3,5.
- 5.2. сформировать множество простых чисел (от 1 до 100) с помощью алгоритма "решето Эратосфена"
- 5.3. Выяснить, есть ли общий элемент во всех данных одномерных массивах, вводимых с клавиатуры
- 5.4. Вывести на экран множество элементов, которые входят хотя бы в один из данных одномерных массивов
- 5.5. В двумерном массиве определить, есть ли общий элемент во всех строках
- 5.6. Дан русский текст. Подсчитать количество слов.
- 5.7. Дан текст, состоящий из арифметических выражений и слов. Подсчитать количество арифметических выражений не содержащих переменных и функций.



## **Лабораторная работа № 20 Работа с записями**

### ***Задание 1***

Загрузите программу lab5.pas и исполните ее, используя процедуру prismswainie

### ***Задание 2***

Дополните процедуру вывода так, чтобы вместе с фамилиями печатались и инициалы каждого студента.

### ***Задание 3***

Дополнить основную программу процедурой, которая вычисляет и печатает:

- 3.1 число студентов сдавших все экзамены и зачеты
- 3.2 средний балл группы
- 3.3 количество студентов, сдавших экзамены без двоек
- 3.4 количество студентов, сдавших экзамены без троек
- 3.5 количество студентов, сдавших экзамены на одни пятерки.
- 3.6 фамилия студента, имеющего самый высокий средний балл
- 3.7 предмет, успеваемость по которому была самой высокой
- 3.8 предмет, успеваемость по которому была самой низкой

### ***Задание 4***

Дополнить основную программу процедурой, которая вычисляет и печатает размер стипендии назначенной каждому студенту.

### ***Задание 5***

Дополнить основную программу процедурой, которая вычисляет и печатает

- 5.1 количество человек не получающих стипендию
- 5.2 количество человек получающих обычную стипендию
- 5.3 количество человек получающих стипендию повышенную на 25%
- 5.4 количество человек получающих стипендию повышенную на 50%

## **Лабораторная работа N21 Работа с записями**

### ***Задание 1***

Загрузить программу lab6.pas исполнить ее. Дополнить программу так, чтобы вычислялось и печаталось:

- 1.1 число мужчин;
- 1.2 средний возраст;
- 1.3 количество живущих в Ярославле;
- 1.4 число русских;
- 1.5 количество родившихся в первой половине месяца;
- 1.6 количество родившихся в зимние месяцы;
- 1.7 количества совершеннолетних.

### ***Задание 2***

Изменить основную программу так, чтобы вычислялся и печатался:

- 2.1 минимальный номер дома;
- 2.2 максимальный номер квартиры;
- 2.3 улица с самым длинным названием;
- 2.4 город с самым коротким названием;
- 2.5 самая длинная фамилия;
- 2.6 самое короткое имя;
- 2.7 самый маленький индекс.

### **Задание 3**

Изменить программу так, чтобы проверялось одно из приведенных ниже условий и, либо печаталась пара соответствующих записей, либо печаталось сообщение о том, что условие не выполнено:

- 3.1 есть ли однофамильцы;
- 3.2 есть ли тезки по имени и отчеству;
- 3.3 есть ли люди с одинаковым днем рождения;
- 3.4 есть ли люди, живущие на одной улице;
- 3.5 есть ли люди, живущие в одном доме;
- 3.6 есть ли люди, живущие в одной квартире.

### **Задание 4**

Изменить основную программу так, чтобы вычислялся и печатался:

- 4.1 распределение числа родившихся по месяцам года;
- 4.2 распределение числа родившихся по дням месяца;
- 4.3 распределение числа родившихся по годам (нулевые числа должны пропускаться);
- 4.4 распределение числа живущих по городам [+1 балл].

### **Задание 5**

Упорядочить массив записей по возрастанию:

- 5.1 длины фамилии;
- 5.2 года рождения;
- 5.3 длины названия улицы;
- 5.4 номера дома;
- 5.5 длины названия города;
- 5.6 номера квартиры;
- 5.7 возраста [+ 1 балла];
- 5.8 по алфавитному порядку фамилий.

### **Задание 6**

Выбрать и распечатать (в том же виде, что и в программе) всех:

- 6.1 людей, живущих в Ярославле;
- 6.2 родившихся весной;
- 6.3 русских;
- 6.4 женщин;
- 6.5 людей с фамилией короче 10 букв;
- 6.6 с трехзначным номером квартиры;
- 6.7 мужчин с именем короче 8 букв;
- 6.8 русских с номером дома из одной цифры.

## Лабораторная работа N22 Работа с файлами записей

### Задание 1

Составить программу для чтения файла записей со следующей структурой:

фамилия ( string[20] )  
имя ( string[20] )  
специальность ( string[20] )  
год рождения ( 1..9999 )  
год смерти ( 1..9999 )  
год открытия ( 1..9999 )  
открытие ( string[255] )  
страна ( string[40] )

Исполнить программу, дав файловой переменной имя "sciense". Распечатать фамилии, которые есть в этом файле.

### Задание 2

Оформить печать, составив процедуру вывода. При этом на экране должно быть:  
Фамилия и первая буква имени с позиции 3 до позиции 20

- годы жизни с 22 по 31
- страна с 33 по 41
- открытие с 43 по 67
- год открытия с 69 по 75

### Задание 3

Дополнить печать колонкой "Число прожитых лет" с 77 по 80. Вывести на экран фамилию ученого, прожившего дольше остальных.

### Задание 4

Распечатать с помощью процедуры вывода данные (в том же виде)

- 4.1 только о русских и советских ученых
- 4.2 только ученых, родившихся в XVIII веке
- 4.3 только ученых, родившихся в XIX веке
- 4.4 только ученых физиков

### Задание 5

Определить, наиболее часто встречающуюся в данном файле

- 5.1 страну
- 5.2 специальность
- 5.3 век (среди лет открытий)

## Лабораторная работа N23 Работа с файлами прямого доступа

### Задание 1

- 1.1. Составить процедуру, которая создает файл из вещественных чисел, количество и значение которых, так же как и имя создаваемого файла вводится с клавиатуры;
- 1.2. То же задание, что и в 1.1, но для целых чисел.

### Задание 2

Составить процедуры для чтения данных из файла в массив и печати полученного массива. Исполнить программу, дав файловой переменной имя файла, созданного Вами при выполнении задания 1.

### ***Задание 3***

Составить логическую функцию, которая не использует массивов и проверяет, выполняется ли свойство:

- 3.1 упорядоченности элементов некоторого файла по возрастанию;
- 3.2 что отрицательных элементов некоторого файла больше, чем положительных;
- 3.3 наличия нулевых элементов в некотором файле;
- 3.4 упорядоченности элементов некоторого файла по убыванию;
- 3.5 одинаковы ли два файла.

Проверить работу созданной функции на Ваших файлах.

### ***Задание 4***

- 4.1 Составить и опробовать работу процедуры, которая, не используя массивов, ищет в указанном файле максимальный элемент, и выдает в качестве ответа его номер в файле;
- 4.2 То же, но для минимального элемента;
- 4.3 То же но для первого положительного элемента;
- 4.4 То же но для первого отрицательного элемента.

### ***Задание 5***

Составить и опробовать работу процедуры, которая не использует массивов и решает одну из следующих задач:

- 5.1. из двух файлов F и G одинаковой длины создает третий файл, в который записывается больший из компонентов двух файлов при их последовательном и одновременном просмотре или выдает сообщение о том, что файлы имеют разную длину;
- 5.2. в данном файле изменяет его содержимое таким образом, чтобы все его элементы с нечетными номерами были удвоены;
- 5.3. переписывает компоненты файла F1 в файл F2 в обратном порядке;
- 5.4. создает новый файл из тех целых чисел исходного файла, которые являются полными квадратами;
- 5.5. подсчитывает количество элементов файла F, меньших среднего арифметического всех элементов этого файла;
- 5.6. из файла, количество компонент которого кратно 3, переписывает содержимое в другой файл, производя в каждой тройке подряд идущих чисел (a b c) перестановку в порядке (c b a), причем, если количество компонент файла не кратно трем, то процедура выдает соответствующее сообщение и не выполняется.

### ***Дополнительные задания***

#### ***1. Бинарный поиск (1 балл)***

Составить и опробовать работу процедуры, которая определяет, есть ли в данном упорядоченном по возрастанию файле вещественных (целых) чисел данное число, с помощью бинарного поиска (дихотомии), т.е. делением на каждом шаге интервала поиска числа на две примерно равные половины.

#### ***2. Недостающее число (1 балл)***

Дан текстовый файл, имя которого вводится с клавиатуры. В его первой строке записано некоторое натуральное число  $N$  ( $N \leq 30\,000$ ). Далее записано  $N-1$  попарно различных натуральных чисел, каждое из которых не превосходит  $N$ , разделенных пробелами или символами концов строк. Составить программу, которая анализирует содержимое файла и сообщает, какое число пропущено. При этом время работы не должно превышать 1 сек.

### **3. Общие буквы (1 балл)**

Даны два текстовых файла, имена которых вводятся с клавиатуры. Составить программу, которая создает третий текстовый файл, состоящий из трех строчек. В первой должны быть записаны все русские буквы, которые есть в обоих файлах (маленькие буквы отличаются от больших), во второй строке – те буквы, которые есть только в первом файле, и в третьей строке – те буквы, которые есть только во втором файле. Имя получающегося файла тоже вводится с клавиатуры.

## **Лабораторная работа N 24 Работа с текстовыми файлами.**

### **Задание 1**

Составить и опробовать процедуру, считывающую текстовый файл с именем "sem1.21?", в котором каждая строка представляет собой фамилию студента и список всех оценок по информатике, полученных им за первый семестр. При этом фамилия отделяется от оценок запятой, так же как и оценки друг от друга. Знак "?" в имени файла надо заменить на порядковый номер вашей группы. Для проверки правильности чтения из каждой строки надо выделять фамилию и распечатывать их в столбик.

### **Задание 2**

Дополнить предыдущую процедуру так, чтобы после считывания строки и выделения фамилии остаток строки разбивался на отдельные оценки, из которых создаются следующие числовые массивы:

- оценки за лабораторные работы (по 15 оценок у каждого)
- баллы за самостоятельные работы (по 5 оценок у каждого)
- баллы за собеседования (по 5 оценок у каждого)
- оценки за контролируемые программы (по 6 оценок у каждого)
- дополнительные баллы
- балл за итоговую контрольную работу

При этом должны вычисляться итоговые суммы по оценкам, записанным в строке и печататься вместе с фамилией на экран.

### **Дополнительное задание (+1 балл):**

Составить и проверить работу процедуры, которая проверяет по полученным массивам корректность выставления оценок и выдает сообщение о том, что все оценки допустимые или полные сообщения о недопустимых оценках. При этом процедура должна производить пересчет неверных оценок. Например, если оценка за лабораторную работу 7 или 10, то ее надо заменить на 5, а оценку -3 за самостоятельную работу надо заменять на оценку -2 и т.д.

### **Задание 3**

Составить процедуру печати информации в виде таблицы.

### **Задание 4**

Составить и проверить работу процедуры, которая создает текстовый файл, в каждой строке которого записана фамилия студента и набранная им за семестр общая сумма баллов, отделенная от фамилии знаком "-".

### ***Задание 5***

Дополнить предыдущую программу так, чтобы в файл данные записывались упорядоченными по убыванию суммы набранных студентом за семестр баллов.

### ***Дополнительное задание (1 балл).***

С помощью составленной программы создать два текстовых файла с итогами работы за семестр двух групп вашего потока (или курса). Затем составить процедуру, которая из двух отсортированных файлов создает один общий отсортированный файл, где к каждой строке добавлен еще номер группы.

## Примерные вопросы к собеседованиям

### Величина. Команды присваивания, ветвления и выбора

1. Что называется алфавитом языка?
2. Дайте определения величины, выражения, оператора языка программирования.
3. Опишите общую структуру программы на языке Паскаль. Как называются основные части программы?
4. Как по назначению разделяются величины в программе? Дайте определения.
5. Какие типы величин вы знаете? Приведите примеры.
6. Есть ли разница (с точки зрения языка Паскаль) между числами 100 и 100.0, 20 и 2E+1?
7. По какому признаку (величине или форме записи) отличаются целые числа от вещественных?
8. Расскажите о форме представления целых чисел в ПЭВМ.
9. Какое выражение может быть идентификатором на языке Паскаль?
10. Дайте определение арифметического выражения.
11. Сформулируйте правила вычисления арифметических выражений.
12. Расскажите о пользовательских типах данных.
13. Объясните назначение и правила записи команды присваивания.
14. Объясните правила исполнения команды присваивания.
15. Какие ограничения в зависимости от типов величин накладываются при записи команды присваивания?

- |          |           |           |              |                                    |
|----------|-----------|-----------|--------------|------------------------------------|
| 1. ABC   | 5. A.B.C. | 9. A 1    | 13. ABC_D    | 17. 10A                            |
| 2. 1ABC  | 6. A      | 10. AB1.C | 14. ...A     | 18. 1998                           |
| 3. A1BC  | 7. 1      | 11. .ABC  | 15. H2SO4    | 19. H <sub>2</sub> SO <sub>4</sub> |
| 4. A B C | 8. ABC123 | 12. AB?   | 16. 23-42-17 | 20. "ABC"                          |

16. Укажите, какие из данных выражений компьютер не воспримет как числа:

|         |      |       |        |
|---------|------|-------|--------|
| -1.5E-7 | 6.   | 6.E-3 | 5.0E-4 |
| E-05    | 0.6  | 6E-3  | 5.0E   |
| 70E2    | .264 | 6E3   | 5.1E12 |

17. Можно ли утверждать, что в Паскале значение выражения  $(1/3)*3-1=0$ ?

18. Определите, верна ли запись команды:

var x, y: real;

g: integer;

b: boolean; a, c: string;

- |             |             |                      |                         |
|-------------|-------------|----------------------|-------------------------|
| (1) g:=x+y; | (5) x/y:=x; | (9) "Овод":=a;       | (13) b:=3<5;            |
| (2) b:=12;  | (6) x:=x/y; | (10) c:=Война и мир; | (14) g:=x*x;            |
| (3) x:=g;   | (7) g:=x-y; | (11) x:=g-b;         | (15) b:="true"          |
| (4) x:=c;   | (8) a:=a+1; | (12) b:=x+y;         | (16) b:=(2=5) or (6>3); |

19. Найдите ошибки в записи команд.

|  |   |   |  |
|--|---|---|--|
| (1). Var x,y:real;<br>Begin<br>Y:= '125';<br>X+2:=y;<br>End. | (2). var x,y:integer;<br>begin<br>x:=25;<br>y:=x/5;<br>end. | (3). var x,y:char;<br>begin<br>x:= '1'; y:= 'qwert'<br>y:=y+x; x:=y-x;<br>end | (4). var 3x;<br>y_5:integer;<br>begin<br>3x:=x*y_5;<br>y_5:=sin(3x);<br>end. |
|--|---|---|--|

|  |  |  |   |
|--|--|--|---|
| (5). var<br>x:char;y:string;<br>begin<br>x='a'; x:=x+x;<br>y:=3*x;<br>end. | (6). var x,y:string;<br>begin<br>x:='ток';<br>y:=-x;<br>end. | (7). var w,q:boolean;<br>begin<br>w:='true';<br>q:='фальшь';<br>end;             | (8). var<br>x:real;y:integer;<br>s:boolean;<br>begin x:=5; y:=5;<br>x:=x+y; y:=x-y;<br>s:=y; end. |
| (9). var x:y:real;<br>begin x:=2,3; y:=1\ x;<br>writeln(y:5:10);<br>end.   | (10). Var x,y:real;<br>Begin read(x);<br>y:=x mod 3;<br>end. | (11). var s,p:integer;<br>begin readln(x);<br>y:=2*x+123;<br>writeln(y:2:3);end. | (12). var<br>x,y,z,a,x,w:real;<br>begin x:=2e+1.2;<br>y:=2x/5;<br>writeln(xy);end,                |

20. Определите типы величин x, y и z, входящих в каждую серию команд.

|   |   |   |   |   |
|---|---|---|---|---|
| (1). Readln(x,y);<br>z:=x+y;<br>y:=2*x+z; | (2). Readln(x,y);<br>z:=x+y;<br>z:=x+z; | (3). readln(x,y);<br>z:=x/y;<br>y:=x*z; | (4). readln(x,y);<br>z:=x+y;<br>z:=x-z; | (5). readln(x,y);<br>z:=x+y+t;<br>y:=z; |
|---|---|---|---|---|

21. Найдите значение y после исполнения серии команд:

|  |   |  |
|--|---|--|
| (1) x:=1; y:=2;<br>x:=x+y;<br>y:=x-y;<br>x:=x*x;<br>y:=x;              | (2) x:=1; y:=2; z:=3;<br>x:=y;<br>y:=z;<br>z:=x+y;<br>y:=z;           | (3) x:= -1; y:=2;<br>y:=x*y;<br>y:=y*y;<br>x:=y-1;<br>y:=x-y;          |
| (4) x:=2; y:=3;<br>x:=y-x;<br>x:=x*x;<br>y:=x;<br>y:=y-x;              | (5) x:=0; y:=1; z:=2;<br>y:=x-1;<br>z:=y;<br>x:=y;<br>y:=x+y+z;       | (6) x:= -1; y:=2; z:=3;<br>x:=y+1;<br>y:=y*x;<br>z:=y;<br>y:=y*z;      |
| (7) y:=4;<br>y:=y+1;<br>y:=y*4;<br>y:=20-y;<br>y:=12-y;<br>y:=y*y-100; | (8) y:=2;<br>y:=y+4;<br>y:=y-4*y;<br>y:=y+y/3;<br>y:=y+22;<br>y:=y*y; | (9) y:=2;<br>y:=y-y*3;<br>y:=y-2;<br>y:=y-y/3;<br>y:=y+24;<br>y:=y/2;  |
| (10) y:=6;<br>y:=y*y;<br>y:=y-12;<br>y:=y+y/2;<br>y:=y-16;<br>y:=y/5;  | (11) y:=0;<br>y:=y+7;<br>y:=y*y-4;<br>y:=y+3;<br>y:=y/8+2;<br>y:=y/4; | (12) y:= -1;<br>y:=y*y;<br>y:=y+5;<br>y:=y-2;<br>y:=y*y-11;<br>y:=y*2; |
| (13) y:='12';<br>y:='34'+y;<br>y:=y+'56';<br>y:=y+'0';                 | (14) y:='09';<br>y:=y+'87';<br>y:=y+y;<br>y:='-1'+y;                  | (15) y:='1';<br>y:=y+y+'0';<br>y:='01'+y;<br>y:='0'+y;                 |
| (16) y:=2;<br>y:='1'+y;  | (17) y:='10';<br>y:=y+'1';  | (18) y:='3.';<br>y:=y+'4.';  |



|  |   |   |
|--|---|---|
| y:=y+2'+y;<br>y:='11'+y;                 | y:='0'+y+'0';<br>y:='1'+y;                | y:='2.'+y;<br>y:='1.'+y+'5';                  |
| (19) y:='н';<br>y:='ба'+y;<br>y:=y+'ка'; | (20) y:='н';<br>y:='ба'+y;<br>y:='ка'+y;  | (21) y:='с';<br>y:='со'+y;<br>y:='на'+y;      |
| (22) y:='со';<br>y:=y+'с';<br>y:=y+'на'; | (23) y:='те';<br>y:='ка'+y;<br>y:=y+'па'; | (24) y:='ди';<br>y:='ар'+y;<br>y:='г'+y+'на'; |

22. Каково было значение величины x, если после исполнения серии команд получилось y=5?

|                                  |                                 |                                   |                                   |                                   |                                   |
|----------------------------------|---------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|
| (1). z:=x*4<br>y:=24-z<br>y:=5+y | (2). z:=x/4<br>y:=2-z<br>y:=5*y | (3). z:=x*2<br>y:=5+z<br>y:=6/y+3 | (4). z:=x*4<br>y:=12-z<br>y:=25-y | (5). z:=x*4<br>y:=12+z<br>y:=15/y | (6). z:=x*4<br>y:=12+z<br>y:=20/y |
|----------------------------------|---------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|-----------------------------------|

23. При каком начальном значении переменной после исполнения серии команд получится указанный ответ:

|  |  |   |   |
|--|--|---|---|
| (1) x:=x-1;<br>x:=x/3;<br>x:=x+4;<br>Ответ:<br>a) x=1<br>b) x=-1<br>c) x=0<br>d) x=4 | (2) y:=y+1;<br>y:=y*2;<br>y:=y-7;<br>Ответ:<br>a) y=3<br>b) y=-1<br>c) y=0<br>d) y=1 | (3) y:=y-4;<br>y:=y*3;<br>y:=y+2;<br>Ответ:<br>a) y=5<br>b) y=-5<br>c) y=0<br>d) y=11 | (4) y:=y/2;<br>y:=y+4;<br>y:=y-6;<br>Ответ:<br>a) y=6<br>b) y=-6<br>c) y=12<br>d) y=0 |
|--|--|---|---|

|  |   |  |   |
|--|---|--|---|
| (5) y:=y+'ло'+y;<br>y:=y+'л';<br>Ответ:<br>y='колокол'                     | (6) y:=y+'т'+y;<br>y:=y+'к';<br>y:=y+'а';<br>Ответ:<br>y='атака'      | (7) y:=y+'к'+y;<br>y:='с'+y;<br>y:=y+'л';<br>Ответ:<br>y='сокол'             | (8) y:=y+y;<br>y:=y+'а';<br>y:=y+'т';<br>Ответ:<br>y='арарат'           |
| (9) y:=y+'01'+y+'1';<br>y:=y+'2';<br>y:='20'+y+'0';<br>Ответ: y='20101120' | (10) y:=y+'3'+y;<br>y:='2'+y+'4';<br>y:='1.'+y;<br>Ответ: y='1.2.3.4' | (11) y:=y+'3'+y;<br>y:='2'+y+'4'+y;<br>y:='1'+y+'5';<br>Ответ: y='1+2+3+4+5' | (12) y:='0'+y+'1'+y;<br>y:=y+'01';<br>y:='1'+y;<br>Ответ: y='101111101' |
| (13) y:=y+'г'+y;<br>y:='т'+y+'р';<br>y:='in'+y;<br>Ответ: y='integer'      | (14) y:='b'+y+y;<br>y:=y+'le';<br>y:=y+'an';<br>Ответ: y='boolean'    |  |   |

24. Найдите значение выражения.

|               |               |         |                  |                  |
|---------------|---------------|---------|------------------|------------------|
| 15-3*2=       | 110-10/5=     | 18-6/2= | 2.8e2-0.08e3=    | 0.123e4-3.2e2=   |
| 18/3*2=       | 20/10*2=      | 20/5*4= | 6e4*0.5e-4+10=   | 4.0e2*0.25e-2-1= |
| 5.8e2-0.08e4= | 8e2/0.16e3-1= | 16-3*5= | 8.9e3-0.09e5=    | 5.036e1-6e-2=    |
| 15/3*5=       | 10-2/4=       | 21-3/3= | 4.50e4/0.3e-4-1= | 2.50e-4/5e-6+1=  |
| 18-3/2=       | 12/4*3=       | 16/8*4= | 7.50e4/0.15e5+1= | 0.3e4-0.025e3=   |

24. Поменяйте значения переменных А и В:
- с помощью вспомогательной переменной;
  - без использования вспомогательной переменной.
25. Поменяйте значения переменных М, N, P, T так, чтобы М приобрела значение Т, N - M, P - N, T - P:
- с помощью вспомогательной переменной;
  - без использования вспомогательной переменной.
26. Перечислите виды условий. Как записывается условие на Паскале?
27. Перечислите операции отношения и логические операции. Назовите двуместные и одноместные логические операции.
28. Дайте определение логического выражения.
29. Сформулируйте правила вычисления логических выражений.
30. Объясните назначение и правила записи команды ветвления.
31. Объясните правила исполнения команды ветвления.
32. Объясните назначение и правила записи команды выбора.
33. Объясните правила исполнения команды выбора.
34. Сравните возможности команды выбора на алгоритмическом языке и Паскале.
35. Как надо поступать в случае, если в серию команд входит более одной команды?
36. Найдите синтаксические ошибки в записи команд:
- if 5 then S:=S+5;
  - if B then x:=y; else B:=x;
  - if x=y then x:=1; else x:=0;
  - if (A=B) and P then P:=P+12;
  - if (A<B) or C then C:='false';
  - if P then P:='a=b';
  - if x>y then begin x:=y else x:=z end;
  - if C then A:=0 else C:=2;
  - if x="false" then write ("ложь") else x:=true;
  - if X or Y then begin write (Z) else write (T) end;
  - if x<>y then else x:=y;
  - if y=z then end;
  - if x>0 then  
y:=x  
else y:=-x;  
end;
  - if c>30  
then  
begin  
y:='жара';  
else y:=1-t  
end;
  - case n of  
1..5: y:=3;  
6..9: y:=4;  
10: y:=5;
  - case k of  
1.51: y:=2;  
else 1.73: y:=3  
end;

38. Найдите значение у после исполнения серии команд:

|   |  |   |
|---|--|---|
| <p>(1). x:=4;y:=5;<br/>if x&gt;0<br/>then if y&lt;8<br/>then y:=sqr(x)<br/>else y:=2*x+1<br/>else y:=0;</p> | <p>(2). N:=7;<br/>case N of<br/>1..10: y:=1;<br/>7: y:=2;<br/>end;</p> | <p>(3). x:=-4;y:=5;<br/>if x&gt;0<br/>then if x&gt;0<br/>then if x&lt;8<br/>then y:=sqr(x)<br/>else y:=2*x+1;</p> |
|---|--|---|

|  |   |   |
|--|---|---|
| (4). x:=12;y:=5;<br>if x>0<br>then if x<8<br>then y:=sqr(x)<br>else y:=2*x+1;<br>y:=0; | (5). x:=8;<br>if x>9<br>then y:=2*x+1<br>else if x<7<br>then y:=0<br>else y:= sqrt(10-x); | (6). x:=5;<br>if x>0<br>then y:=2*x+1<br>else if x<8<br>then y:=sqr(-x)<br>else y:=0; |
|--|---|---|

40. Перепишите команду ветвления через команду выбора: (x и t-целые числа в диапазоне от 0 до 70).

- (1). if (t>=16) and (t<25) then writeln ('норма') else if (t>=26) then writeln ('повышена') else writeln ('понижена');
- (2). if (x>=10) and (x<37) then y:=2\*x+1 else if (x>=0) and (x<55) then y:=x\*x\*x else y:=0;

41. Перепишите команду выбора через команду ветвления.

- (1). case t  
  -273..0: o:='лед';  
  1..100: o:='вода';  
  101..200: o:='пар';  
end;
- (2). case k of  
  10..20: f:=sqrt(k-5);  
  0..150: f:=sqr(k+1)  
  else f:=sqrt(abs(k))  
end;

42. Определите, при каком значении аргументов после исполнения фрагмента программы получится указанный ответ (для каждого случая указать все решения).

- (1). if (a=0) and (b>0) then writeln ('х-любое число') else if (a=0) and (b<=0) then writeln ('решений нет') else if (a<0) then writeln ('x<',-b/a:5:2) else if (a>0) then writeln ('x>',-b/a:5:2);
- (2). if (x >=-1) and (x <=3) then y:=x\*2 else if (x >=-4) and (x<5) then y:=x\*x-10 else if (x >=5) then y:=x else y:=1;

ОТВЕТ:1. x<5

2. решений нет  
3. x>-3

ОТВЕТ:1. y=6 2.

2. y=-25 3.  
3. y=1

43. Как работает при разных значениях аргументов следующий фрагмент программы?

```
d:=b*b-4*a*c;
if d<=0
then writeln('Действительных корней нет')
else x1:=(-b-sqrt(d))/2/a; x2:=(-b+sqrt(d))/2/a;
writeln('x1=',x1:5:2,' x2=',x2:5:2);
```

44. Перепишите команду ветвления без составных условий.

- (1). if (t>=36) and (t<37) then writeln ('норма') else if (t>=37) and (t<42) then writeln ('повышена') else if (t>=35) and (t<36) then writeln ('понижена') else writeln ('ошибка');
- (2). if (a=0) and (b=0) then writeln ('х-любое число') else if (a=0) and (b<>0) then writeln ('решений нет') else if (a<>0) then writeln ('x=',b/a:5:2);

```
(3). if (x >=-6) and (x<7)
    then y:=x+2
    else if (x >=0) and (x<12)
        then y:=-x*x
        else if (x >=5) and (x<26)
            then y:=x else y:=1;
```

```
(4). if (x>=0) and (x<7)
    then y:=2*x+1
    else if (x>=-7) and (x<25)
        then y:=x*x*x
        else if x>=35
            then y:=1-x
            else y:=0;
```

```
(5). if (a=0) and (b>0)
    then writeln ('x-любое число')
    else if (a=0) and (b<=0)
        then writeln ('решений нет')
        else if (a<0)
            then writeln ('x<',-b/a:5:2)
            else if (a>0)
                then writeln ('x>',-b/a:5:2);
```

## Циклы

1. Объясните назначение команды повторения.
2. Назовите различные формы команды повторения на Паскале.
3. В каких случаях используется оператор FOR?
4. Какие формы записи оператора FOR существуют?
5. Как происходит работа оператора FOR?
6. Как могут располагаться друг относительно друга различные циклы внутри одной программы?
7. Какими должны быть идентификаторы параметров вложенных циклов?
8. Для каких значений параметра исполняется серия команд цикла с параметром?
9. Какие ограничения на начальное, конечное значение параметра и величину шага существуют на языке Паскаль?
10. Сколько раз исполнится тело цикла FOR в зависимости от формы оператора, начального и конечного значений:

|            | НЗ<КЗ | НЗ=КЗ | НЗ>КЗ |
|------------|-------|-------|-------|
| for to     |       |       |       |
| for downto |       |       |       |

11. Как организовать цикл с помощью оператора REPEAT?
12. Как организовать цикл с помощью оператора WHILE?
13. В чем отличие и сходство циклов REPEAT и WHILE?
14. Сравните работу циклов FOR, WHILE, REPEAT по следующим параметрам:
  - a) обязателен ли параметр цикла, и каким он должен быть;
  - b) требуется ли начальная установка параметра цикла;
  - c) возможно ли изменение параметра цикла в теле цикла;
  - d) в каком случае цикл завершается, и как определяется количество повторов;
  - e) может ли цикл не выполниться ни разу;
  - f) исполняется ли цикл всегда хотя бы один раз;
  - g) обязательно ли наличие операторных скобок в составном операторе тела цикла.
15. Переписать цикл FOR через WHILE (с положительным шагом).
16. Переписать цикл FOR через WHILE (с отрицательным шагом).

17. Переписать цикл FOR на алгоритмический язык:
- 17.1. с положительным шагом.
  - 17.2. с отрицательным шагом.
18. Переписать цикл REPEAT на алгоритмический язык.
19. Переписать цикл WHILE через REPEAT.
20. Переписать цикл REPEAT через WHILE.
21. Переписать команду повторения "ДЛЯ" алгоритмического языка через FOR на Паскаль:
- 21.1. с положительным шагом.
  - 21.2. с отрицательным шагом.
  - 21.3. в общем виде.
22. Какие команды повторения на языке Паскаль и в каком случае не исполняются ни разу?
23. Сколько раз исполнится данная серия команд:

|   |   |  |   |
|---|---|--|---|
| (1). m:=0;<br>for i:=20 downto 2<br>do<br>m:=m+1;                       | (2). x:=1;<br>repeat<br>y:=sqr(x);<br>x:=x+2;<br>until x>=25; | (3). t:=-10;<br>while t<0 do<br>begin y:=sqr(x);<br>t:=t+1;<br>end;                          | (4). p:=1;<br>for i:=10 downto 8<br>do<br>p:=p*2; |
| (5). x:=1; y:=0;<br>for i:=1 to x do<br>j:=5 downto y do<br>write(i+j); |   | (6). i:=1;<br>a:=i<=2; p:=false;<br>while a or p do begin<br>i:=i+1; a:=i<=2; p:=not(p) end; |   |
| (7). b:='123';<br>repeat<br><br>b:=b+b;<br><br>until true;              |   | (8). a:= -1;<br>repeat<br>a:=a+1;<br>until a<>0;   |   |

24. При каком последнем значении будет исполняться цикл:

|                            |                                       |  |
|----------------------------|---------------------------------------|--|
| (1). for i:=10 downto 8 do | (2). g:=0;<br>while g<0 do<br>g:=g-1; | (3). L:=13;<br>repeat L:=13/2;<br>until L>6; |
|----------------------------|---------------------------------------|--|

25. Каково будет значение параметра цикла после его окончания:

|  |   |  |
|--|---|--|
| (1). n:=2;<br>for i:=10 downto 1 do<br>n:=n+3; | (2). y:=-1;<br>repeat h:=sqr(y);<br>y:=y+1;<br>until y>=16; | (3). u:=-25;<br>while u<=0 do<br>begin r:=sqr(u);<br>u:=u+5;<br>end; |
| (4). w:=1;<br>for i:=8 to 8 do<br>w:=w*2-1;    | (5). for c:='a' to 'z' do<br>write(c);                      | (6). i:=1;<br>repeat<br>x:=10-i;<br>i:=i-4;<br>until i<-8;           |

|  |   |
|--|---|
| (7). i:=1;<br>a:=i<=2; p:=false;<br>while a or p do<br>begin i:=i+1;<br>p:=not(p) end; | (8). y:=4<5; x:=-2;<br>repeat<br>x:=sqr(x);<br>y:=not(y);<br>until y; |
|--|---|

26. Найдите значение у после исполнения серии команд:

|   |   |  |
|---|---|--|
| (1). y:=1;<br>for i:=14 downto 2 do<br>y:=3*y-2;                        | (2). y:=12;<br>repeat y:=y/2+3;<br>until y>0;<br>y:=y/3;  | (3). y:=1;<br>while (y>12) and (y<25)<br>y:=sqr(y)-5;  |
| (4). readln(b);<br>y:=56;<br>repeat<br>q:=b+10<br>y:=y/8;<br>until q>b; | (5). y:=8;<br>w:=true;<br>while (y>0) and w do begin<br>if not(y mod 2=0)<br>then w:=not(w)<br>else y:=y/2;<br>end; | (6). y:=45;<br>w:=false;<br>while not(w) do begin<br>if y mod 3=0<br>then y:=y/3<br>else w:=true;<br>end;  |
| (7). f:=256;y:=f;<br>for i:=5 to trunc(f/32) do<br>y:=sqrt(y);          | (8). s:='';y:=-2;<br>repeat<br>if y>5<br>then s:='big'<br>else y:=y+2;<br>until s<>'big';                           | (9). soob:=true;y:=0<br>for i:=0 to 5 do begin<br>if soob<br>then begin<br>soob:=false; y:=y+5; end;<br>if not(soob)<br>then begin<br>soob:=true; y:=y+2; end;<br>end; |

27. Найдите ошибки в записи серии команд:

|   |  |
|---|--|
| (1). r:=0;<br>for i:=1 to n do<br>k:=k+i end;   | (2). q:=0;<br>for i:=1 to n<br>if q<0 then begin<br>q:=q+i end;                                |
| (3). p:=6;<br>for j:=9 to 8 do begin<br>p+1:=i+1 end; readln('p=',p);   | (4). q:=10;s:='';<br>while q<>0 do<br>s:=q*5-1;<br>q:=q-1;<br>writeln(s);                      |
| (5). n:=0;q:=true;<br>while i<=5 and q do<br>if i mod 2 =3<br>then trunc(i/2)=i/2 then q:=not(q);i:=0;<br>else q:=false;i:=i-1; | (6). readln(a);<br>repeat;<br>a:=a/2;<br>until if a<2;   |
| (7). writeln(q)<br>y:=t-8;<br>q:=y+t-q;<br>until q<y or t>y;  | (8). for i=1 to p/2 do q:=q+2;<br><br>(9). q:=10; q:=(q+25)/5;<br>for p:=1 to q do write('*'); |

|  |   |
|--|---|
| (10). t:='ypa'<br>for k to 1 do<br>t:=t+'!';                               | (11). S:='1';<br>for i:=1 to S do<br>write(S);  |
| (12). S:=0 or S:=1;<br>while (S=0) end (S<'1')<br>begin k:=k+1;<br>S:=S-1; | (13). a:= -1; b:=1;<br>repeat:<br>if a<1 then b:=b-1<br>if a>1 then b:=b+1;<br>until: a>5 |
| (14). for ch:='a' to 'z'<br>S:=ch*2-ch;<br>write(S);                       | (15). while i<7 do<br>begin if i:=1 then<br>i:=4-x else i:='8';                           |

28. Определите, что делает данный фрагмент программы:

|  |   |
|--|---|
| (1). S:=0; n:=0;<br>for i:=1 to 25 do<br>if i mod 3 =0 then begin n:=n+1;<br>S:=S*(n-1)/n+i/n end;   | (2). for i:=1900 to 2000 do<br>if (i mod 4=0) and (i mod 100<>0)<br>or (i mod 400=0) then write(i);   |
| (3). while m<>n do<br>if m>n then m:=m-n else n:=n-m;<br>write(m);   | (4). write('m='); read(m);<br>n1:=1; n2:=1;<br>repeat<br>n:=n1+n2;<br>n2:=n1; n1:=n;<br>until n>m;<br>write(n);   |
| (5). f:=false; i:=1;<br>write('n='); readln(n);<br>repeat<br>write('m='); readln(m);<br>i:=i+1;<br>if m<n then f:=true else n:=m;<br>until f or (i=100);<br>if f then writeln('нет') else writeln('да'); | (6). k:=0; i:=0; f:=true;<br>repeat<br>write('n='); readln(n);<br>i:=i+1;<br>if (n<0) and f then k:=k+1 else<br>f:=false;<br>until (i=50) or not(f);<br>writeln(k); |
| (7). write('n='); readln(n);<br>m:=0;<br>for i:=1 to n-1 do<br>if n mod i=0 then m:=m+i;<br>if m=n then write('да') else write('нет');   |   |

29. Напишите фрагмент программы для вычисления:

- (1) суммы чисел, обратных к первым N натуральным числам;
- (2) суммы квадратов чисел от 1 до n;
- (3)  $n!$ ;
- (4)  $(2n+1)!!$
- (5)  $(2n)!!$
- (6) произведения  $(a-n)(a-2n)\dots(a-kn)$ ;
- (7) суммы  $a^{-1}+a^{-2}+a^{-3}+\dots+a^{-n}$ ;
- (8) суммы  $a^{-1}+a^{-2}+a^{-4}+a^{-8}+\dots+a^{-2^n}$

## Массивы

1. Дайте определение массива.
2. Перечислите три основных свойства табличных величин (массивов).
3. Как описываются массивы на языке Паскаль?
4. Может ли массив содержать 1 элемент? не содержать ни одного элемента?
5. Можно ли во время выполнения программы изменить размер массива (количество элементов в нем)?
6. Верно ли, что тип элементов массива может быть любым?
7. Что такое простой тип? Как определяются новые типы на языке Паскаль? Приведите примеры.
8. Могут ли числа 1, 1.41, 1.73, 2 быть элементами одного массива?
9. Что такое индекс элементов массива?
10. Какие ограничения наложены на индекс?
11. Как происходит обращение к элементу массива?
12. Как подразделяются массивы по количеству размерностей (индексов)?
13. Что такое формальное и фактическое количество элементов массива? Какой из этих параметров больший?
14. Как происходит первоначальное заполнение массива?
15. Какие операции возможны над массивами?
16. Перечислите различные примеры ввода элементов массива.
17. Какие из операций допустимы в Паскале для данных переменных, если есть следующее описание:  
var A, B: array [1..15, 0..8] of real; t: boolean;  
a) A:=B; b) A:=A+B; c) t:=A<>B; d) read(A); e) A[1]:=A[15];  
f) A[2,3]:=B[4][8]+B[1,1];
18. Одинаковы ли типы array [1..15, 0..3] of char и array [1..15] of array [0..3] of char?
19. Есть следующее описание переменных:  
Type line=array [1..20] of char;  
mas1=array [1..10] of line;  
mas2=array [1..15, 1..20] of integer;  
var A: mas1; B: mas2;  
Укажите тип переменных: A, A[3], A[9][18], A[1,1], B, B[15], B[3,3], B[10][16].
20. Укажите синтаксические ошибки в описании массивов:  
(1) Type mas=array [1..20];  
var A: mas;  
(2) var X = array [3..7] of char  
(3) var X= array [integer] of char;  
(4) Type Mark: array [1..24] of integer;  
var X: Mark;  
(5) var mas: array [1..7], [-1..6] of real;  
(6) var B: array [4.9] of real;  
(7) Type Season = array of boolean;  
var X: Season;  
(8) var Y: array [real] of boolean;  
(9) var X: array ['A'..'C', 1..11] of array;
21. Для данного описания укажите случаи, в которых происходит неправильное обращение к элементу массива:  
Type Color = (black, white, yellow, green, blue);  
Mark = array [1..5] of integer;



```

Name = (Alex, Fred, Ann, Mery, Peter);
var A: array [1..12] of Color;
    B: array ['A'..'C'] of array [Color] of integer;
    C: array [true..false] of array [1..15] of Name;
    D: array [Mark] of boolean;

```

- |                     |                      |
|---------------------|----------------------|
| (1) A[green];       | (11) D(5);           |
| (2) A[13];          | (12) D[4, false];    |
| (3) A[2];           | (13) D[true];        |
| (4) B('C', red);    | (14) C[true, Mery];  |
| (5) B[black, 'B'];  | (15) C[Ann, false];  |
| (6) C[15, false];   | (16) C[false, Fred]; |
| (7) B['B', blue];   | (17) C[true, 2];     |
| (8) C[false, Alex]; | (18) D[16];          |
| (9) C[true, 16];    | (19) A[9, yellow];   |
| (10) D[1, true];    | (20) B['A', 'C'];    |

23. Что делает фрагмент программы?

- |   |  |  |
|---|--|--|
| (1). s:=0;<br>for i:=1 to n do<br>if a[i]>0 then s:=s+1;  | (2). for i:=1 to n do<br>if a[i]>0 then a[i]:=-a[i];   | (3). f:=0;<br>for i:=1 to n do<br>if a[i]>10 then f:= 1  |
| (4). s:=0;<br>for i:=1 to n do<br>if a[i]>0 then s:=s+a[i];   | (5). k:=0;<br>for i:=1 to n do<br>if a[i]<0 then k:=i;   | (6). for i:=1 to n do<br>if a[i]>0 then a[i]:=2*a[i]   |
| (7). for i:=1 to n do<br>if a[i]<0 then a[i]:=0;  | (8). f:=1;<br>for i:=1 to n do<br>if a[i]<>0 then f:=f*a[i];   | (9). s:=0;<br>for i:=1 to n do<br>if a[i]<0 then s:=s+a[i];                                      |
| (10). p:=0;i:=1;<br>repeat if a[i]<a[i+1] then<br>p:=p+1;i:=i+1;<br>until i>9;<br>if i-1=p then write('да')<br>else write('нет'); | (11). i:=0;s:=0;<br>while (a[i+1] >=0) and<br>(i<n) do begin<br>s:=s+a[i];i:=i+1;end;<br>writeln(i,s); | (12). i:=n;s:=1;<br>while (a[i] >0) and (i>0) do<br>begin s:=s*a[i];i:=i-1;end;<br>writeln(i,s); |

24. Найдите значение последнего элемента массива после исполнения серии команд:

|                                |                                    |  |
|--------------------------------|------------------------------------|--|
| for i:=2 to 4 do<br>a[i]:=i+2; | for i:=5 downto 1 do<br>a[i]:=i+1; | a[1]:=1; i:=2;<br>repeat a[i]:=a[i-1]+i;<br>i:=i+1 until i>=5; |
|--------------------------------|------------------------------------|--|

25. Укажите значения аргументов, при которых после исполнения указанного фрагмента алгоритма получается данный ответ.

|   |  |
|---|--|
| s:=0;k:=0;<br>for i:=1 to n do s:=s+a[i];<br>s:=s/n;<br>for i:=1 to n do if a[i]>s then k:=k+1;             | ОТВЕТ:<br>1. k=1<br>2. k=n-1<br>3. k=0 |
| k:=1; m:=a[1];<br>for i:=1 to n do if a[i]>m<br>then begin m:=a[i]; k:=1 end<br>else if a[i]=m then k:=k+1; | ОТВЕТ:<br>1. k=1<br>2. k=3             |

|  |  |
|--|--|
| <pre>k:=1; m:=a[1]; for i:=1 to n do   if a[i]&gt;m then begin m:=a[i]; k:=k+1; end;</pre> | <p>ОТВЕТ:</p> <ol style="list-style-type: none"> <li>1. k=1</li> <li>2. k=3</li> </ol> |
|--|--|

26. Укажите смысловые ошибки в записи фрагмента программы. Попробуйте предсказать реакцию машины или привести пример данных, при которых данный фрагмент будет работать неверно.

- |   |   |
|---|---|
| <p>(1). Max:=a[i];<br/>for i:=1 to n do<br/>if a[i]&gt;max then max:=a[i];</p> <p>(3). a[1]:=max;<br/>for i:=1 to n do<br/>if a[i]&gt;max then a[i]:=max;</p> <p>(5). K:=0;S:=0;<br/>FOR i:=1 TO N DO<br/>IF A[i]&gt;=0 THEN K:=K+1;S:=S+A[i]</p> <p>(7). I:=1;<br/>while i&lt;=n do<br/>if a[i]&lt;0 then s:='да' else s:='нет';<br/>writeln(s);</p> | <p>(2). F:=1;K:=0;<br/>FOR i:=1 TO N DO<br/>IF A[i]&lt;&gt;0 THEN K:=K*A[i];</p> <p>(4). k:=0;<br/>FOR i:=1 TO N DO<br/>IF A[i]&lt;&gt;0 THEN<br/>k:=k*A[i];k:=i;</p> <p>(6). max:=a[1]; k:=1;<br/>for i:=1 to n do begin<br/>if a[i]&gt;max then max:=a[i];<br/>end;writeln('max=',a[k]);</p> <p>(8). i:=1;k:=0;<br/>repeat if a[i]*a[i+1]&lt;0 then<br/>k:=k+1;<br/>until i&gt;=n;<br/>writeln('количество<br/>знакоперемен=',k);</p> |
|---|---|

27. Написать фрагмент программы:

- (1). для ввода элементов одномерного массива
- (2). для вывода элементов одномерного массива
- (3). для вычисления суммы элементов одномерного массива
- (4). для вычисления произведения элементов одномерного массива
- (5). для нахождения первого элемента одномерного массива со свойством P
- (6). для нахождения последнего элемента в одномерном массиве со свойством P.
- (7). для подсчета количества элементов со свойством P
- (8). для нахождения минимума в одномерном массиве
- (9). для удваивания максимального элемента в одномерном массиве
- (10). для нахождения количества максимумов
- (11). для нахождения первого отрицательного элемента массива, имеющего четный номер
- (12). для нахождения суммы максимального и минимального элементов
- (13). для нахождения количества положительных элементов, имеющих нечетные номера
- (14). для ввода элементов двумерного массива с клавиатуры
- (15). для вывода элементов двумерного массива в виде матрицы
- (16). для вычисления суммы положительных элементов двумерного массива
- (17). для нахождения количества отрицательных элементов двумерного массива
- (18). для нахождения максимального элемента двумерного массива
- (19). для обнуления минимального элемента двумерного массива
- (20). для вычисления произведения элементов квадратного массива выше главной диагонали

(21). для вычисления суммы элементов квадратного массива ниже побочной диагонали

### Литерные переменные

1. Дайте определение литерной величины.
2. Как описываются литерные переменные на языке TurboPascal?
3. Объясните, какие значения могут принимать строковые величины A, B, C (что общее и в чем различия), если они описаны следующим образом:

```
var A: string;  
    B: string[20];  
    C: string[255];
```

4. Перечислите все операции над литерными переменными в алгоритмическом языке.
5. Перечислите все стандартные операции над литерными переменными в Турбопаскале. Проведите сравнительный анализ операций в Турбопаскале и в ЯША.
6. Объясните, для чего предназначены и как используются стандартные функции обработки строковых величин.
7. Объясните, для чего предназначены и как используются стандартные процедуры обработки строковых величин.
8. Всегда ли справедливы следующие утверждения:
  - а) если с-литера и  $'0' \leq c \leq '9'$ , то с-цифра;
  - б) если с-литера и  $'a' \leq c \leq 'z'$ , то с-строчная латинская буква;
  - в)  $'c' < 'A'$ ;
  - г)  $'0' = 0$ ;
  - д)  $\text{ord}('0') = 0$ ;
  - е)  $'a' = 'A'$ ;
  - ж) если с и d-литеры, то  $c < d$  тогда и только тогда, когда  $\text{ord}(c) < \text{ord}(d)$ ;
  - з)  $\text{ord}(\text{chr}(k)) = k$  для любого целого k;
  - и)  $\text{chr}(\text{ord}(c)) = c$  для любого с-литеры.
9. Дайте формальное описание типа `string[100]` как массива.

10. Объясните, для чего предназначены и как исполняются следующие команды:

- |                                       |                                  |
|---------------------------------------|----------------------------------|
| (1). <code>Q:=a+ba;</code>            | (7). <code>delete(f,3,6);</code> |
| (2). <code>S:=concat(a,b,c,d);</code> | (8). <code>insert(f,r,5);</code> |
| (3). <code>Pos(Q,W);</code>           | (9). <code>chr(n);</code>        |
| (4). <code>Copy(r,11,3);</code>       | (10). <code>ord(f1);</code>      |
| (5). <code>Length(s);</code>          | (11). <code>str(f:5:2,q);</code> |
| (6). <code>Length(a+c);</code>        | (12). <code>val(a,b,x);</code>   |

11. Укажите синтаксические ошибки:

- |                                      |                                     |
|--------------------------------------|-------------------------------------|
| (1). insert('т','top')               | (11). 12) val(123.4567,x,i)         |
| (2). copy('бас',3);                  | (12). 13) ai=delete(1,a,3);         |
| (3). str(string[10]);                | (13). 14) str('123.4567',x);        |
| (4). insert('п','кис','з');          | (14). 15) d:=ord(chr(d))+32);       |
| (5). k:=str('120',n);                | (15). 16) t:=copy(t,1,length(t/2)); |
| (6). insert('3','124','3');          | (16). 17) val('123.4567',x);        |
| (7). ord(25);                        | (17). 18) concat('па','ма');        |
| (8). o:=pos(4,'нога');               | (18). 19) c:=concat(4,'5');         |
| (9). delete('сенокос',5,3);          |                                     |
| (10). c:='3,1415'; delete(c,'14',2); |                                     |

12. Укажите смысловые и синтаксические ошибки:

|   |   |
|---|---|
| (1). if length(a/2)=trunc(length(a/2))<br>then a:=8;<br>else a:=concat(a,a,3);      | (2). for i:=1 to length(i) do begin<br>c:=pos(c,I);   |
| (3). if a mod 2=2 then c:='12345'<br>else c:='12345'-'345';                         | (4). a:='abcdefgh';<br>for i:=1 to 8 do begin<br>a:=insert(a,'12',i);end;                                   |
| (5). s:='1234567890'<br>for i:=2 to lenght(s) delete(s,i,i);                        | (6). i:=1;n:=length(a);<br>wihle i>=n do<br>insert(a,i,'23'); i:=i+1;                                       |
| (7). w:='123.4567';<br>q:=copy(w,4,1); r:=copy(w,1,3);<br>s:=copy(w,5,4); t:=q-r+s; | (8). concat(a,a[pos(a,3)],q);<br>t:=pos(ord(chr(45)), '12345,67890');<br>s:=str('qwert',x,i); val(s:3:2,i); |

13. Укажите значение переменной с после исполнения серии команд:

- (1). c:='гардемарин';  
c:=copy(c,4,3)+'o'+copy(c,length(c)-1,1);
- (2). c:='баскетбол';  
c:=copy(c,3,1)+copy(c,6,1)+copy(c,length(c)-1,2);
- (3). c:='баскетбол';  
c:=copy(c,length(c)-2,3)+copy(c,length(c)-3,1);
- (4). c:='баскетбол';  
c:=copy(c,4,3)+copy(c,2,1);
- (5). c:='баскетбол';  
c:=copy(c,3,2)+copy(c,length(c)-1,1)+copy(c,7,1)+copy(c,2,1);
- (6). c:='графарет';  
c:=copy(c,4,2)+copy(c,2,2);
- (7). c:='графарет';  
c:=copy(c,length(c),1)+ copy(c,length(c)-1,1)+copy(c,3,1)+copy(c,1,2);
- (8). c:='графарет';  
c:=copy(c,2,2)+copy(c,length(c)-2,1)+ copy(c,length(c),1)+copy(c,7,2);  
insert('и',c,4);
- (9). c:='графарет';  
c:=copy(c,4,1)+copy(c,2,2)+'к';
- (10). c:='графарет';  
c:=copy(c,5,length(c)-4)+ copy(c,3,1);  
insert('к',c,1);

- (11). c:='трафарет';  
 c:=copy(c,1,3)+copy(c,length(c),1)+copy(c,3,1);  
 insert('y',c,1);
- (12). c:='сигма';  
 delete(c,2,2);  
 insert('к',c,4);
- (13). c:='информатика';  
 delete(c,1,2);  
 delete(c,7,3);
- (14). c:='работа';  
 delete(c,1,1);  
 insert('з',c,1);
- (15) c:='3.1415';  
 b:=copy(c,1,4);  
 val(b,x,p);  
 str(x:3:1,c);
- (16) c:='1999';  
 insert('21',c,3);  
 val(c,x,p);  
 str(x:4,c);  
 str(x:2,c);
- (17) c:='1803';  
 delete(c,2,2);  
 insert('99',c,2);  
 val(c,x,p);

14. Что делает данный фрагмент программы?

|   |  |
|---|--|
| (1).for i:=1 to length(A) do<br>if copy(A,i,1)='o' then begin<br>delete(A,i,1); insert('a',A,i);<br>end;  | (6).for i:=1 to length(A) do<br>begin c:=copy(A,i,1);<br>insert(c,A,2*i-1);<br>end;                            |
| (2).s:='абракадабра'; i:=1;<br>while i<=length(s) do<br>if copy(s,i,1)='a' then delete(s,i,1)<br>else i:=i+1;   | (7).p:='информатика'; i:=1;<br>while i<=length(p) do<br>if copy(p,i,1)='и' then delete(p,i,1)<br>else i:=i+1;  |
| (3).k:=length(A);<br>if k/2=trunc(k/2)<br>then delete(A,1,1)<br>else delete(A,k,1);   | (8).k:=length(A);<br>if k/2=trunc(k/2)<br>then delete(A,trunc(k/2),2)<br>else delete(A,trunc(k/2),1);          |
| (4).for i:=1 to length(A) do<br>insert(copy(A,i,1),A,2*i-1);  | (9).k:='карактица'; i:=1;<br>while i<=length(k) do<br>if copy(k,i,1)='a'<br>then delete(k,i,1)<br>else i:=i+1; |
| (5).b:=trunc(length(k)/2);<br>for i:=1 to b do<br>k:=copy(k,1,i-1)+copy(k,length(k)-i+1,1)+copy(k,i+1,length(k)-<br>2*i)+copy(k,i,1)+copy(k,length(k)-i+2,i-1); |  |

15. Дополните описание стандартных процедур и функций обработки строковых переменных в соответствии с образцом, данным в пунктах а) и е):

- (1). procedure delete(var a: string; i: integer; j: integer);  
 (2). procedure insert();  
 (3). procedure val();  
 (4). procedure str();  
 (5). function length(a:string):integer;  
 (6). function concat();  
 (7). function copy();

- (8). function pos();
  - (9). function ord();
  - (10). function chr();
16. Используя функции и процедуры обработки литерных данных перепишите команды:
- (1) t:=copy(t,3,length(t)-2); через delete;
  - (2) t:=copy(t,1,2)+y+copy(t,3,length(t)-2); через insert;
  - (3) delete(t,3,1) через copy;
  - (4) insert(b,t,4) через copy.
17. Запишите в виде команды:
- (1) если в слове нечетное число букв, то в конце поставить '\*!';
  - (2) если в слове нечетное число букв, то в начале поставить '!';
  - (3) поставить '!' в конце данного предложения, если 'a' там нет (использовать оператор insert).
18. Напишите фрагмент программы:
- (1). Проверки: является ли слово палиндромом.
  - (2). Подсчета количества слов в предложении.
  - (3). Удаления символов, стоящих на нечетных местах.
  - (4). Вставки после каждого символа слова восклицательного знака.
  - (5). Удаления лишних пробелов между словами предложения.
  - (6). Замены одной точки на многоточие.
  - (7). Проверки: есть ли в слове символы отличные от букв.
  - (8). Обмена местами первого и последнего слов в предложении.
  - (9). Инвертирования каждого слова в предложении.
  - (10). Разбиения слова на слоги.
  - (11). Разбиения слова на склады (Склад – это пара согласная + гласная или одна из нескольких идущих подряд гласных или согласных. Например: О-БО-З-РЕ-НИ-Е).

## Процедуры и функции

1. В каком месте программы и в каком порядке располагаются функции и процедуры? Сравните с алгоритмическим языком.
2. Что такое локальные и глобальные переменные и как они различаются на языке Паскаль? Сравните с алгоритмическим языком.
3. Как выглядит заголовок процедуры на языке Паскаль?
4. Как выглядит заголовок функции на языке Паскаль?
5. Что собой представляет содержательная часть процедуры на языке Паскаль?
6. Что собой представляет содержательная часть функции на языке Паскаль?
7. Как различить по заголовку процедуры на языке Паскаль аргументы, результаты и промежуточные величины?
8. Сравните правила записи процедур на Паскале и на школьном алгоритмическом языке.
9. Сравните правила записи функций на Паскале и на школьном алгоритмическом языке.
10. Сформулируйте правила обращения к функции и к процедуре.
11. Что такое формальные и фактические параметры?
12. Какая взаимосвязь существует между формальными и фактическими параметрами?
13. Какие виды параметров могут быть указаны при описании процедуры или функции в ее заголовке?
14. Для чего используются параметры-значения?
15. Для чего используются параметры-переменные?

16. Найдите синтаксические ошибки в следующих командах:

- |   |   |
|---|---|
| (1). Procedure (var a:real, b,s:char; var d:real);  | (6). 5.function r(a:real;var b;t:real);     |
| (2). Procedure(b[1]+b[2],b[2]+b[3],b[3]+b[1],b[2]); | (7). ...                                    |
| (3). Procedure str(a;b;c:integer; var s:real);      | a. k:=r(A[3],A,r);                          |
| (4). ...  | (8). 6.function g(var a,b:integer):integer; |
| a. str(k[1],k[2],k[3],k[4]);                        | ...   |
| (5). 4.procedure prim(k:real; var p,integer)        | a:=g(a[i],g);                               |
| ...   |   |
| a. prim(a+b,a-b);                                   |   |

17. Перечислите, какие переменные в следующей процедуре являются локальными, глобальными, аргументами, результатами.

- |  |  |
|--|--|
| (1). procedure multiply(d:stroka;var P:real);<br>var i:integer;<br>begin<br>P:=1;<br>For i:=1 to n do<br>if d[i]<>0<br>then P:=P*d[i];<br>end;     | (2). procedure quantity(n:integer;var k:integer);<br>var i:integer;<br>begin<br>k:=0;<br>for i:=1 to n do if d[i]<0 then k:=k+1;<br>end;       |
| (3). procedure poisk(n:integer;var k:char);<br>var i:integer;<br>begin<br>k:='н';<br>for i:=1 to n do if d[i]=13 then k:='д';<br>end;              | (4). procedure summa(n:integer;var s:real);<br>var i:integer;<br>begin<br>s:=0;<br>for i:=1 to n do s:=s+d[i];<br>end;                         |
| (5). procedure maximum(d:stroka;var max:real);<br>var i:integer;<br>begin<br>max:=d[1];<br>for i:=2 to n do<br>if d[i]>max then max:=d[i];<br>end; | (6). procedure maxmod(d:vector);<br>var i:integer;<br>begin<br>max:=abs(d[1]);<br>for i:=2 to n do<br>if abs(d[i])>max then max:=d[i];<br>end; |

18. Написать заголовок следующих процедур и функций:

delete, insert, chr, ord, val, str, copy, length, concat, pos.

19. Перепишите процедуры из пункта 17 в виде функций.

20. Изучите программу PR1 и ответьте на следующие вопросы:

- 1) назовите переменные, локальные для процедуры pr3 и недоступные для процедуры pr2 и основной программы;
- 2) назовите переменные, являющиеся глобальными для процедуры pr3, недоступные в основной программе, но локальные для процедуры pr2;
- 3) назовите переменные, являющиеся глобальными как для процедуры pr2, так и для процедуры pr3.

```

program PR1;
var A, B, C: real;
  procedure pr2;
    var X, Y, Z: real;
      procedure pr3;
        var i, j, k: real;
        begin
          ...
        end;
      begin
        ...
      end;
  begin
    ...
  end;
begin
  ...
end.

```

21. Изучите программу PR и ответьте на следующие вопросы:

- 1) какие переменные, являясь одна локальной, а другая глобальной, имеют одно и то же имя?
- 2) будет ли переменная L иметь одно и то же значение в основной программе и в процедуре pr3?
- 3) доступна ли переменная L, описанная в основной программе, в процедурах pr3 и pr4?
- 4) окажет ли влияние на переменную L, описанную в основной программе, изменение значения переменной L, описанной в процедуре pr3?

```

program PR;
var L, D, F: real;
  procedure pr2;
    var M, N: integer;
    begin
      ...
    end;
  procedure pr3;
    var L, L4: integer;
      procedure pr4;
        var L5: integer;
        begin
          ...
        end;
      begin
        ...
      end;
  begin
    ...
  end;
begin
  ...
end.

```



22. Какой смысл имеет величина k в следующей серии команд

(1). k:=0;

for i:=1 to n do

begin

g:=summamod(a[i]);

if g=0 then k:=k+1

end;

(3). k:=0;

for i:=1 to n do begin

g:=summamod(a[i]);

d:=summa(a[i]);

if g=d then k:=k+1;

end;

(5). k:=0; k:=0;

for i:=1 to n do begin

maximum(a[i],g,ma);

maximummod(a[i],l,mam);

if mam<>ma then k:=k+1;

end;

(2). k:=0;

for i:=1 to n do

begin

g:=summamod(a[i]);

maximummod(a[i],l,ma);

if g=ma then k:=k+1 end;

(4). k:=0;

for i:=1 to n do begin

maximum(a[i],g,ma);

if ma<0 then k:=k+1

end;

(6). k:=0;

for i:=1 to n do

begin g:=summamod(a[i]);

d:=summa(a[i]);

if g<>abs(d) then k:=k+1;

end;

23. Составьте процедуры для:

(1). ввода одномерного числового массива,

(2). вывода одномерного числового массива,

(3). ввода двумерного числового массива,

(4). вывода двумерного числового массива,

(5). нахождения суммы элементов одномерного массива,

(6). нахождения суммы элементов двумерного массива,

(7). нахождения произведения элементов одномерного массива,

(8). нахождения номера максимального элемента одномерного массива,

(9). нахождения номеров максимального элемента двумерного массива,

(10). нахождения номера последнего отрицательного элемента массива,

(11). нахождения номера первого нулевого элемента одномерного массива,

(12). подсчета в данном слове количества вхождений заданной буквы,

(13). определения, есть ли в данном слове две одинаковые буквы подряд.

## Графика

a. Объясните, для чего предназначен модуль GRAPH. Каким образом он подключается к работе, как совместить его работу с модулем CRT?

b. Как инициализируется и выключается графический режим?

c. Объясните, для чего предназначены и как используются стандартные процедуры построения основных элементов графических изображений.

d. Объясните, для чего предназначены и как используются стандартные функции основных элементов графических изображений.

e. Дайте формальное описание типа POINTTYPE как записи.

f. Объясните, для чего предназначены и как исполняются следующие команды:

(1) moveto(50,25);

(2) moverel(-55,15);

(3) t:=getmaxx;

(4) p:= getmaxy div 2;

(17) line(0,0,getmaxx,getmaxy);

(18) lineto(10,30);

(19) linerel(40,-10);

(20) setlinestyle(1,0,3);

- |   |  |
|---|--|
| (5) w:=getx+gety;                       | (21) rectangle(10,20,30,40);           |
| (6) outtext('s');                       | (22) bar(40,30,20,10);                 |
| (7) outtextxy(18,getx,'sin');           | (23) bar3d(0,10,100,120,topon);        |
| (8) closegraph;                         | (24) bar3d(10,100,110,120,topoff);     |
| (9) putpixel(10+getx,100,0);            | (25) p[8].x:=random(getmaxx);          |
| (10) c:=getpixel(15,56);                | (26) p[i].y:=10+i*5;                   |
| (11) circle(100,150,80);                | (27) arc(100,120,0,1.57,50);           |
| (12) sector(15,30,3.14,4.71,10,5);      | (28) drawpoli(10,t);                   |
| (13) floodfill(10+getx,20+gety,8);      | (29) ellipse(150,160,1.57,6.28,50,60); |
| (14) var pol:array[1..25] of pointtype; | (30) pieslice(100,150,0,3.14,70);      |
| (15) sound(1000);                       | (31) fillellipse(100,50,20,30);        |
| (16) delay(2000);                       | (32) nosound;                          |

7. Укажите смысловые и синтаксические ошибки:

|  |   |
|--|---|
| (1) if getx mod 2 = 0<br>then circle(100,-15,80);<br>else ellipse(150,160,50);   | (2) for i:=1 to getmaxx/2 begin<br>p[i].x:=random(10);<br>drawpoly(10,p);end; |
| (3) if getmaxx mod 2=2<br>then circle(100,50.5,35)<br>else arc(100,50,0,10,200); | (4) s:=123;<br>for i:=1 to 8 do<br>outtextxy(10,100+2*(i-1),'s',s+i);         |
| (5) moveto(15,30);<br>moverel(-30,80);<br>outtext(sin(15));                      | (6) i:=1;n:=length(a);<br>while i>=n do<br>insert(a,i,'23'); i:=i+1;          |
| (7) setcolor(20); setlinestyle(1,0,5);<br>triangle(10,20,30,40,50,60);           | (8) bar3d(10,100,110,120,true);   |
| (9) circle(10,10,50);<br>floodfill(10,60,0);                                     | (10) ellipse(150,160,-<br>1.57,6.28,5.0,6);                                   |

8. Что делает данный фрагмент программы?

|  |   |
|--|---|
| (1) for i:=1 to 15 do begin<br>circle(10+(i-1)*2,15+(i-1)*5,15*i);<br>putpixel(10+(i-1)*2,15+(i-1)*5,0);<br>end;   | 4) setcolor(4);<br>setstyleline(2,0,3);<br>for i:=1 to 15 do begin k:=(i-1)*2;<br>rectangle(1+k,1+k,10+k*3,10+k*3);<br>end; |
| (3)i:=1; while i<=15 do<br>begin p[i].x:=random(getmaxx);<br>p[i].y:=random(getmaxy);i:=i+1;end;<br>p[i].x:=p[1].x; p[i].y:=p[1].y<br>drawpoly(16,p);outtext(s); | (3) x:=100;y:=150;moveto(x,y);<br>for i:=1 to 20 do begin<br>linerel(10,10);linerel(-10,10);end;                            |
| (2) x:=100;y:=150;moveto(x,y);<br>for i:=1 to 20 do begin<br>if i mod 2=0 then x:=x-20<br>else y:=y+15;<br>lineto(x,y);end;                                      | (4) setcolor(5);<br>circle(100,100,50);<br>ellipse(100,100,0,6.28,25,50);<br>floodfill(101,101,5);                          |

### Датчик случайных величин

1. Как вы понимаете термин «случайная величина»?
2. Что такое, по-вашему, равномерно распределенная случайная величина?
3. По какому принципу устроен датчик случайных чисел в языке Паскаль?
4. Приведите примеры использования датчика случайных чисел.

5. Каков смысл и порядок употребления оператора randomize?
6. Как получить в программе на Паскале:
  - (1) случайное число от 0 до 1?
  - (2) случайное число от 0 до X?
  - (3) целое случайное число от 0 до M?
  - (4) случайное число от A до B?
  - (5) целое случайное число от N до M?
7. Напишите фрагмент программы, который:
  - (1) Выбирает номера 5 вопросов из 100 (возможно повторение вопросов)
  - (2) Выбирает номера 5 вопросов из 100 разбитых на группы по 20 вопросов (из каждой группы вопросов должен быть выбран только один)
  - (3) Выдает случайным образом номера K выигрышных номеров лотереи при M участниках этой лотереи
  - (4) Перемешивает случайным образом и печатает все элементы массива A
  - (5) Моделирует бросание монеты 100 раз и подсчитывает долю «орлов»
  - (6) Моделирует бросание игральной кости 1200 раз и считает частоту выпадения шестерок
  - (7) Зажигает M случайных точек (пикселей) на экране компьютера
  - (8) Вычисляет примерную площадь единичной окружности методом Монте-Карло, т.е. выбирает в единичном квадрате, куда вписана эта окружность, большое количество точек со случайными координатами и вычисляет долю тех точек, которые попадут внутрь окружности (эта доля и равна площади)
  - (9) Моделирует на экране компьютера броуновское движение, строя ломаную, которая изображает движение точки
  - (10) Моделирует бросание единичного отрезка на клетчатый лист (размеры каждой клетки 1x1) и считает долю случаев, когда отрезок не пересекает линии сетки. Каков, по-вашему, должен быть ответ в этой задаче?
  - (11) Моделирует дискретную числовую случайную величину с заданным распределением (т.е. числом N возможных исходов, их значениями и вероятностями наступления каждого из возможных исходов).

## Множества

1. Какого типа может быть множество?
2. Как ввести множество с клавиатуры?
3. Как выводить множество на экран?
4. Выполните операции:
  - (1) ['C','I','M','N'] \* ['C','M','A','B','H'];
  - (2) ['A','B','C','E'] + ['A','B','F','K'];
  - (3) ['L','M','Z','P','R'] - ['K','M','H','P','Q'];
  - (4) ['A','B','F','K'] + ['K','B','R','M'];
  - (5) [3,5,6,9,2] \* [9,6,5,4] + [5,2,1,6,7];
  - (6) ['n','I','o','b'] + ['I','o'] - ['b','n'];
  - (7) [1,2,6,7,10] - [2,7,5,4,8] + [6,1,25];
  - (8) [ ] + ['1','2'];
  - (9) ([-4,7,2,1,5,3] + [-3,1,3,7,4,2,9]) - ([-4,7,2,1,5,3] \* [-3,1,3,7,4,8,9]);
  - (10) [4,7,8,10,11,15,6] \* [11,7,2,3,1,0,6]\*[6,5,11,3,20];
  - (11) [-2,0,2,1,3,-6] - ([0,2,3,1,6,-5,8,-6] + [-5,3,1,0,6,2]);
  - (12) [5,7,10,26,3,8,7] \* [10,2,1,5,6,27] + [5,7,10,8,7,3] \* [8,3,7,11,15,19] + [10,2,5,1,27,6] \* [8,11,3,7,15,19];

5. В каком случае множество описано неверно:

- |   |   |
|---|---|
| (1) type rn = set of char;                          | (7) type rn = set of '0'..'27';         |
| (2) type rn = set of real;                          | type rn = set of integer;               |
| (3) type rn = set of string[40];                    | (8) type tp = array[6..56] of char;     |
| (4) type tp = real; rn = set of tp;                 | rn = set of tp;                         |
| (5) type rn = set of '0'..'9'                       | (9) type rn = set of boolean;           |
| (6) type tp = array[1..10] of 1..6; rn = set of tp; | (10) type pr = (январь, февраль, март); |
| type rn = set of 1..31;                             | rn = set of pr;                         |

6. Верно ли выполнены операции:

- |   |   |
|---|---|
| (1) [2,5,3,4] * [3,2,8,5,6,1] = [2,5,3];  | (7) [4,6,8,9] * [3,6,7,8] = [8,6];  |
| (2) [7,8,9,11,25] * [3,2,8,9,27] = [8,8,9,9];   | (8) [1,9,7,6] * [7,3,0,1] = [1,7];  |
| (3) ['a','b','c','d','e','f'] + ['a','k','l','m','n'] =<br>['a','a','b','c','d','e','f','k','l','m','n']; | (9) [1,2,9,5] - [2,9,7,6,3] = [1,5];  |
| (4) ['k','l','m','o','p'] + ['m','f','r','z'] =<br>['k','l','m','o','p','f','r','z'];                     | (10) ['A','K','L','M','O'] -<br>['K','L','F','R','Z'] =<br>['A','M','O','F','R','Z']; |
| (5) [1,2,9,20,6] + [4,1,11] = [1,2,9,20,6,4,1,11];  | (11) [] + ['3','4'] = ['0','3','4'];  |
| (6) [5,7,10] * [6,8,10] = [10,10];  |   |

7. Верно ли сравнения:

- (1) ['a'..'z'] < ['a'..'Z'];  
 (2) [1,2,3,4] < ['1','2','3','4','5'];  
 (3) [январь, февраль, март] + [март, май, июнь] > [январь, февраль, март, май, июнь];  
 (4) ([6,7,8,9] + [27..40]) \* [8..30] = [8..30] - [9..27];  
 (5) ([5,3,10,11] + [16..35]) \* [10..20] <> [10..20] - [11..16];  
 (6) [2,3,5] = [5,2,3]

8. Сравнить:

- |   |  |
|---|--|
| (1) A = [6,7,2];<br>B = [2,3,7,8,2,6,1];  | (5) A = [6,9,3,7,14]-[9,14,5,G,4];<br>B = [6,9,11,3,22,7] * [11,5,6,7,9,3]                 |
| (2) A = [8,9,10,11,12]; -[56,17,9,11];<br>B = [8..12];  | (6) A = [1,2,3,4] * [1,5,6,7] -<br>[1,3,7,5,8,9];<br>B = [2,9,1,10,5] * [1,6,2,3] + [4,6]; |
| (3) B = [19..15];<br>A = ([8..20] - [4,6,11,20,14,13]) *<br>[9,25,16,11,7,4,13];  | (7) A = ['a'..'d'];<br>B = ['a'..'f'];   |
| (4) A = ([25..45] + [40..50]) -<br>[32,40,52,26,18,28];<br>B = ([24..31] * [25,27,29,30,31,16,42]) +<br>([33..50] - [40,16,28,51]); |  |

9. Какие операции надо осуществить с множествами B и C, чтобы получить в результате множество A:

- (1) B = ['A','K','L','Z']; C = ['D','L','V','K','F']; A = ['K','L'];  
 (2) B = [1..12]; C = [6..11]; A = C;  
 (3) B = [25..36]; C = [30..40]; A = B;  
 (4) B=['A','K','R','F']; C=['K','H','N','R']; A = ['A','F'];  
 (5) B=['A','K','F','R','Z','N','P','H']; C = ['K','R','N','H']; A = [ ]

10. Какие операции надо совершить с множествами B, C и D, чтобы получить в результате множество A:

|   |   |   |   |
|---|---|---|---|
| (1)<br>B = ['1','2','4','6','8'];<br>C = ['1','4','6'];<br>D = ['2','5','4','8','3'];<br>A = ['8','2']; | (2)<br>B = [1,2,3,4,5];<br>C = [8,9,11,15,20];<br>D = [3,4,9,11];<br>A = D;               | (3)<br>B = ['K','L','M','N'];<br>C = ['H','F','C','R'];<br>D = ['L','C'];<br>A=['K','L','F','M','C',<br>'N','H','R']; | (4)<br>B = ['O','H','Q','K','M','N'];<br>C = ['H','K','F','R','P','Z'];<br>D = ['H','N','A','K','W','O'];<br>A = ['H','K'];   |
| (5)<br>B = [3,8,6,7,10,26,5];<br>C = [8,7,2,5,3,11,12];<br>D = [6,12,26,9,20,13,15];<br>A = [6,12,26];  | (6)<br>B=[5,2,3,4,12,13,20]<br>C = [8,10,12,6,7,2];<br>D = [6,2,15,12,21];<br>A = [2,12]; | (7)<br>B=[7,12,13,25,30,29];<br>C=[13,16,26,30,18,12];<br>D= [5,6,7,12,29,13];<br>A = [30];                           | (8)<br>B=['A','B','C','D','E','F','G'];<br>C = ['B','D','K','F','L','M'];<br>D = ['M','P','Q','Z','A','G'];<br>A = ['C','E']; |

11. В каком случае:

- (1)  $A + B = [ ]$ ;
- (2)  $A - B = [ ]$ ;
- (3)  $A * B = [ ]$ ;
- (4)  $A + B = A$ ;
- (5)  $A - B = A$ ;
- (6)  $A - B = B$ ;
- (7)  $A * B = A$ .

## Записи

1. Что такое запись? В чем ее отличие от других структур данных?
2. Из чего состоит запись? Какого типа могут быть ее компоненты?
3. Как употребляется в программе оператор with?
4. Укажите синтаксические ошибки:

(1) type koor:record  
x,y,z=0..100;end;  
vektor:record  
n,k=koor;end;  
var A:array[1..4]of vektor;  
B:koor; x:integer;

(4) type ent=record  
author,tite:string[50]  
bibl=record  
nom:real;  
kn:ent;

var A:array[1..10] of bibl;  
B:array[1..10] of ent;

(2) type date=record  
day:1..31;  
month:1..12;  
year:1..9999; end;  
rem=record  
mes=array[1..5] of string[9];  
ev=date;end;  
var tod:date; a:day;  
mem:array[1..100] of rem;

(5) type klass=record  
f,im:string[20];end;  
oc=record  
ocenka:array[1..3] of integer;  
f:klass;end;  
var B:array[1..10] of oc;

(3) type com:record  
a,b:real;end;  
plo:record  
c:real

```

d: complex;end;
var a,b:array[1..100] of
com;plo:real;

```

5. Укажите тип переменной, если есть следующее описание:

```

type data=record
    day:1..31;
    month:1..12;
    year:1..9999;
end;
rem=record
    mes:array [1..5] of string[9];
    event:data;
end;
var memos:array[1..100] of rem;
    today:data; k:rem;
    calendar:array[1..400] data;

```

- |                         |                      |
|-------------------------|----------------------|
| (1) today.year          | (7) memos            |
| (2) memos[2]            | (8) k.mes            |
| (3) calendar[200]       | (9) calendar         |
| (4) memos[16].mes[2]    | (10) memos[i].event  |
| (5) k.event.year        | (11) calendar[1].day |
| (6) memos[16].mes[2][1] | (12) rem.event       |

6. Для описания из пункта 5 приведите примеры команд с переменными типа:

- (1) 1..12            (2) data            (3) rem

7. Укажите номера недопустимых операций, если дано следующее описание:

```

type zap=record
    a:integer;
    b:string[25];
    c:array[1..10] of real; end;
kar=record
    p:zap;
    d:array[1..10] of char;end;
var T:zap;k:integer;
    G:array[1..100] of zap;
    O:array[1..100] of kar;

```

- |                                       |                                       |
|---------------------------------------|---------------------------------------|
| (1) for G[i].a=10 downto 1 do k:=k+1; | (7) if O[99].c[6]=1945 then T.a:=200; |
| (2) if G[7].c=56.9 then O.d[4]:='9';  | (8) if K=3 then O.d[4]:='9';          |
| (3) for T.c[2]:=1 to 10 do k:=k+1;    | (9) O[17].p.a:=1990;                  |
| (4) G[5].c:=1990;                     | (10) if G[99].c[6]=1945 then T.a:=20; |
| (5) if K=3 then O[4].d:='9';          | (11) G[17].d:='ошибка';               |
| (6) for T.a:=1 to 10 do k:=k+1;       | (12) G[17].d[i]:='1';                 |

8. Укажите номер заголовка программы, который подходит для решения следующей задачи:

- (1) По успеваемости в сессию (зачеты, экзамены) определить размер назначаемой стипендии у каждого студента группы.

|   |   |  |
|---|---|--|
| I. program stipendiya;<br>type student=record<br>fam:string[20];<br>zachet:array[1..10] of<br>string[9];<br>marks:array[1..5] of real;<br>end;<br>var A:array[1..30] of<br>student; | II. program stipendiya;<br>type student=record<br>fam:string[20];<br>zachet:array[1..10] of<br>string[9];<br>marks:array[1..5] of integer;<br>end;<br>var A:array[1..30] of<br>student; | III. program stipendiya;<br>type student=record<br>fam:string[20];<br>zachet:array[1..10] of<br>string[9];<br>marks:array[1..5] of<br>integer;end;<br>var A,B:student; |
|---|---|--|

(2) По данным заболеваемости на заводах Ярославля определить завод, на котором меньше всего пропущено дней по болезни.

|  |  |   |
|--|--|---|
| I. program zdorow;<br>Type zawod=record<br>Kolrab:integer;<br>Den:integer; end;<br>Var Z:array[1..5] of zawod; | II. program zdorow;<br>type zawod=record<br>naz:string[40];<br>kolrab:array[1..500] of<br>integer;<br>den:integer; end;<br>var Z:array[1..5] of zawod; | III. program zdorow;<br>type zawod=record<br>naz:string[40];<br>kolrab:integer;<br>den:integer; end;<br>var Z:array[1..5] of zawod; |
|--|--|---|

(3) По результатам игр в футбольном турнире определить команду победителя.

|   |  |  |
|---|--|--|
| I. program turnir;<br>Type komand=record<br>Naz:string[15];<br>Rez:array[1..6] of real;<br>Ochki:integer; end;<br>Var K:array[1..7] of<br>komand; | II. program turnir;<br>type komand=record<br>naz:string[15];<br>rez:array[1..6] of string[5];<br>ochki:array[1..6] of real;<br>end;<br>var K:array[1..7] of<br>komand; | III. program turnir;<br>type komand=record<br>naz:string[15];<br>rez:array[1..6] of string[5];<br>ochki:integer; end;<br>var K:array[1..7] of<br>komand; |
|---|--|--|

(4) По данным изменениям валютного курса для ряда стран в течение текущего года определить изменение курса в % для каждой страны.

|  |  |   |
|--|--|---|
| I. program walyuta;<br>Type strana=record<br>Naz:string[20];<br>Dened:real;<br>Mestcurs:string[20]; end;<br>Var W:array[1..12] of<br>strana; | II. program walyuta;<br>type strana=record<br>naz:char;<br>dened:string[20];<br>mestcurs:array[1..12] of<br>real; end;<br>var W:array[1..12] of<br>strana; | III. program walyuta;<br>type strana=record<br>naz:string[20];<br>dened:string[20];<br>mestcurs:array[1..12] of<br>real; end;<br>var W:array[1..12] of<br>strana; |
|--|--|---|

(5) По итоговой турнирной таблице по хоккею определить команду, забившую больше всего шайб.

|   |  |   |
|---|--|---|
| <pre>I.program turnir; type komanda=record naz:string[40]; result:integer; zab,prop:integer;end; var K:array[1..12] of komanda;</pre> | <pre>II.program turnir; type komanda=record naz:string[40]; result:array[1..15] of string[5]; zab,prop:integer;end; var C,K:array[1..12] of komanda;</pre> | <pre>III.program turnir; type komanda=record naz:integer; result:integer; zab,prop:real;end; var K:array[1..12] of komanda;</pre> |
|---|--|---|

9. Напишите имя поля, которое отвечает за:

(1) результат последней игры  $i$ -ой команды, если заголовок программы имеет вид:

```
program turnir;
type komand=record
    naz:string[15];
    rez:array[1..6] of string[5];
    och: integer; end;
var K:array[1..7] of komand;
```

(2) Номер дома последнего человека в списке, если заголовок программы имеет вид:

```
type adr=record
    house:real;
    street:string[20];end;
ank=resord
fam:string[20];
m:adr;end;
var spisok:array[1..10] of ank;
```

(3) год рождения  $i$ -го избирателя, если заголовок программы имеет вид:

```
type date=record
    day:1..31;
    month:1..12;
    year:1..1999;
end;
izbiratel=record
fam:string[30];
rochd:date;
end;
```

```
var uchast:array[1..100] of izbiratel;
```

(4) ординату начала  $j$ -го вектора, если заголовок программы имеет вид:

```
type koor=record
    h,y,z=0..100;end;
vektor=record
nach :koor;
kon:koor; end;
var A:array[1..4]of vektor;
```

(5) количество баллов  $i$ -го студента за 6-ю лабораторную работу, если заголовок программы имеет вид:

```
program upr;
type zacet=record
lab:array[1..10] of integer;
samrab:array[8] of integer;
```



```
konrab: integer; end;  
var группа:array[1..25] of zacht;
```

10. Составьте заголовок программы (типы и переменные) для решения задач:
- (1) Дан список группы, в котором указан размер стипендии, получаемой каждым студентом в каждом из 10 семестров, а также стоимость обучения студентов за каждый год.
  - (2) Дан список группы, в котором указано количество учебных часов пропущенных каждым студентом (в том числе и по уважительной причине) в каждом месяце за семестр.
  - (3) Дана спортивная таблица с указанием количества забитых и пропущенных мячей в каждой игре по футболу.
  - (4) Дан список рабочих, в котором указана начисленная заработная плата в каждом из четырех месяцев.
  - (5) Дан список районов Ярославской области, в котором указан план и фактические показатели надоев молока каждым районом в каждом квартале года.
  - (6) Дана таблица, в которой указано количество человек, прикрепленных к каждому округу, количество кандидатов в каждом округе, количество голосов отданных за каждого кандидата в каждом округе, количество бюллетеней, в которых вычеркнуты все кандидаты.
  - (7) Дана таблица, в которой указаны заводы, количество работающих на них человек, число обращений в поликлинику, количество пропущенных дней по болезни, скольким человекам выдавался больничный.
  - (8) Дана таблица, в которой указано количество лиц подававших заявления ( по категориям) для поступления в ЯГПИ (по факультетам),а так же указано число лиц принятых в ЯГПИ (по категориям и по факультетам).
  - (9) Дана таблица, являющаяся протоколом соревнований прыгунов в длину.(результаты в сантиметрах)
  - (10) Дана таблица, в которой указан курс валюты некоторых стран за определенный промежуток времени.
  - (11) Дана таблица, в которой за несколько лет указано количество студентов поступивших на каждый факультет ЯГПИ и успешно окончивших его через 5 лет.
  - (12) Дана таблица, в которой за несколько месяцев указано количество макулатуры и металлолома, собранных каждым классом.
  - (13) Дана таблица, в которой указана успеваемость студентов группы в данную сессию (фамилия, отметки, зачеты (проставить как + или -), оценки за экзамены).

## Файлы

1. Что такое файл? Какие виды и типы файлов вы знаете?
2. Как описываются файловые переменные в заголовке программы?
3. Какие режимы работы с файлами вы знаете? Как их переключить?
4. Какова реакция машины на неверное имя файла при работе в разных режимах?
5. С какими файлами можно работать и как с файлами прямого доступа и как с файлами последовательного доступа?
6. Какие команды относятся только к одному типу файлов?
7. Как зависит от типа переменной S выполнение команды read(f,s) для текстового файла?
8. В чем разница между командами read(f,s), readln(f,s), read(s) и readln(s) при работе с текстовыми файлами?

9. В чем разница между командами write(f,s), writeln(f,s), write(s) и writeln(s) при работе с текстовыми файлами?

10. Объясните, для чего предназначены и как используются следующие команды:

- |                           |                    |
|---------------------------|--------------------|
| (1) write(f,e);           | (7) filesize(f);   |
| (2) read(s,p);            | (8) seek(f,5);     |
| (3) eof(f);               | (9) writeln(f,s);  |
| (4) reset(w);             | (10) readln(f1,s); |
| (5) rewrite(p);           | (11) close(f);     |
| (6) assign(f,'lab8.pas'); | (12) eoln(f);      |

11. Укажите ошибки:

|  |   |
|--|---|
| <p><b>(1)</b> var f:file of integer;<br/> i:integer;<br/> x:array[1..100] of integer;<br/> begin assign(f,'файл1'); rewrite(f); i:=1;<br/> while not(eoln(f)) do begin read(f,x[i]);<br/> i:=i+1; end;<br/> end.</p> | <p><b>(4)</b> var f:file of real;<br/> i:integer;<br/> x:array of integer;<br/> n:=1;<br/> repeat reset(f); read(f,x[n]);<br/> n:=n+1;<br/> until eof(f); end.</p>                                      |
| <p><b>(2)</b> var f:file of integer;<br/> i:integer;<br/> x:array[1..100] of real;<br/> begin assign(f,'файл1'); reset(f); i:=1;<br/> while eof(f) do<br/> begin read(f,x[n]);<br/> i:=i+1; end;<br/> end.</p>       | <p><b>(5)</b> var f:file of real;<br/> i:integer;<br/> x:array of real;<br/> reset(f); i:=1;<br/> repeat readln(f,x[i]);<br/> until not(eof(f));<br/> end.</p>  |
| <p><b>(3)</b> var f:file of integer;<br/> i,x:integer;<br/> reset(f,'файл1'); i:=1;<br/> while not(eof(f)) do<br/> begin readln(f,x[i]);<br/> i:=i+1; end;<br/> end.</p>   | <p><b>(6)</b> var f:file of integer;<br/> i:integer;<br/> a:array[1..100] of integer;<br/> begin assign(f,'файл1');rewrite(f); n:=1;<br/> repeat<br/> read(f,x[i]);<br/> i:=i+1;<br/> until eof(f);</p> |

12. Что делает данный фрагмент программы?

|   |  |
|---|--|
| <p><b>(1)</b> var f:file of real;<br/> i,j:integer;<br/> begin assign(f,'файл1');<br/> rewrite(f);<br/> for i:=1 to 100 do<br/> begin j:=sqr(i); write(f,j) end;close(f);</p> | <p><b>(4)</b> type t=file of integer;<br/> var f1,f2:t;<br/> a:integer;<br/> begin assign(f1,'файл1');<br/> assign(f2,'файл2');<br/> reset(f1); rewrite(f2);<br/> while not(eof(f1)) do<br/> begin read(f1, a); write(f2,a);end;<br/> a:=100; write(f2,a); end; close(f2);</p> |
|---|--|

|  |   |
|--|---|
| <pre>(2) var f:file of real; s,i:integer; r:real; begin assign(f,'файл1'); reset(f); s:=0; i:=1; while not eot(f) do begin read (f,r); s:=sqr(r)+s; i:=i+1;end;</pre>  | <pre>(5) type t=file of integer; var f1,f2:t; a:integer; begin assign(f1,'файл1');assign(f2,'файл2'); reset(f1); rewrite(f2); while not(eot(f1)) do begin read(f1,a); write(f2,a);end; close(f2);</pre>                 |
| <pre>(3) type t=file of integer; var f1,f2,f3:t; a:integer; begin assign(f1,'файл1'); assign(f2,'файл2');assign(f3,'файл3'); reset(f1); reset(f2); rewrite(f3); while not(eot(f1)) do begin read(f1, a); write(f3,a); end; while not(eot(f2)) do begin read(f2, a); write(f3,a); end; close(f3);</pre> | <pre>(6) type t=file of real; var f1,f2:t; a:real; begin assign(f1,'файл1');assign(f2,'файл2'); reset(f1); rewrite(f2); while not(eot(f1)) do begin read(f1,a); if a&lt;0 then a:=-a; write(f2,a);end; close(f2);</pre> |

13. Составьте процедуры на языке Паскаль для:

- (1) чтения файла записей некоторой структуры в массив записей той же структуры;
- (2) создание файла из 100 случайных целых чисел в диапазоне от 4 до 10;
- (3) записи массива вещественных чисел в файл;
- (4) объединения двух файлов вещественных чисел одинакового размера в один, в котором сначала идут числа из первого файла, а потом из второго;
- (5) объединения двух файлов вещественных чисел одинакового размера в один, в котором числа из этих файлов чередуются,
- (6) выделения из файла вещественных чисел первых двадцати чисел в новый файл;
- (7) выделения из файла вещественных чисел последних двадцати чисел в новый файл;
- (8) добавления одного числа в конец файла целых чисел;
- (9) добавления одного числа в начало файла вещественных чисел;
- (10) разбиения файла на два, один из которых содержит четные компоненты исходного файла, а второй – нечетные;
- (11) разбиения файла на два – из положительных и неположительных компонент;
- (12) создания из числового файла двух новых–из положительных и отрицательных чисел;
- (13) объединения двух упорядоченных числовых файлов в один упорядоченный

## **Тексты программ для выполнения лабораторных работ**

Файл PRIMER1.pas

```
program upr1;
uses crt;
var f,i,o:string[15]; v,g,year:integer;
begin
clrscr;
write('Введите номер текущего года ');readln(year);
write('Введите Вашу фамилию ');readln(f);
write('Введите Ваше имя ');readln(i);
```

```
{ write('Введите Ваше отчество ');readln(o);}
write('Введите Ваш возраст ');readln(v);
write('Введите номер Вашей группы ');readln(g);
```

```
gotoxy (15,15);write('Ваша фамилия - ',f);
gotoxy (1,17);write('Вас зовут ',i);
{ gotoxy (5,18);write('Ваше отчество ',o);}
gotoxy (51,20);write('Вы родились в ',year-v, ' году');
gotoxy (11,25);write('Вы учитесь в ',g, ' группе');
end.
```

#### Файл PRIMER2.pas

```
Begin
clrscr;
soob:='Круговой сектор';
write(soob,' ',R=);readln(r);
write('alpha=');readln(alpha);
s:=r*r*alpha/2;
writeln('S=',S:3:0);
writeln('l=',2*S/r:5:2);
end.
```

#### Файл PRIMER2а.pas

```
program upr4;
uses crt;
var r,s,alpha:real;
soob:string[15];
begin clrscr;
soob:='Круговой сектор';
write(soob,' ',R=);readln(r);
write('alpha=');readln(alpha);
s:=r*r*alpha/2;
writeln('S=',S:9);
writeln('l=',2*S/r:5:2);end.
```

```
program lab6;
uses crt;
var a,i,s,n,k,b,d:integer;
soob:string;
y,x:real;
begin
clrscr;
write('a=');readln(a);
write('b=');readln(b);
{ 2.1 while -> for }
{soob:='no';
i:=2;
while i<=sqrt(a) do begin
if a mod i=0
then soob:='yes';
i:=i+1;
end;
writeln(soob);}
{ 2.2 repeat -> for }
{i:=1;
repeat
i:=i+1;
if a mod i=0
then writeln (i);
until i>a/2;}
{ 2.3 for -> repeat }
{s:=0;
for i:=1 to trunc(a/2) do
if a mod i=0
```

#### Файл PRIMER3.pas

```
program upr4;
uses crt;
var a,b,c:real;
begin
{Соотношения в прямоугольном треугольнике}
write('a=');readln(a);
write('b=');readln(b);
c:=sqrt(a*a+b*b);
write('c=',c);
end.
```

#### Файл PRIMER4.pas

```
Program upr1;
Uses crt;
var b:char;
begin clrscr;
write('input letter ');readln(b);
writeln(b+b+b+b+b+b);
writeln(b,' ',b); writeln(b,' ',b);
writeln(b,' ',b); writeln(b,' ',b);
writeln(b,' ',b);
writeln(b+b+b+b+b+b);
end.
```

#### Файл PRIMER7.pas

```
{ 2.5 repeat -> for }
{k:=0;
repeat
s:=a-trunc(a/10)*10;
a:=trunc(a/10);
k:=(k+s)*10;
until a=0;
writeln (k/10:2:0);}
```

#### { 2.6 while -> for }

```
{y:=a;n:=1;
while n<=6 do
begin x:=y;
y:=(5+x)/2;
n:=n+2;
end;
writeln(y:5:4);}
```

#### { 2.7 for -> while }

```
{for i:=1 to a do
begin
if trunc(a/i)=a/i
then writeln(i);
i:=i+1;
end;}
```

#### { 2.8 while -> repeat }

```
{if b>a
then begin a:=a+b;
b:=a-b;
a:=a-b; end;
```

```

then s:=s+i;
if s=a
then writeln('yes')
else writeln('no');}
{ 2.4 repeat -> for }
{n:=1; k:=10;
repeat
k:=k*10;
n:=n+1;
until a<=k;
writeln(n);}

```

```

writeln('a=',a,' b=',b);
k:=a mod b;
n:=b;
while k<>0 do begin
d:=k;
k:=n mod k;
n:=d;
writeln(n);}
end.

```

## Файл lab10.pas

```

program str_lab;
uses crt;
type str=string[50];
      mass=array[1..20] of str;
var s,s1,s2,s3:str;
      i,j,k,l,n,t:integer;
      x:mass;
{-----}
function poisk(s1,s2:str):str;
var i:integer;k:str;

begin
k:='no';
for i:=1 to length(s2)-length(s1)+1 do
if copy(s2,i,length(s1))=s1
then k:='yes';
poisk:=k;
end;
{-----}
procedure invers(var s1:str);
var i:integer;k:str;
begin
k:="";
for i:=1 to length(s1) do
k:=copy(s1,i,1)+k;
s1:=k;
end;
{-----}
procedure double(var s1:str);
var i:integer;k:str;
begin
k:="";
for i:=1 to length(s1) do
k:=k+copy(s1,i,1)+copy(s1,i,1);
s1:=k;
end;
{-----}
procedure zamena(var s1:str; s2,s3:str);
var i:integer;k:str;
begin
k:="";
for i:=1 to length(s1) do
if copy(s1,i,length(s2))=s2
then begin k:=k+s3; i:=i+length(s2)-1; end
else k:=k+s1[i];
s1:=k;
end;

```

```

{-----}
function quantity(s1,s2:str):integer;
var i,k:integer;
begin
k:=0;
for i:=1 to length(s2)-length(s1)+1 do
if copy(s2,i,length(s1))=s1
then k:=k+1;
quantity:=k;
end;

{-----}
procedure del(var s1:str; s2:str);
var i:integer;k:str;
begin
zamena(s1,s2,"");
end;
{-----}
procedure slova(s:str; var k:integer; var x:mass);
var i:integer;
begin
repeat
zamena(s,' ',' ');
until pos(' ',s)=0;
if s[length(s)]=' '
then delete(s,length(s),1);
k:=0;
repeat
l:=pos(' ',s); k:=k+1;
x[k]:=copy(s,1,l-1); delete(s,1,l);
until pos(' ',s)=0;
k:=k+1;
x[k]:=s;
end;
{-----}
procedure printmass(k:integer; x:mass);
var i:integer;
begin
for i:=1 to k do writeln(x[i]);
end;
begin clrscr;
s2:= '11 52 3 15 467 15';
writeln(s2);
slova(s2,k,x);
printmass(k,x);
end.

```

## Файл lab11.pas

```
program matrix_lab;
uses crt;
type st=array[1..20] of real;
matr=array[1..20] of st;
var n, m, j, i, k, l, r: integer; s,s1,s2,s3,ext:real;
x,y:st; a:matr;
{-----}
function summa(m:integer; x:st):real;
var i:integer; s:real;
begin s:=0;
for i:=1 to m do s:=s+x[i];
summa:=s;end;
{-----}
function summamod(m:integer; x:st):real;
var i:integer; s:real;
begin s:=0;
for i:=1 to m do s:=s+abs(x[i]);
summamod:=s;
end;
{-----}
procedure wwod(var a:matr);
var i,j:integer;
begin clrscr;
write('k-vo strok=');readln(n);
write('k-vo stolb=');readln(m);
for i:=1 to n do begin
for j:=1 to m do begin
write('a['i','j']='');read(a[i,j]);
end; writeln; end; end;
{-----}
procedure wywod;
var i,j:integer;
begin
for i:=1 to n do begin
for j:=1 to m do write(a[i,j]:3:2, ' ');
writeln;end; writeln; end;
{-----}
```

```
program polinoms;
uses crt;
type polinom=array[0..50] of real;
var a,b,c,d:polinom;
n1,n2,n3,n4,i,j,k,n:integer;
f,g,h,t,q,r:real;

procedure input(var n:integer; var p:polinom);
var i:integer;
begin
write('Введите степень многочлена n=');readln(n);
writeln('Введите коэффициенты многочлена,
начиная со старшего. ');
for i:=n downto 0 do readln(p[i]);
end;

procedure output(n:integer; p:polinom);
var i:integer;
begin
```

```
procedure priswaiwanie;
begin n:=3;m:=4; a[1,1]:=3; a[1,2]:=0; a[1,3]:=-5;a[1,4]:=3;
a[2,1]:=-4; a[2,2]:=-1;a[2,3]:=-2; a[2,4]:=-3;
a[3,1]:=1; a[3,2]:=3; a[3,3]:=0; a[3,4]:=8; end;
{-----}
procedure znak(m:integer; var x:st);
var i:integer;
begin for i:=1 to m do x[i]:=-x[i]; end;
{-----}
procedure maximum(m:integer; x:st; var max:real; var k:integer);
var i:integer;
begin max:=x[1];k:=1;
for i:=2 to m do if x[i]>max then begin max:=x[i];k:=i;end; end;
{-----}
procedure maximummod(m:integer; x:st; var max:real; var k:integer);
var i:integer;
begin max:=abs(x[1]);k:=1;
for i:=2 to m do if abs(x[i])>max then begin max:=abs(x[i]);k:=i;end;
end;
{-----}
begin clrscr;
wwod(a); wywod;
for i:=1 to n do
begin s:=summa(m,a[i]); writeln(i,'str. s=',s:3:2); end;
end.
```

## Файл Polinom.pas

```
procedure division(n:integer; p:polinom;
m:integer; q:polinom;
var k:integer; var c:polinom;
var r:integer; var s:polinom);
var i:integer;
begin
if n<m
then begin k:=0;c[0]:=0;
r:=n;
for i:=0 to n do s[i]:=p[i];
end
else
begin
k:=n-m;
for i:=k downto 0 do
begin
c[i]:=p[m+i]/q[m];
for j:=m downto 0 do
p[j+i]:=p[j+i]-c[i]*q[j];
```

```

for i:=n downto 0 do write(p[i]:3:2, ' ');
writeln;
end;

procedure sum(n:integer; p:polinom;
             k:integer; q:polinom;
             var m:integer; var r:polinom);
var i:integer;
begin
if k<n
  then m:=n
  else m:=k;
for i:=0 to m+n-k do r[i]:=p[i]+q[i];
if k<n
  then for i:=k+1 to n do r[i]:=p[i]
  else if k>n
    then for i:=n+1 to k do r[i]:=q[i];
end;

procedure num_mult(n:integer; p:polinom;
                  k:integer;
                  var m:integer; var r:polinom);
var i:integer;
begin
m:=n;
for i:=0 to m do r[i]:=k*p[i];
end;

procedure multiply(n:integer; p:polinom;
                 k:integer; q:polinom;
                 var m:integer; var r:polinom);
var i,j:integer;
begin
m:=n+k;
for i:=0 to m do r[i]:=0;
for i:=0 to n do
  for j:=0 to k do
    r[i+j]:=r[i+j]+p[i]*q[j];
end;

end;
r:=m-1;
for j:=0 to r do s[j]:=p[j];
end;
end;

procedure derivation(n:integer; p:polinom;
                    var m:integer; var r:polinom);
var i:integer;
begin
m:=n-1;
for i:=m downto 0 do
r[i]:=(i+1)*p[i+1];
end;

procedure value(n:integer; p:polinom;
               c:real;
               var m:real);
var i,t:integer;
    b,r:polinom;
begin
i:=1;b[1]:=1;b[0]:=-c;
division(n,p,i,b,t,r,i,b);
m:=b[0];
end;

begin
clrscr;
writeln('Введите первый многочлен');input(n1,a);
writeln(' Введите второй многочлен '); input(n2,b);
division(n1,a,n2,b,n3,c,n4,d);
writeln('Частное равно');
output(n3,c);
writeln('Остаток от деления');
output(n4,d);
writeln('Сумма коэффициентов первого многочлена
равна');
value(n1,a,1,f);writeln(f);
end.

```

### Файл List1.pas

```

program upr1;
type vector = array[1..20] of real;
var m,g,s,f,a,b,c,d,p,q:vector;i,j,k,n:integer;l,w,v,x,y,z:real;

function det2(a,b,c,d:real):real;
function det3(a,b,c:vector):real;
procedure input(n:integer;var a:vector);
procedure output(n:integer;var a:vector);
procedure sum(n:integer;a,b:vector; var s:vector);
procedure subtract(n:integer;a,b:vector;var c:vector);
procedure multiply(n:integer;l:real;a:vector;var q:vector);
procedure system2 ( A,B,C:vector;var x,y:real);
procedure system3 ( A,B,C,P:vector;var x,y,z:real);

```

### Файл List2.pas

```

program upr1;
type vector = array[1..20] of real;
var a,b,c,d,p,q,m,g,f,h:vector;i,j,k,n:integer;s,l,u,w,v,x,y,z:real;

```

```

function det2(a,b,c,d:real):real;
function det3(a,b,c:vector):real;
function scalar(n:integer;a,b:vector):real;
function modulus(n:integer;a:vector):real;
procedure input(n:integer;var a:vector);
procedure output(n:integer;var a:vector);
procedure sum(n:integer;a,b:vector; var s:vector);
procedure subtract(n:integer;a,b:vector;var c:vector);
procedure multiply(n:integer;l:real;a:vector;var q:vector);
procedure angle(n:integer;a,b:vector;var u:real);
procedure vectormult(a,b:vector;var c:vector);

```

### Файл LAB5.pas

```

program upr;          uses crt;
{-----описание типов-----}
type st=string[20];   str=string[8];
ocenka=record   ekz:array[1..6] of integer; zach1,zach2:str;      end;
student=record   fam,im,ot:str;          end;
vedomost=record  ocen:ocenka;   stud:student;   end;
A=array [1..20] of vedomost;
{-----конец описания типов-----}
var X:A;i,j,k,l,n:integer;  S,S1,S2:st;t,c,v,d:real;
{-----процедура вывода-----}
procedure wywod(n:integer;h:A);
{-----процедура присваивания-----}
procedure priswaiwanie(var h:A);

```

### Файл LAB6.pas

```

program upr;    uses crt;
{-----описание типов-----}
type st=string[20];
data=record
day:1..31;          month:st;      year:integer;   end;
adress=record
index,town,street:st;   hause,flat:integer;   end;
pasp=record
fam,im,ot:st;   d:data;  ad:adress;      end;
anketa=record
pas:pasp;          p,nat:st;      end;
A=array [1..20] of anketa;
{-----конец описания типов-----}
var X:A;i,j,k,l,n:integer;  S,S1,S2:st;t,c,v,d:real;
{-----процедура вывода-----}
procedure wywod(n:integer;h:A);
{-----процедура присваивания-----}
procedure priswaiwanie(var h:A);
begin          priswaiwanie(X);          wywod(n,X);          end.

```



## Решение задач в ЭТ Excel

Большинство студентов хорошо справляются с решением задач в электронных таблицах. Они правильно задают имена столбцов и строк, грамотно задают функции. Поэтому мы не будем подробно останавливаться на описании решения задач, лишь приведем правильно оформленную страницу решения.

### Пример 1.

Разберем решение следующей задачи.

Укажите, какое значение будет получено в ячейке Н7 таблицы

|   | G | H                    |
|---|---|----------------------|
| 2 | 3 | =G2                  |
| 3 | 5 | =ЕСЛИ(G3>H2; G3; H2) |
| 4 | 2 | =ЕСЛИ(G4>H3; G4; H3) |
| 5 | 7 | =ЕСЛИ(G5>H4; G5; H4) |
| 6 | 8 | =ЕСЛИ(G6>H5; G6; H5) |
| 7 | 4 | =ЕСЛИ(G7>H6; G7; H6) |

### Решение.

Решение задачи удобнее записывать, продолжив предложенную таблицу.

|   | G | H                    | Вычисления  | Значения в столбце H |
|---|---|----------------------|---|----------------------|
| 2 | 3 | =G2                  | <b>взять значение из ячейки G2</b>  | 3                    |
| 3 | 5 | =ЕСЛИ(G3>H2; G3; H2) | <u>если</u> 5>3 (истина)<br><u>то</u> <b>взять значение из ячейки G3</b><br><u>иначе</u> <b>взять значение из ячейки H2</b><br><u>все</u> | 5                    |
| 4 | 2 | =ЕСЛИ(G4>H3; G4; H3) | <u>если</u> 2>5 (ложь)<br><u>то</u> <b>взять значение из ячейки G4</b><br><u>иначе</u> <b>взять значение из ячейки H3</b><br><u>все</u>   | 5                    |
| 5 | 7 | =ЕСЛИ(G5>H4; G5; H4) | <u>если</u> 7>5 (истина)<br><u>то</u> <b>взять значение из ячейки G5</b><br><u>иначе</u> <b>взять значение из ячейки H4</b><br><u>все</u> | 7                    |
| 6 | 8 | =ЕСЛИ(G6>H5; G6; H5) | <u>если</u> 8>7 (истина)<br><u>то</u> <b>взять значение из ячейки G6</b><br><u>иначе</u> <b>взять значение из ячейки H5</b><br><u>все</u> | 8                    |
| 7 | 4 | =ЕСЛИ(G7>H6; G7; H6) | <u>если</u> 4>8 (ложь)<br><u>то</u> <b>взять значение из ячейки G7</b><br><u>иначе</u> <b>взять значение из ячейки H6</b><br><u>все</u>   | <b>8</b>             |

Заметим, что данная схема расчета предназначена для поиска максимального элемента в столбце G.

**Ответ:** значение ячейки Н7 равно 8

### Пример 2.

Укажите, какое значение будет получено в ячейке Н7 таблицы

|   | G | H |
|---|---|---|
| 2 | 3 | 0 |

|   |   |                        |
|---|---|------------------------|
| 3 | 5 | =ЕСЛИ(G3>G2; H2+1; H2) |
| 4 | 2 | =ЕСЛИ(G4>G2; H3+1; H3) |
| 5 | 7 | =ЕСЛИ(G5>G2; H4+1; H4) |
| 6 | 8 | =ЕСЛИ(G6>G2; H5+1; H5) |
| 7 | 4 | =ЕСЛИ(G7>G2; H6+1; H6) |

### Решение.

Решение задачи удобнее записывать, продолжив предложенную таблицу.

|   | G | H                      | Вычисления   | Значения в столбце H |
|---|---|------------------------|--|----------------------|
| 2 | 3 | 0                      |  | 0                    |
| 3 | 5 | =ЕСЛИ(G3>G2; H2+1; H2) | <u>если</u> 5>3 (истина)<br><u>то</u> взять значение из ячейки H2,<br>увеличенное на 1 {0+1=1}<br><u>иначе</u> взять значение из ячейки H2<br><u>все</u> | 1                    |
| 4 | 2 | =ЕСЛИ(G4>G2; H3+1; H3) | <u>если</u> 2>3 (ложь)<br><u>то</u> взять значение из ячейки H3,<br>увеличенное на 1<br><u>иначе</u> взять значение из ячейки H3<br><u>Все</u>           | 1                    |
| 5 | 7 | =ЕСЛИ(G5>G2; H4+1; H4) | <u>если</u> 7>3 (истина)<br><u>то</u> взять значение из ячейки H4,<br>увеличенное на 1 {1+1=2}<br><u>иначе</u> взять значение из ячейки H4<br><u>Все</u> | 2                    |
| 6 | 8 | =ЕСЛИ(G6>G2; H5+1; H5) | <u>если</u> 8>3 (истина)<br><u>то</u> взять значение из ячейки H5,<br>увеличенное на 1 {2+1=3}<br><u>иначе</u> взять значение из ячейки H5<br><u>Все</u> | 3                    |
| 7 | 4 | =ЕСЛИ(G7>G2; H6+1; H6) | <u>если</u> 4>3 (истина)<br><u>то</u> взять значение из ячейки H5,<br>увеличенное на 1 {3+1=4}<br><u>иначе</u> взять значение из ячейки H5<br><u>Все</u> | 4                    |

Заметим, что данная система команд находит количество элементов в столбце G, которые больше числа из ячейки G2 (в нашем случае G2 содержит число 0, т.е. считается количество положительных чисел).

**Ответ:** значение ячейки H7 равно 4.

### Пример 3.

Укажите, какое значение будет получено в ячейке H7 таблицы

|   | G  | H                     |
|---|----|-----------------------|
| 2 | 3  | =ЕСЛИ(G2<0; 1; 0)     |
| 3 | -5 | =ЕСЛИ(G3<0; H2+1; H2) |
| 4 | -2 | =ЕСЛИ(G4<0; H3+1; H3) |
| 5 | 7  | =ЕСЛИ(G5<0; H4+1; H4) |
| 6 | 8  | =ЕСЛИ(G6<0; H5+1; H5) |
| 7 | -4 | =ЕСЛИ(G7<0; H6+1; H6) |

**Решение.**

Решение задачи удобнее записывать, продолжив предложенную таблицу.

|   | <b>G</b> | <b>H</b>              | <b>Вычисления</b>   | <b>Значения в столбце H</b> |
|---|----------|-----------------------|---|-----------------------------|
| 2 | 3        | =ЕСЛИ(G2<0; 1; 0)     | если 3<0 (ложь)<br>то занести в ячейку H2 число 1<br>иначе занести в ячейку H2 число 0<br>Все                                 | 0                           |
| 3 | -5       | =ЕСЛИ(G3<0; H2+1; H2) | если -5<0 (истина)<br>то взять значение из ячейки H2,<br>увеличенное на 1 {0+1=1}<br>иначе взять значение из ячейки H2<br>Все | 1                           |
| 4 | -2       | =ЕСЛИ(G4<0; H3+1; H3) | если -2<0 (истина)<br>то взять значение из ячейки H3,<br>увеличенное на 1 {1+1=2}<br>иначе взять значение из ячейки H3<br>все | 2                           |
| 5 | 7        | =ЕСЛИ(G5<0; H4+1; H4) | если 7<0 (ложь)<br>то взять значение из ячейки H4,<br>увеличенное на 1<br>иначе взять значение из ячейки H4<br>все            | 2                           |
| 6 | 8        | =ЕСЛИ(G6<0; H5+1; H5) | если 8<0 (ложь)<br>то взять значение из ячейки H5,<br>увеличенное на 1<br>иначе взять значение из ячейки H5<br>все            | 2                           |
| 7 | -4       | =ЕСЛИ(G7<0; H6+1; H6) | если -4<0 (истина)<br>то взять значение из ячейки H6,<br>увеличенное на 1 {2+1=3}<br>иначе взять значение из ячейки H6<br>Все | 3                           |

Заметим, что данная система команд находит количество отрицательных элементов в столбце G.

**Ответ:** значение ячейки H7 равно 3

**Пример 4.**

Укажите, какое значение будет получено в ячейке H7 таблицы

|   | <b>G</b> | <b>H</b>               |
|---|----------|------------------------|
| 2 | 3        | =ЕСЛИ(G2<0; G2; 0)     |
| 3 | -5       | =ЕСЛИ(G3<0; H2+G3; H2) |
| 4 | -2       | =ЕСЛИ(G4<0; H3+G4; H3) |
| 5 | 7        | =ЕСЛИ(G5<0; H4+G5; H4) |
| 6 | 8        | =ЕСЛИ(G6<0; H5+G6; H5) |
| 7 | -4       | =ЕСЛИ(G7<0; H6+G7; H6) |

**Решение.**

Решение задачи удобнее записывать, продолжив предложенную таблицу.

| G | H  | Вычисления   | Значения в столбце H |
|---|----|--|----------------------|
| 2 | 3  | =ЕСЛИ(G2<0; G2; 0)<br><u>если</u> 3<0 (ложь)<br><u>то</u> взять значение из ячейки G2<br><u>иначе</u> занести в ячейку H2 число 0<br><u>все</u>                                      | 0                    |
| 3 | -5 | =ЕСЛИ(G3<0; H2+G3; H2)<br><u>если</u> -5<0 (истина)<br><u>то</u> взять сумму значений ячеек H2 и G3<br>$\{0+(-5)=-5\}$<br><u>иначе</u> взять значение из ячейки H2<br><u>все</u>     | -5                   |
| 4 | -2 | =ЕСЛИ(G4<0; H3+G4; H3)<br><u>если</u> -2<0 (истина)<br><u>то</u> взять сумму значений ячеек H3 и G4<br>$\{(-5)+(-2)=-7\}$<br><u>иначе</u> взять значение из ячейки H3<br><u>все</u>  | -7                   |
| 5 | 7  | =ЕСЛИ(G5<0; H4+G5; H4)<br><u>если</u> 7<0 (ложь)<br><u>то</u> взять сумму значений ячеек H4 и G5<br><u>иначе</u> взять значение из ячейки H4<br><u>все</u>                           | -7                   |
| 6 | 8  | =ЕСЛИ(G6<0; H5+G6; H5)<br><u>если</u> 8<0 (ложь)<br><u>то</u> взять сумму значений ячеек H5 и G6<br><u>иначе</u> взять значение из ячейки H5<br><u>все</u>                           | -7                   |
| 7 | -4 | =ЕСЛИ(G7<0; H6+G7; H6)<br><u>если</u> -4<0 (истина)<br><u>то</u> взять сумму значений ячеек H6 и G7<br>$\{(-7)+(-4)=-11\}$<br><u>иначе</u> взять значение из ячейки H6<br><u>Все</u> | <b>-11</b>           |

Заметим, что данная система команд находит сумму отрицательных элементов в столбце G.

**Ответ:** значение ячейки H7 равно (-11)

**Пример 5.**

Укажите, какое значение будет получено в ячейке H7 таблицы

| G | H  |                           |
|---|----|---------------------------|
| 2 | 3  | =ЕСЛИ(G2>0; G2; -G2)      |
| 3 | -5 | =ЕСЛИ(G3>0; H2+G3; H2-G3) |
| 4 | -2 | =ЕСЛИ(G4>0; H3+G4; H3-G4) |
| 5 | 7  | =ЕСЛИ(G5>0; H4+G5; H4-G5) |
| 6 | 8  | =ЕСЛИ(G6>0; H5+G6; H5-G6) |
| 7 | -4 | =ЕСЛИ(G7>0; H6+G7; H6-G7) |

**Решение.**

Решение задачи удобнее записывать, продолжив предложенную таблицу.

| G | H | Вычисления | Значения в столбце H |
|---|---|------------|----------------------|
|---|---|------------|----------------------|

|   |    |                           |  |    |
|---|----|---------------------------|--|----|
| 2 | 3  | =ЕСЛИ(G2>0; G2; -G2)      | <p><u>если</u> 3&gt;0 (истина)<br/> <u>то</u> <b>взять значение из ячейки G2</b><br/> <u>иначе</u> взять противоположное значение для числа из ячейки G2</p> <p><u>все</u></p>           | 3  |
| 3 | -5 | =ЕСЛИ(G3>0; H2+G3; H2-G3) | <p><u>если</u> -5&gt;0 (ложь)<br/> <u>то</u> взять сумму значений ячеек H2 и G3<br/> <u>иначе</u> <b>взять разность значений ячеек H2 и G3</b></p> <p>{3-(-5)=5}</p> <p><u>все</u></p>   | 8  |
| 4 | -2 | =ЕСЛИ(G4>0; H3+G4; H3-G4) | <p><u>если</u> -2&gt;0 (ложь)<br/> <u>то</u> взять сумму значений ячеек H2 и G3<br/> <u>иначе</u> <b>взять разность значений ячеек H3 и G4</b></p> <p>{8-(-2)=7}</p> <p><u>все</u></p>   | 10 |
| 5 | 7  | =ЕСЛИ(G5>0; H4+G5; H4-G5) | <p><u>если</u> 7&gt;0 (истина)<br/> <u>то</u> <b>взять сумму значений ячеек H4 и G5</b></p> <p>{10+7=14}</p> <p><u>иначе</u> взять разность значений ячеек H4 и G5</p> <p><u>все</u></p> | 17 |
| 6 | 8  | =ЕСЛИ(G6>0; H5+G6; H5-G6) | <p><u>если</u> 8&gt;0 (истина)<br/> <u>то</u> <b>взять сумму значений ячеек H5 и G6</b></p> <p>{17+8=22}</p> <p><u>иначе</u> взять разность значений ячеек H4 и G5</p> <p><u>все</u></p> | 25 |
| 7 | -4 | =ЕСЛИ(G7>0; H6+G7; H6-G7) | <p><u>если</u> -4&gt;0 (ложь)<br/> <u>то</u> взять сумму значений ячеек H6 и G7<br/> <u>иначе</u> <b>взять разность значений ячеек H6 и G7</b></p> <p>{25-(-4)=29}</p> <p><u>все</u></p> | 29 |

Заметим, что данная система команд находит сумму модулей элементов в столбце G.

**Ответ:** значение ячейки H7 равно 29.

### Пример 6.

Опишите решение задачи в Microsoft Excel. Нарисуйте таблицу с указанием имен столбцов и строк, внесите конкретные данные, в ячейках запишите необходимые формулы, вычисления производить не надо.

Дана таблица, содержащая данные об учениках школы, занимающихся в лыжной секции: фамилия, возраст, пол.

Определить

- 1) возраст самого младшего и самого старшего ученика;
- 2) средний возраст учеников
- 3) для каждого ученика найти категорию: возраст 7-10 лет – младшее звено, 11-15 лет – среднее звено, больше 15 лет – старшее звено.

**Решение.**

Достаточно в таблице привести данные о 5 школьниках. Это не обременительно для абитуриента, и позволяет задать значения возраста и пола из всех указанных в условии диапазонов. В качестве верного ответа принимается и одна правильно составленная логическая формула ЕСЛИ() в крайней ячейке с комментарием в каком направлении нужно протянуть формулу.

|          | A                | B              | C   | D   |
|----------|------------------|----------------|-----|---|
| <b>1</b> | Фамилия          | Возраст        | Пол | Категория   |
| <b>2</b> | Иванов           | 17             | м   | =ЕСЛИ(B2>=7;ЕСЛИ(B2<=10;"младшее звено"; ЕСЛИ(B2<=15;"среднее звено"; "старшее звено"));"ошибка") |
| <b>3</b> | Петрова          | 9              | ж   | =ЕСЛИ(B3>=7;ЕСЛИ(B3<=10;"младшее звено"; ЕСЛИ(B3<=15;"среднее звено"; "старшее звено"));"ошибка") |
| <b>4</b> | Сидоров          | 13             | м   | =ЕСЛИ(B4>=7;ЕСЛИ(B4<=10;"младшее звено"; ЕСЛИ(B4<=15;"среднее звено"; "старшее звено"));"ошибка") |
| <b>5</b> | Козлова          | 8              | ж   | =ЕСЛИ(B5>=7;ЕСЛИ(B5<=10;"младшее звено"; ЕСЛИ(B5<=15;"среднее звено"; "старшее звено"));"ошибка") |
| <b>6</b> | Зайцева          | 15             | ж   | =ЕСЛИ(B6>=7;ЕСЛИ(B6<=10;"младшее звено"; ЕСЛИ(B6<=15;"среднее звено"; "старшее звено"));"ошибка") |
| <b>7</b> | Средний возраст  | =СРЗНАЧ(B2:B6) |     |   |
| <b>8</b> | Возраст младшего | =МИН(B2:B6)    |     |   |
| <b>9</b> | Возраст старшего | =МАКС(B2:B6)   |     |   |

**Пример 7.**

Опишите решение задачи в Microsoft Excel. Нарисуйте таблицу с указанием имен столбцов и строк, внесите конкретные данные, в ячейках запишите необходимые формулы, вычисления производить не надо.

Дана таблица результатов соревнований по метанию гранаты: ФИО участника, дата рождения, его результаты (в м) за каждую попытку (всего 3 попытки).

Определите:

- 1) Для каждого человека наилучший результат.
- 2) Средний результат в каждой попытке.
- 3) Для каждого человека разряд: если результат более 40 метров – I разряд, 30-40 метров – II разряд, менее 30 метров – III разряд.

**Решение.**

Замечание: разряд присваивается по лучшему результату.

Так как искомые формулы занимают много места, можно при оформлении решения вынести их в отдельные строки. При этом основную таблицу лучше заполнить данными, которые легко обчисляются устно.

|   | A       | B             | C         | D         | E         | F                | G       | H         |
|---|---------|---------------|-----------|-----------|-----------|------------------|---------|-----------|
| 1 | Фамилия | Дата рождения | 1 попытка | 2 попытка | 3 попытка | Лучший результат | Среднее | Разряд    |
| 2 | Иванов  | 01.12.1988    | 42        | 41        | 43        | 43*              | 42*     | 1 разряд* |

|   |         |            |    |    |    |    |    |          |
|---|---------|------------|----|----|----|----|----|----------|
| 3 | Петров  | 13.08.1987 | 30 | 29 | 31 | 31 | 30 | 2 разряд |
| 4 | Сидоров | 15.03.1988 | 40 | 44 | 42 | 44 | 42 | 1 разряд |
| 5 | Смирнов | 02.10.1986 | 25 | 25 | 28 | 28 | 26 | 3 разряд |
| 6 | Хохлов  | 13.08.1988 | 35 | 33 | 37 | 37 | 35 | 2 разряд |

\* - Приведем формулы в ячейках 2-й строки столбцах F, G и H.

|   | F            | G              | H  |
|---|--------------|----------------|--|
| 2 | =МАКС(C2:E2) | =СРЗНАЧ(C2:E2) | =ЕСЛИ(F2<30;"3 разряд";ЕСЛИ(F2<40;"2 разряд";"1 разряд ")) |

Выделив диапазон формул второй строки (F2:H2) и протянув его до 6-й строки, заполним пустые ячейки таблицы. Так как при написании формул использовались относительные адреса, сдвиг имен в формулах произойдет автоматически.

## **Компьютерная математическая система MathCAD**

Компьютерная математическая система (КМС) – программное средство, среда для выполнения математических расчетов, предоставляющая пользователю возможности выполнения численных расчетов и символьных преобразований.

Возможность выполнения символьных преобразований (выполнения операций с выражениями, содержащими переменные, значения которых не определены) отличает КМС от других программных средств обработки числовой информации. К символьным преобразованиям относятся:

- упрощение выражений, символьное вычисление;
- замена переменной;
- разложение на множители и раскрытие скобок;
- разложение в ряд Тейлора;
- символьное решение уравнений, неравенств, систем уравнений;
- символьное интегрирование и дифференцирование;
- символьные операции с матрицами и векторами.

Кроме того, компьютерные математические системы предоставляют пользователю возможность выполнения численных расчетов (вычисление значения выражений, не содержащих переменные, или выражений, содержащих переменные, значения которых были заданы), использования стандартных математических функций и задания собственных функций, инструменты построения и форматирования графиков различных типов:

- графики функций в прямоугольной декартовой системе координат;
- графики функций в полярной системе координат;
- построение кривых, заданных параметрически;
- графики функций двух переменных.

Вычисления могут быть дополнены текстовыми комментариями.

Некоторые компьютерные математические системы позволяют также использовать программные конструкции, позволяющие писать программы для решения задач, которые невозможно или сложно решить стандартными инструментами КМС.

Одной из распространенных компьютерных математических систем является MathCAD (фирма MathSoft Inc).

MathCAD предоставляет пользователю перечисленные выше возможности выполнения численных расчетов и символьных преобразований, построения графиков различных типов, а также возможности создания математической анимации. Пользователь может определять значения переменных и использовать их в вычислениях, определять собственные операторы и функции. Интерфейс приложения достаточно прост в освоении.

### **Ввод выражений в MathCAD, понятие области**

MathCAD допускает ввод формул и текста в любом месте рабочего документа. Каждое математическое выражение, график или фрагмент текста является отдельным объектом, заключенным в прямоугольную область. В процессе ввода или редактирования выражения видна прямоугольная рамка вокруг области; после завершения редактирования эта рамка скрывается. Однако в процессе вычислений учитывается расположение выражений в областях и тип каждой области: текстовые области рассматриваются как комментарии и в вычислениях не участвуют; выражение, разбитое на несколько областей, будет при вычислении рассматриваться как совокупность отдельных областей даже в случае, если на экране эта совокупность выглядит как единое выражение. Кроме того, если одна область целиком или частично перекрывает другую, невидимая область продолжает участвовать в вычислениях. Таким образом, для вычислений в MathCAD первостепенное значение имеет не то, что пользователь видит на экране, а то, каким образом на самом деле расположены математические выражения в областях.



Чтобы начать ввод в новой области, нужно сначала щелкнуть мышью в свободном месте документа. Появится красный крестик. Затем можно создавать новую область. При наборе математического выражения автоматически создается математическая область, которую можно затем преобразовать в текстовую с помощью ввода символа «пробел» внутри области.






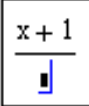
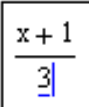
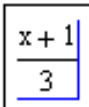
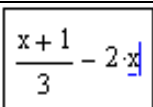
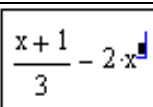
В зависимости от того, где находится курсор, он выглядит по-разному.

1) Крестообразный курсор (визир). Используется для указания расположения новой области. Может появиться только в свободном пространстве между другими областями.

2) Маркер ввода: вертикальная и горизонтальная полоска («уголок»). Используется при вводе и редактировании выражений и текстов. Горизонтальная черта указывает границы активного выражения (к этому выражению будет применена очередная операция), вертикальная – место ввода очередного символа. Нажатие клавиши «пробел» увеличивает границы выделения, нажатие клавиш со стрелками «влево» и «вправо» изменяет положение маркера ввода.

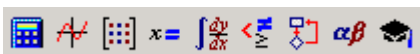
Например, для ввода выражения « $x+1$ » достаточно набрать на клавиатуре « $x+1$ ». При вводе первого же символа выражения крестообразный курсор заменяется маркером ввода; после нажатия на клавиатуре клавиши «+» можно увидеть выделяющую рамку, которая окружает небольшой черный прямоугольник, называемый полем ввода. Поле ввода появляется каждый раз после ввода оператора и содержит места для ввода чисел и выражений. Введенное число замещает поле ввода в выражении.

Можно разбить ввод выражения « $\frac{x+1}{3} - 2 \cdot x^2$ » на шаги:

|   |   |
|---|---|
|    | С помощью крестообразного курсора указываем место ввода выражения (щелчок левой кнопкой мыши).                                      |
|  | Вводим переменную x (появляется маркер ввода).  |
|  | Вводим оператор сложения (появляется поле ввода, на котором находится курсор).  |
|  | Вводим число 1.   |
|  | Нажимаем клавишу «пробел», чтобы увеличить границу выделения.   |
|  | Нажимаем клавишу «/», чтобы ввести черту дроби.   |
|  | Вводим знаменатель дроби  |
|  | С помощью клавиши «пробел» увеличиваем границы активной части выражения.  |
|  | Набирая на клавиатуре « $-2 \cdot x$ », вводим часть выражения  |
|  | С помощью комбинации клавиш Shift+6 (символ ^) или кнопки $x^y$ в арифметической палитре создаем поле ввода для показателя степени. |


|                               |  |
|-------------------------------|--|
| $\frac{x+1}{3} - 2 \cdot x^2$ | <p>Чтобы завершить ввод выражения, нажимаем клавишу Enter или устанавливаем крестообразный курсор в любое свободное место документа (щелчок левой кнопкой мыши по выбранному месту).</p> |
|-------------------------------|--|

При вводе выражения «x+1» все символы, входящие в него, набирались с клавиатуры. Для ввода математических символов, которых нет на клавиатуре, можно использовать различные комбинации клавиш, но обычно удобнее пользоваться специальной панелью инструментов «Математика». Кнопки этой панели открывают палитры символов, служащие для вставки операторов, греческих букв, шаблонов графиков и т.п. Они значительно облегчают ввод выражений в MathCAD.




Список палитр:

 Арифметические операторы.

 Шаблоны графиков функций.

 Матричные и векторные операции.

 Операторы присваивания, численного и символьного вычисления.

 Производные, интегралы и пределы. Ряды и произведения.

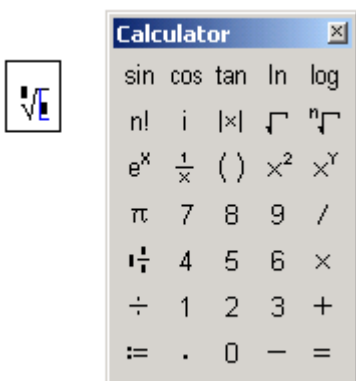
 Знаки отношения. Логические операции.

 Программные структуры.

 Греческие буквы.

 Символьные вычисления.

При выборе одной из кнопок палитры символов в документ вставляется отдельный символ или шаблон выражения с полями ввода, которые необходимо заполнить:



С помощью палитры арифметических операторов можно вводить операторы сложения, вычитания, умножения, деления, извлечения квадратного корня и корня n-ой степени, возведения выражения в степень, комплексно сопряженного, абсолютной величины, факториала, вводить символ бесконечности, константы e, i, π, функции синус, косинус, тангенс, десятичный логарифм, а также вводить символы цифр, десятичной точки и скобок. С помощью

палитры операторов высшей математики можно вводить операторы для вычисления сумм и произведений, оператор суммирования элементов вектора и нахождения произведения элементов вектора, оператор производной (предназначенный для нахождения численного значения производной функции в точке), оператор нахождения производной n-го порядка, определенного и неопределенного интеграла, предела. Есть также специальная палитра для ввода греческих букв.

Без помощи палитр, пользуясь только клавиатурой, удобно вводить символы, обозначения которых прямо указаны на клавиатуре (цифры, скобки, оператор сложения, вычитания и т.п.), а также символы, комбинации клавиш для которых легко запоминаются. Например, для введения оператора умножения можно набрать на клавиатуре символ «звездочка» (\*), а для оператора возведения в степень - символ ^, которые используются в тех же целях в различных языках программирования; для введения оператора деления - косую черту (/), для знака корня квадратного - обратную косую черту (\), которая является частью графического изображения знака квадратного корня.

При вводе оператора необходимо обращать внимание на положение курсора. Если курсор имеет вид маркера ввода, оператор будет введен в том месте, где находится курсор, и будет действовать на число или выражение, стоящее непосредственно перед ним (учитывая порядок следования операций). Например, если было набрано выражение «b·c» и маркер ввода находится после символа «с», то, вводя оператор деления или возведения в степень, мы увидим, что он действует только на переменную с, а оператор сложения или вычитания – на все выражение b·c.

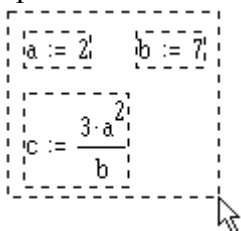
Если горизонтальная синяя черта охватывает некоторое выражение, то вводимый оператор будет действовать на все это выражение. Например, для того, чтобы извлечь квадратный корень из выражения a·b+c·d, нужно установить выделение (синюю горизонтальную черту) таким образом, чтобы подчеркивалось все выражение a·b+c·d, и ввести оператор извлечения квадратного корня. Для изменения размеров и положения выделяющей рамки можно пользоваться клавиатурой (клавиши со стрелками, «пробел»).

Кроме математических выражений, в MathCAD можно также осуществлять ввод текста. Для создания текстовой области нужно нажать на клавиатуре клавишу «кавычка». Можно также преобразовать математическую область в текстовую, введя внутри области символ «пробел». Математические и текстовые области имеют разный стиль оформления, в том числе различный шрифт символов. Если набираемый текст отображается в виде нечитаемой последовательности символов, необходимо изменить вид шрифта с помощью панели форматирования (например, использовать шрифт MS Sans Serif). Ввод текста в текстовой области производится так же, как в простейших текстовых редакторах. Курсор в текстовой области также принимает вид, стандартный для графических редакторов. Чтобы выйти из текстовой области, нужно щелкнуть мышью в свободном месте рабочего документа.

## Действия с областями

Как уже говорилось, каждое математическое выражение, график или фрагмент текста является отдельной областью.

Чтобы выделить область, нужно нажать левую кнопку мыши и, не отпуская ее, переместить мышью, чтобы заключить область внутри пунктирного прямоугольника. Как только прямоугольник охватит все, что необходимо, кнопку мыши отпускают. На рисунке показан процесс выделения трех областей:



Если область выделена, ее можно перемещать, копировать, удалять (аналогично тому, как это делается в графическом редакторе). В режиме перемещения курсор принимает вид ладони. Для перемещения выделенных областей нужно поместить курсор над одной из них; для перемещения отдельной области – навести курсор на границу этой области.

Для того, чтобы вставить пустые строки между областями, используется клавиша Enter.

Отдельную область можно пометить цветом с помощью команды меню Format | Properties, установив флажок Highlight Region (можно также выбрать цвет подсветки). В том же диалоговом окне Properties на вкладке Calculation можно отключить вычисления в выбранной области, установив флажок Disable Evaluation. Снятие флажка снова включит область в вычисления.

## Вычисление значений выражений

Введя некоторое выражение, можно узнать его численное значение или выполнить символьное преобразование. Для выполнения численного расчета после выбора области, содержащей выражение, нужно нажать на клавиатуре клавишу со знаком равенства (=). При этом выражение не обязательно должно быть выделено целиком, но курсор должен находиться в пределах данного выражения (в виде маркера ввода или выделяющей рамки).


После того, как выражение выделено и введен знак равенства, MathCAD автоматически вычислит значение выражения. Это позволяет использовать MathCAD в качестве калькулятора. Знак « $\Rightarrow$ » можно использовать только для вычислений, не содержащих переменных, значения которых не определены. Для символьных преобразований используется другой оператор, вставляемый с помощью панели инструментов: « $\rightarrow$ ». В частности, этот знак используется при нахождении производной, неопределенного интеграла, пределов и т.д. Попытка вычислить численно значение выражения, содержащего переменные, значения которых не были определены, приведет к сообщению об ошибке.

## Переменные в MathCAD

К именам переменных в MathCAD, как правило, предъявляются требования, аналогичные требованиям к записи идентификаторов в языках программирования (в некоторых версиях MathCAD допускается использование в именах переменных символов кириллицы). При этом регистр символов имеет значение (заглавные и строчные буквы в именах переменных и функций считаются различными символами). Типы переменных в документе MathCAD определяет автоматически.

Для того, чтобы использовать переменную в численных расчетах, необходимо определить ее значение (это не относится к предопределенным переменным, значения которых известны по умолчанию, например, переменные  $\pi$ ,  $e$ , а также к переменным, используемым в символьных вычислениях).

Для определения значения переменной вместо обычно используемого в математике знака равенства применяется символ присваивания « $:=$ ». Это обстоятельство связано с тем, что знак равенства интерпретируется в математических выражениях по контексту. Например,  $x = y$  означает либо присвоение переменной  $x$  значения ранее определенной переменной  $y$ , либо просто факт логического равенства значения  $x$  значению  $y$ .

Такая двойственность недопустима в машинных программах. Поэтому в системе MathCAD знак  $=$  используется как знак вывода результатов вычислений, а для присваивания переменным значений используется знак  $:=$ . Наконец, для обозначения отношения величин  $x$  и  $y$  как равенства используется жирный знак равенства  **$=$** . Эти символы можно ввести с помощью клавиатуры или с помощью панели инструментов (для открытия палитры операторов отношений служит экранная кнопка , палитры операторов присваивания и вычисления – кнопка  $^*:=$ ).

Ввод знака присваивания может также осуществляться с клавиатуры. Для этого используется клавиша с двоеточием (:). Таким образом, даже не зная комбинации клавиш для ввода

различных символов, можно ввести знак присваивания интуитивно, так как двоеточие - начальная часть в записи оператора присваивания.

При выполнении численных расчетов, в которых используется команда присваивания, необходимо особое внимание обратить на взаимное расположение областей в рабочем документе. MathCAD «читает» рабочий документ, как книгу: сверху вниз и слева направо. Определив переменную локально, ее можно использовать в вычислениях везде ниже и правее выражения, определяющего переменную (координаты области определяются как координаты так называемой точки привязки, которая находится чуть ниже левого верхнего угла области). Если переменная используется в документе раньше, чем она была определена, при выполнении численных расчетов, MathCAD рассматривает это, как ошибку, выделяя переменную цветом.

Пример:

| Неправильное расположение выражений        | Правильное расположение выражений            |
|--|--|
| $a + 1 =$<br>$a := 1$                      | $b := 2$<br>$b - 3 = -1$                     |
| $\sin(\alpha) =$ $\alpha := \frac{\pi}{2}$ | $\beta := \frac{\pi}{6}$ $\sin(\beta) = 0.5$ |

Для того, чтобы исправить подобную ошибку, нужно изменить расположение математических выражений так, чтобы переменная была определена ранее, чем она впервые используется.

Существует также команда глобального присваивания. Знак глобального присваивания  $\equiv$  вставляется с помощью панели инструментов. При использовании глобального присваивания заданное значение переменной может использоваться в любом месте документа, в том числе выше места присваивания.

## Определение дискретного аргумента

Для выполнения повторяющихся вычислений, в том числе для нахождения значений функции, MathCAD использует специальный тип переменных - дискретные аргументы. Переменная типа дискретный аргумент принимает диапазон значений, например, все числа от -10 до 10 с шагом 0,1.

Для определения дискретного аргумента необходимо выполнить следующее:

| Действие  | Пример           |
|---|------------------|
| 1) Ввести имя переменной и знак присваивания.   | t:=              |
| 2) Ввести первое число диапазона.   | t:=-10           |
| 3) Ввести запятую и второе число диапазона (тем самым определяется шаг).  | t:=-10,-9.9      |
| 4) Напечатать на клавиатуре точку с запятой (;), при этом MathCAD добавит в документ два символа точки подряд (нельзя просто ввести с клавиатуры два символа точки!).   | t:= -10,-9.9..   |
| 5) Ввести последнее число диапазона (в действительности дискретный аргумент может и не принимать значения, в точности равного этому числу, так как данное число определяет только верхнюю границу диапазона). | t:= -10,-9.9..10 |

Величина шага может быть равна любому действительному числу. В случае, если шаг равен единице, третий шаг алгоритма можно опустить. Например, если дискретный аргумент должен принимать все целые значения от 10 до 20, команда присваивания будет выглядеть так: «t:=10..20».

Если включить переменную данного типа в какое-либо выражение, MathCAD вычисляет значение этого выражения для всех значений дискретного аргумента, входящих в диапазон.

## Определение функции

Еще большей гибкости вычислений можно добиться, используя определения функций. Оно осуществляется аналогично определению переменных. При этом все переменные, входящие в выражение функции, должны быть определены заранее.

Пример определения функции:

$$h(t) := 1600 + \frac{a \cdot t^2}{2}$$

Когда функция определена, можно ее использовать в вычислениях.

Пример: вычислим значение определенной нами функции при  $t=12.5$ , для этого подставим вместо  $t$  данное число.

$$h(12.5) = 834.375$$

Чтобы вычислить значение функции для каждого значения  $t$  из диапазона, определенного ранее, достаточно ниже определения функции ввести « $h(t)=$ ». MathCAD выведет таблицу значений функции.

При использовании функции допускается использование в качестве аргумента переменных, отличных от используемой при определении функции. Например, если была задана функция  $h(t)$ , в вычислениях могут использоваться выражения, включающие  $h(x)$ ,  $h(y^2)$  и т.д.

## Построение графиков функций

MathCAD может строить двумерные графики в декартовых и полярных координатах, картины линий уровня, изображать поверхности и выводить ряд других трехмерных графиков. Все они располагаются в отдельных графических областях документа. Для их создания применяется либо пункт меню Вставка/График, либо палитра построения графиков, для вызова которой нужно выбрать экранную кнопку с изображением графика функции:



Рассмотрим создание простого двумерного графика, отображающего функцию  $h(t)$ , введенную в предыдущем разделе.

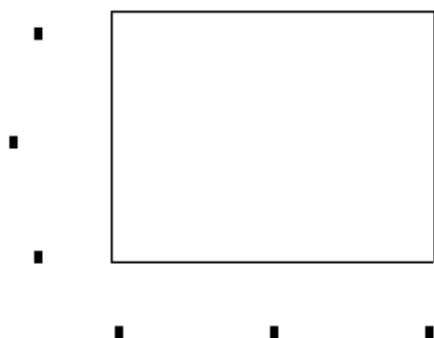
Чтобы создать график в MathCAD, нужно с помощью мыши указать место, в которое будет добавлена графическая область, и выбрать соответствующую кнопку палитры. Для двумерного графика это кнопка палитры графиков с изображением двумерных графиков функций:



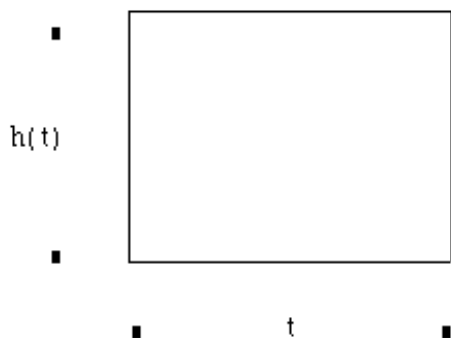
После того, как нажата экранная кнопка, появляется пустой график с полями ввода для выражений, отображаемых по осям графика. Множества точек, из которых состоит график, определяются дискретными аргументами: MathCAD строит одну точку графика для каждого значения дискретного аргумента, задающего график.

Пример построения графика функции  $h(t)$ :

1) После того, как визир (крестообразный курсор) установлен в нужное место и выбрана экранная кнопка, соответствующая декартовой графике, MathCAD создаст пустую графическую область.

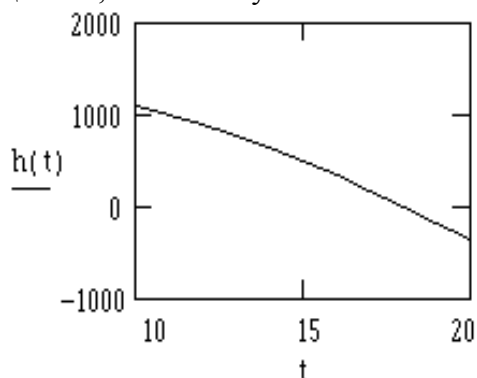


2) В поле ввода под ось абсцисс нужно ввести имя переменной (в примере это  $t$ ), поставив таким образом в соответствие этой оси переменную; в поле ввода напротив середины оси ординат ввести выражение функции. Если функция была определена ранее, вместо выражения можно использовать обозначение, например,  $h(t)$ .



3) Оставшиеся поля ввода предназначены для ввода максимального и минимального значений, откладываемых на оси. Если оставить их пустыми, MathCAD автоматически заполнит их при создании графика.

После щелчка вне графика MathCAD вычисляет и строит точки графика. Под  $h(t)$  появляется образец линии, которой нарисован график. Это помогает идентифицировать различные кривые, если на одно поле выводятся несколько графиков (т.к. различные графики строятся различным цветом, также могут использоваться штриховые и пунктирные линии).



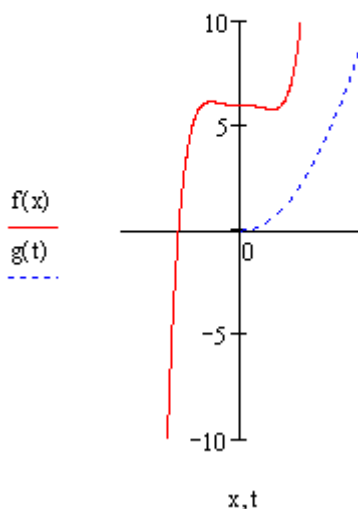
При построении графика функции в большинстве случаев можно не задавать значения аргумента; в этом случае MathCAD самостоятельно определит его значения.

При построении в одной графической области нескольких графиков, отображающих различные зависимости, необходимо, выделив первое выражение для оси ординат, набрать на клавиатуре запятую (,). Непосредственно под первым выражением появится пустое поле, в которое вводится выражение следующей функции. При этом все выражения должны использовать один и тот же аргумент. Если необходимо использовать различные аргументы для графиков разных функций, они перечисляются в поле ввода аргументов через запятую, при этом количество выражений функций и аргументов должно совпадать, а порядок, в котором располагаются выражения функций, должен соответствовать порядку следования аргументов.

$$f(x) := x^5 - x^3 + 6$$

$$t := 0, 0.1 \dots 4$$

$$g(x) := x^2$$

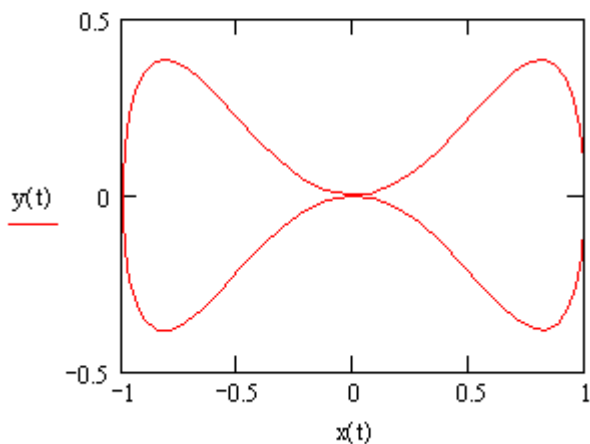


С помощью того же шаблона графика можно строить кривые, заданные параметрически. При этом функции  $x(t)$  и  $y(t)$  записываются в полях ввода оси абсцисс и оси ординат соответственно:

$$t := 0, 0.1 \dots 2 \cdot \pi$$

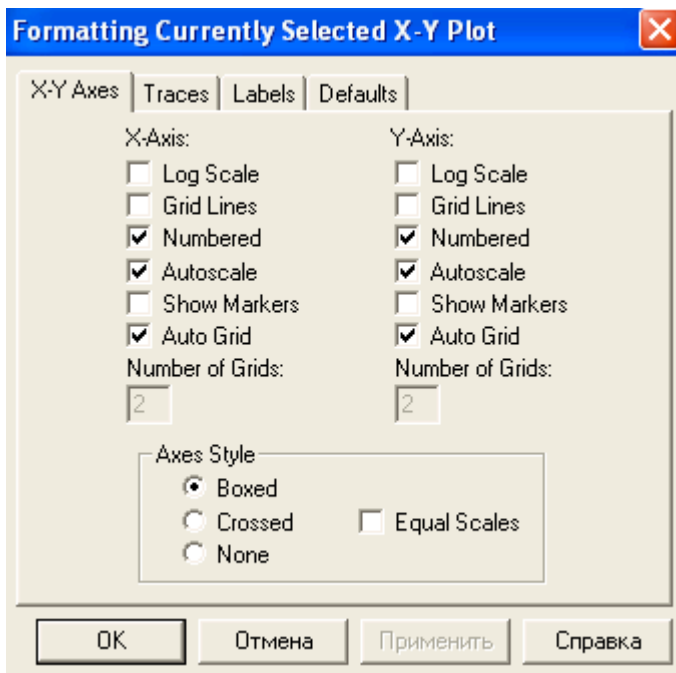
$$x(t) := \cos(t)$$

$$y(t) := \sin(t) \cdot \cos(t)^2$$



При выделении графической области становятся доступными для редактирования поля ввода аргументов и выражений функций, поля ввода максимальных и минимальных значений по осям, отображаемых на графике. В режиме редактирования графической области можно также изменять вертикальный и горизонтальный размеры области (при этом будет изменяться масштаб; максимальное и минимальное значения на каждой оси останутся неизменными). Другие свойства графика (формат осей, линии сетки, формат линий графиков и т.п.) можно изменить с помощью диалогового окна, которое вызывается двойным щелчком по области графика.

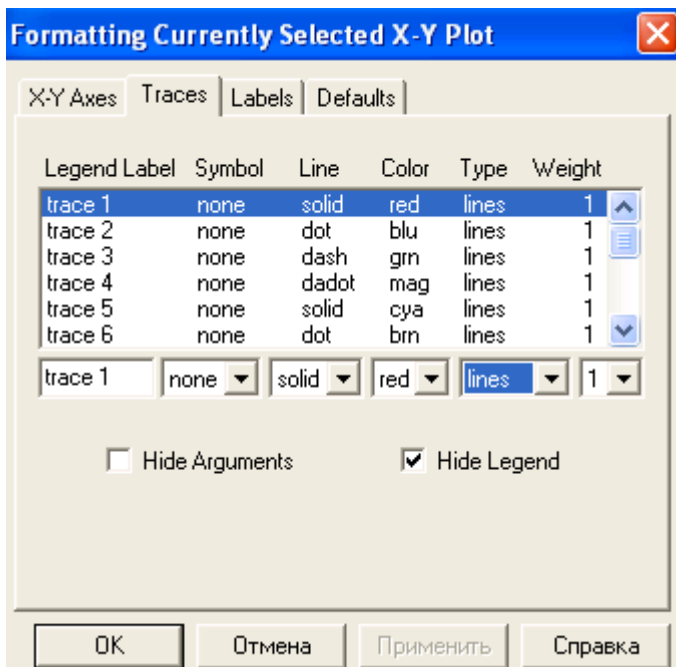




Вкладка X-Y Axes позволяет:

- добавить нумерацию осей (флажок Numbered);
- изменить количество отрезков, на которые разбивается ось (поле Number of Grids становится доступным при снятии флажка Auto Grid);
- добавить линии сетки (Grid Lines);
- изменить стиль осей:

Crossed – оси, пересекающиеся в начале координат  
 Boxed – рамка  
 None – отсутствие осей



Вкладка Traces позволяет изменить стиль линий графика. Порядок записи линий в диалоговом окне соответствует порядку записи выражений функций в области графика. Поле Legend Label показывает метку графика (по умолчанию она не отображается в документе), поле Symbol – тип символа, которым обозначается каждая точка графика, поле Line – стиль линии (solid – сплошная, dot – пунктирная, dash – штриховая, dadot – штрихпунктирная); поле Color – цвет линии, поле Weight – толщину линии. В поле Type отображается тип графика (по умолчанию

это lines – обычный график; можно использовать и другие типы, например, в типе points отображаются только те точки графика, значения абсциссы которых точно известны (например, если при построении графика используется дискретный аргумент), в типе bar значения ординат точек будут отображаться в виде столбчатой диаграммы.

Чтобы изменить свойства линии одного из графиков в графической области, необходимо выбрать линию графика в таблице. Значения свойств изменяются путем выбора новых значений из раскрывающегося списка под таблицей.

## Построение графиков функций в полярной системе координат

Для построения графиков функций в полярной системе используется шаблон графика в полярной системе координат, для вставки которого используется специальная кнопка палитры графиков:



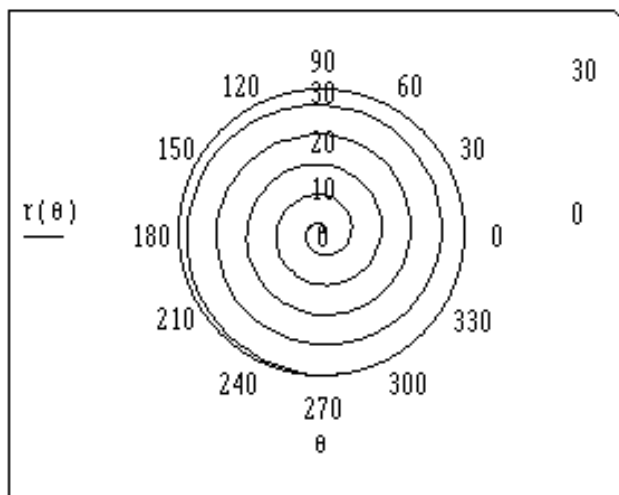
Для построения графика функции в полярной системе координат нужно:

- 1) Выбрать место построения графика и вставить шаблон графика. При этом выше области графика должен быть определен угол - дискретный аргумент, а также функция угла. После выбора кнопки с изображением полярного графика MathCAD показывает круг с четырьмя полями ввода.
- 2) В поле ввода внизу ввести переменную (величина угла) или любое выражение, включающее эту переменную.
- 3) В поле ввода слева ввести выражение для радиуса.
- 4) В два поля ввода справа занести максимальное и минимальное значения радиуса. MathCAD заполняет эти поля по умолчанию. Если требуется, можно изменять эти пределы.

Пример: построение графика функции  $r(\theta) = \theta$ .

$$\theta := 0, 0.1 .. 30$$

$$r(\theta) := \theta$$



Если дискретный аргумент не был определен перед построением графика в полярной системе координат, по умолчанию MathCAD задает значения аргумента от 0 до  $2\pi$ .

## Встроенные функции

В MathCAD есть большое число встроенных функций. Для того, чтобы вставить встроенную функцию, удобно использовать экранную кнопку на панели инструментов: кнопка с надписью « $f(x)$ ».

Некоторые из встроенных функций MathCAD:

| Функция | Возвращает... |
|---------|---------------|
| acos(x) | Арккосинус.   |
| asin(x) | Арсинус.      |

|                    |   |
|--------------------|---|
| atan(x)            | Арктангенс.   |
| aeil(x)            | Наименьшее целое, большее или равное x.   |
| cos(z)             | Косинус.  |
| cosh(z)            | Гиперболический косинус.  |
| cot(z)             | Котангенс.  |
| coth(z)            | Гиперболический котангенс.  |
| csc(z)             | Косеканс.   |
| exp(z)             | Экспонента $e^z$ .  |
| floor(x)           | Наибольшее целое, меньшее или равное x.   |
| if(cond, x, y)     | Возвращает значение x, если cond отличен от 0 (истина).<br>Возвращает значение y, если cond равен (ложь). |
| ln(z)              | Натуральный логарифм z.   |
| log(z)<br>log(z,b) | Логарифм z по основанию 10.<br>Логарифм z по основанию b.   |
| mod(x, modulus)    | Остаток от деления x на modulus. Результат имеет тот же знак, что и x.                                    |
| rnd(x)             | Возвращает равномерно распределенное случайное число между 0 и x.   |
| sec(z)             | Секанс.   |
| sin(z)             | Синус.  |
| sinh(z)            | Гиперболический синус.  |
| tan(z)             | Тангенс.  |
| tanh(z)            | Гиперболический тангенс.  |

Так как заглавные и строчные буквы при записи математических выражений в MathCAD считаются разными символами, необходимо при записи имен функций обращать внимание на регистр символов.

В приведенной таблице x означает вещественное число, z - число, вещественное либо комплексное. Все углы измеряются в радианах.

Кроме приведенных функций, в MathCAD есть встроенные функции для работы с векторами и матрицами, с комплексными числами, статистические функции, функции для работы с файлами и другие.

При работе с функциями в MathCAD необходимо обратить внимание на то, что многозначные функции и функции с комплексным аргументом всегда возвращают главное значение.

## Комплексные числа

MathCAD воспринимает комплексные числа в форме  $a+bi$ , где a и b - действительные числа. При этом для ввода мнимой единицы используется сочетание символов li (символ i, используемый по отдельности, будет считаться именем переменной или частью имени переменной). Когда курсор покидает выражение, содержащее li, MathCAD скрывает избыточную единицу.


В MathCAD есть следующие специальные функции и операторы для работы с комплексными числами:

Re(z) - вещественная часть z;

Im(z) - мнимая часть z;

$\arg(z)$  - угол в комплексной плоскости между вещественной плоскостью и вектором  $z$ ,  
 $-\pi \leq \arg(z) \leq \pi$ ;  
 $|z|$  - модуль  $z$ .

## Решение уравнений и неравенств

MathCAD может решать уравнения и неравенства как численно, так и символично. В записи уравнений используется специальный знак отношения равенства  $=$ ; его, как и символы  $\leq$ ,  $\geq$ , можно вставить в документ с помощью палитры отношений, вызываемой кнопкой .

Чтобы найти корни уравнения, нужно выделить в любом его месте переменную, относительно которой уравнение решается; затем выбрать в меню «Символика» пункт Variable | Solve – решить относительно переменной. Аналогично решаются неравенства: в неравенстве выделяется переменная, относительно которой решается неравенство, выбирается команда меню Variable | Solve; при этом решение имеет вид ограничений. Например, при символьном решении неравенства  $y^2 - 4 > 0$  относительно переменной  $y$  ответ состоит из вектора двух ограничений ( $y < -2$ ,  $2 < y$ ) (ограничения, расположенные в виде вектора, обозначают совокупность). При символьном решении неравенства  $y^2 - 4 < 0$  ответ имеет вид:  $(y < 2) \cdot (-2 < y)$ , причем знак умножения означает, что ограничения должны рассматриваться в системе.

Существуют также функции для численного решения уравнений.

Для решения одного уравнения с одним неизвестным в MathCAD используется встроенная функция `root`. Аргументами этой функции являются выражение и переменная, входящая в выражение. Функция `root(f(z), z)` возвращает значение переменной  $z$ , которое обращает выражение или функцию  $f(z)$  в ноль.

Переменной, которая является вторым аргументом функции `root`, нужно перед использованием функции присвоить числовое значение. MathCAD использует его как начальное приближение при поиске корня. Выбор начального приближения влияет на корень, возвращаемый MathCAD, если выражение имеет несколько корней.

Пример: найдем  $a$  - решение уравнения  $e^x = x^3$ . Так как задача решения уравнения  $f(x) = g(x)$  эквивалентна задаче поиска корня выражения  $f(x) - g(x) = 0$ , используем функцию `root` следующим образом: `root(f(x)-g(x), x)`.

| Действие  | Выражение                        |
|---|----------------------------------|
| 1) Определим начальное значение переменной $x$ .      | $x := 3$                         |
| 2) Определим переменную $a$ как корень уравнения.     | $a := \text{root}(x^3 - e^x, x)$ |
| 3) Напечатаем « $a=$ », чтобы увидеть значение корня. | $a = 1.857$                      |

Для того чтобы найти корни выражения, имеющего несколько корней, нужно использовать различные начальные приближения для различных корней.

Пример: найдем корни многочлена  $x^3 - 10x + 2$ . Удобно воспользоваться графиком, чтобы приблизительно найти корни, а значит, и начальные приближения.

|   |                        |
|---|------------------------|
| 1) Укажем диапазон значений дискретного аргумента $x$ (чтобы построить график). | $x := -10, -9.9 .. 10$ |
|---|------------------------|

|   |   |
|---|---|
| 2) Построим график и определим по нему начальные приближения $x$ .            |   |
| 3) Используем три начальных приближения для нахождения трех различных корней. | $x:=-2 \quad \text{root}(x^3-10\cdot x +2, x)=-3.258$<br>$x:=0 \quad \text{root}(x^3-10\cdot x +2, x)=0.201$<br>$x:=3 \quad \text{root}(x^3-10\cdot x +2, x)=3.057$ |

MathCAD позволяет находить как комплексные, так и вещественные корни. Для поиска комплексного корня следует взять в качестве начального приближения комплексное число.

Для нахождения корней многочлена удобнее использовать функцию `polyroots`, которая не требует начального приближения и возвращает сразу все корни, как вещественные, так и комплексные; при этом все корни находятся численно (с помощью этой функции нельзя символично находить корни).

Функция `polyroots(v)` возвращает корни многочлена степени  $n$ , при этом коэффициенты многочлена находятся в векторе  $v$  длины  $n+1$ . Возвращает вектор длины  $n$ , состоящий из корней многочлена.

Например, для многочлена  $x^3-10\cdot x +2$ , рассмотренного в предыдущем примере, нахождение корней с помощью функции `polyroots` сведется к следующему:

$$v := \begin{bmatrix} 2 \\ -10 \\ 0 \\ 1 \end{bmatrix} \quad \text{polyroots}(v) = \begin{bmatrix} -3.258 \\ 0.201 \\ 3.057 \end{bmatrix}$$

## Системы уравнений

MathCAD дает возможность решать системы уравнений. Результатом решения системы будет набор численных значений или символьных выражений.

При решении систем уравнений используется ключевое слово `Given` (Дано), которое сообщает MathCAD, что далее следует система уравнений, и функция `Find` (Найти).

Для решения системы уравнений нужно выполнить следующее:

- 1) Ввести ключевое слово `Given` (при этом нужно убедиться, что слово `Given` не находится в текстовой области).
- 2) Ввести уравнения и неравенства (ограничения) в любом порядке ниже ключевого слова `Given`. Между левыми и правыми частями уравнений должен стоять жирный символ равенства (**=**), для печати которого используется комбинация клавиш «Ctrl» + «=». Можно также воспользоваться палитрой отношений.
- 3) Ввести любое выражение, которое включает функцию `Find`. Как правило, используется команда символьного вычисления значения функции:

`Find (z1, z2, z3 ...) →`

Функция `Find (z1, z2, z3 ...)` возвращает решение системы уравнений, расположенных между ключевым словом `Given` и функцией `Find`. Если функция `Find` имеет более одного аргумента, то она возвращает ответ в виде вектора (если решение системы единственное) или матрицы (если существует конечное число решений, большее 1). Например, `Find (z1, z2)` возвращает вектор, содержащий значения  $z1$  и  $z2$ , являющиеся решением системы уравнений.

Если система уравнений имеет бесконечное множество решений, решения могут быть записаны в общем виде (при этом значения одних переменных будут выражены через другие; если в

решении системы присутствуют имена переменных, являющихся аргументами функции Find, это означает, что эти переменные могут принимать любые действительные значения).

## **Символьные вычисления**

### **Символьные преобразования**

При численном вычислении MathCAD может работать только с числами и с переменными, значение которых известно. При символьных преобразованиях MathCAD может работать и с переменными, значения которых не определены.

Выполнить символьные преобразования выражения можно, используя меню Symbolic (Символика). При этом нужно сначала заключить нужное выражение в выделяющую рамку; затем выбрать соответствующую команду меню. MathCAD поместит преобразованное выражение в рабочий документ.

Меню «Символика»:

Evaluate→Evaluate Symbolically (Shift+F9) - вычислить символьно

Simplify - упростить

Expand Expression – “Расширить” - разложить все степени и произведения сумм в произведении («раскрыть скобки»)

Factor Expression – “Фактор” - разложить на множители

«Подобные» – привести подобные слагаемые

Действия с переменными (пункт меню Variable):

Differentiate on Variable – “Дифференциалы” - дифференцировать по переменной

Integrate on Variable – “Интеграция” - интегрировать по переменной

Solve for Variable – “Вычислить” - решить относительно переменной

Substitute for Variable – “Замена” - замена переменной

Expand to Series - разложить в ряд Тейлора

### **Замена переменных**

В MathCAD возможна замена переменной каким-либо выражением. Для этого нужно:

1. Выделить выражение, которое будет заменять переменную.
2. Скопировать его в буфер обмена.
3. Выделить переменную, которую нужно заменить, и выбрать в меню Символика пункт Substitute for Variable (Замена) - замена переменной

MathCAD заменит все вхождения переменной в данном выражении на содержимое буфера обмена.

## **Матрицы и векторы**

### **Создание матриц и операции над ними**

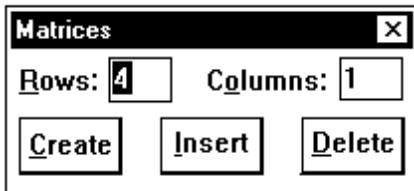
Для создания матрицы (вектора) нужно:

1. Щелкнуть мышью в нужном месте рабочего листа.
2. Выбрать на палитре матричных и векторных операций кнопку «Создать/редактировать матрицу»:



(Этот вид имеет как кнопка вызова палитры, так и кнопка в самой палитре).

Появится диалоговое окно.



3. Указать число строк (Rows) и столбцов (Columns) и нажать кнопку Create («Создать»).

Появится матрица с пустыми полями. Заполнить их элементами матрицы, передвигаясь между полями с помощью мыши или клавиши Tab.

При необходимости добавить в матрицу или удалить из нее строки или столбцы используется то же диалоговое окно; перед вызовом диалогового окна курсор устанавливается в позицию вставки или удаления строк или столбцов (строка и столбец, начиная с которого будут удаляться строки и столбцы). В диалоговом окне указывается количество вставляемых или удаляемых строк и столбцов. Нажатие кнопки Insert вставит в матрицу указанное количество строк и столбцов в позицию курсора; нажатие кнопки Delete удалит указанное количество строк и столбцов, начиная с позиции курсора.

Вектор в MathCAD создается как матрица из 1 столбца. Нельзя записывать вектор как матрицу из одной строки: в этом случае некоторые операции с векторами не будут выполняться.

Матричные операции и функции:

- $\cdot$  - умножение
- $+$  - сложение
- $-$  - вычитание
- $|v|$  - длина вектора
- $|M|$  - определитель матрицы
- $A^T$  - транспонированная матрица
- $A_{m,n}$  - элемент матрицы, находящийся в m-ой строке, n-ом столбце
- $M^{<n>}$  - n-й столбец матрицы M
- $rows(M)$  - число строк в матрице M
- $cols(M)$  - число столбцов
- $length(v)$  - число элементов в векторе v
- $max(M)$  - максимальный элемент в матрице M
- $min(M)$  - минимальный элемент в матрице
- $augment(A,B)$  - матрица, сформированная размещением матриц A и B бок о бок (A и B должны иметь одинаковое число строк)
- $stack(A,B)$  - матрица, сформированная размещением матрицы A над B (A и B должны иметь одинаковое число столбцов)

Для того, чтобы выделить элемент вектора, используется нижний индекс; для выделения элемента матрицы - два нижних индекса, отделяемые запятой. Перейти в нижний индекс можно с помощью палитры матричных операций или с помощью нажатия клавиши «[». Чтобы выйти из нижнего индекса, нужно нажать клавишу «пробел» или с помощью стрелок на клавиатуре изменить положение курсора.

С матрицами можно выполнять и символьные преобразования. Символьно можно транспонировать матрицу, найти обратную матрицу и определитель. Для этого необходимо:

- 1) Заключить матрицу в выделяющую рамку
- 2) Выбрать Matrix Operations (Матрицы) из меню «Символы» и указать нужный пункт: Transpose Matrix (Транспонирование) – транспонировать матрицу, Invert Matrix (Инвертирование) – обратная матрица, Determinant of Matrix (Определитель) – определитель матрицы.

## Решение систем линейных уравнений с использованием матриц

Для решения систем линейных уравнений используют функцию  $\text{lsolve}(M,v)$ , где  $M$  – матрица, содержащая коэффициенты при переменных,  $v$  – столбец свободных членов). MathCAD возвращает вектор решения  $x$  такой, что  $M \cdot x = v$ .

Пример: решение системы уравнений

$$3 \cdot x + 6 \cdot y = 9$$

$$2 \cdot x + 54 \cdot y = 4 \quad (\text{Знаки равенства в уравнениях должны быть жирными!})$$


$$M := \begin{pmatrix} 3 & 6 \\ 2 & 54 \end{pmatrix} \quad v := \begin{pmatrix} 9 \\ 4 \end{pmatrix}$$

$$\text{lsolve}(M, v) = \begin{pmatrix} 3.08 \\ -0.04 \end{pmatrix}$$

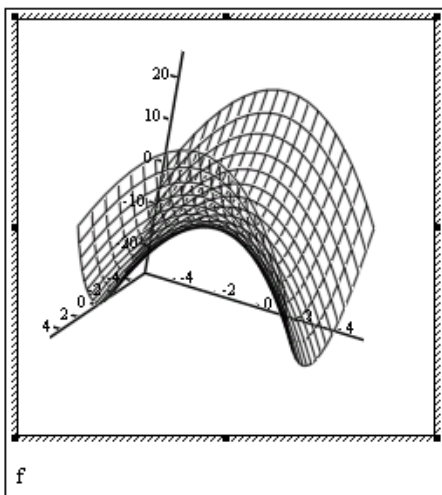
Таким образом,  $x = 3.08$ ,  $y = -0.04$ .

Можно также использовать функцию  $\text{rref}(A)$ , где  $A$  – расширенная матрица системы (содержит коэффициенты при переменных и свободные члены). Функция решает систему линейных уравнений методом Гаусса.

## Построение графиков функций двух переменных

Для построения графика функций двух переменных используется шаблон графика, вставляемый в документ с помощью кнопки . В поле ввода в нижней части шаблона записывается имя функции.

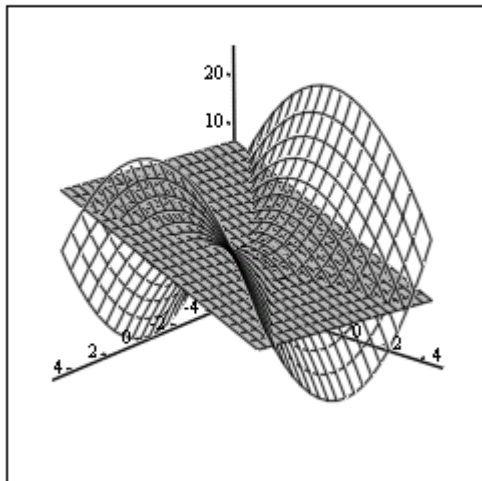
$$f(x, y) := x^2 - y^2$$



Если необходимо в одной области построить несколько графиков, имена функций перечисляются в поле ввода через запятую:




$$f(x,y) := x^2 - y^2 \quad g(x,y) := \frac{x}{2} - 2y$$



f, g

В некоторых случаях удобнее строить графики функций двух переменных, используя матрицу, содержащую значения функции. В каждой ячейке матрицы при этом находится значение функции в определенной точке.

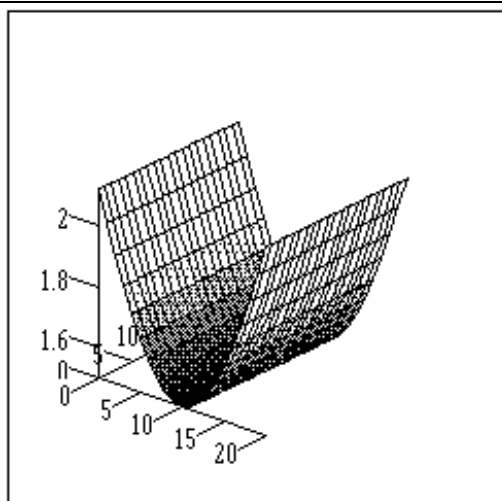
Для построения графика функции двух переменных нужно:

1. Определить функцию двух переменных.
2. Выбрать минимальные и максимальные значения аргументов и количество точек на каждой оси, лежащих в выбранном диапазоне. Определить дискретные аргументы  $i, j$  для того, чтобы проиндексировать эти точки.
3. Определить  $x_i$  и  $y_j$  как равномерно располагаемые точки на осях  $x$  и  $y$ .
4. Заполнить матрицу значениями функции.
5. Выбрать в палитре построения графиков график поверхности: ; напечатать имя матрицы в поле ввода и щелкнуть мышью вне графической области.

Рассмотрим пример: построение графика функции  $f(x,y) = \sqrt{x^2 + y^2}$  на множестве точек:  $-1.5 \leq x \leq 1.5, -1.5 \leq y \leq 1.5$ .

| Действие                                | Пример   |
|---|--|
| 1. Определим функцию двух переменных:   | $f(x,y) := \sqrt{x^2 + y^2}$                                 |
| 2. Определим $i, j$ :                   | $i := 0..20$<br>$j := 0..20$                                 |
| 3. Определим $x$ и $y$ :                | $x_i := -1.5 + 0.15 \cdot i$<br>$y_j := -1.5 + 0.15 \cdot j$ |
| 4. Заполним матрицу значениями функции: | $M_{i,j} := f(x_i, y_j)$                                     |

5. Построим график:



М

На рисунке видно, что на осях отмечаются значения не для  $x$  и  $y$ , а для  $i$  и  $j$ .

## Статистика

### Статистические функции

В MathCAD есть различные встроенные статистические функции. Вот наиболее часто используемые из них:

$\text{mean}(A)$  - среднее значение элементов массива  $A$

$\text{var}(A)$  - дисперсия элементов массива  $A$

$\text{covar}(A,B)$  - ковариация элементов массивов  $A$  и  $B$

$\text{corr}(A,B)$  - коэффициент корреляции для массивов  $A$  и  $B$

$\text{rnd}(X)$  - случайное действительное число от 0 до  $X$

$\text{dbinom}(k,n,p)$  - вероятность  $k$  успехов в серии из  $n$  испытаний;  $p$  - вероятность успеха в одном испытании

$\text{dpois}(k,\lambda)$  - вероятность  $k$  успехов, в случае, когда случайная величина имеет распределение Пуассона;  $\lambda=n*p$ , где  $n$  - число испытаний;  $p$  - вероятность успеха в одном испытании

## Линейная регрессия

Функции линейной регрессии возвращают наклон и смещение линии, которая наилучшим образом приближает данные (метод наименьших квадратов). Если поместить значения  $x$  в вектор  $vx$ , а значения  $y$  в вектор  $vy$ , то линия определится в виде:

$$y = \text{slope}(vx, vy) \cdot x + \text{intercept}(vx, vy),$$

где

$\text{slope}(vx, vy)$  - наклон линии регрессии,

$\text{intercept}(vx, vy)$  - смещение линии регрессии по оси ординат.

Пример 1. Нахождение линии, приближающей данные, находящиеся в файлах  $f1$  и  $f2$ .

$i := 0 .. 50$

$x_i := \text{READ}(f1)$  (считываются из файла значения  $x$ )

$y_i := \text{READ}(f2)$  (считываются из файла значения  $y$ )

$a := \text{slope}(x, y)$

$b := \text{intercept}(x, y)$

$f(x) := a \cdot x + b$

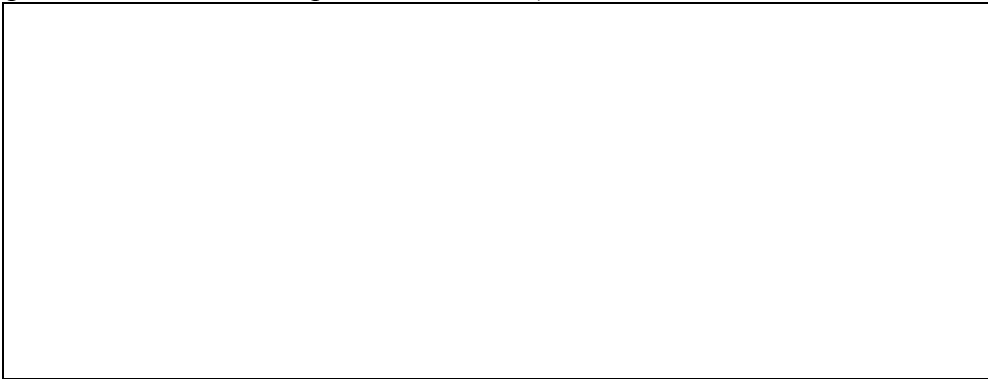
Пример 2 (иллюстрирует также использование функции  $\text{rnd}$ ). Вектор  $x$  формируется следующим образом: чтобы получить элемент  $x_j$ , к порядковому номеру  $j$  элемента добавляется случайная величина. Аналогично формируется вектор  $y$ . Как и в предыдущем примере, вектора  $x$  и  $y$  определяют координаты точек, через которые мы будем проводить линию.

```

j := 0 .. 40
xj:=j + rnd(6)
yj:=j + rnd(5)
a:=slope(x,y)
b:=intercept(x,y)

```

На рисунках показано, как выглядит соотношение исходных данных и полученной линии( при различных типах отображения данных).



## Создание анимации в MathCAD

MathCAD предоставляет пользователю возможность создания анимации. Для создания анимации используется целочисленная переменная FRAME, обозначающая номер кадра анимации. При построении анимации эта переменная принимает значения в выбранном диапазоне, пересчитывается содержимое выделенной части документа для каждого из значений переменной FRAME и присоединяется к анимации в качестве очередного кадра.

Для создания анимации необходимо:

- 1) подготовить анимируемую часть документа (в документе должно быть хотя бы одно выражение, содержащее переменную FRAME);
- 2) вызвать команду меню View | Animate;
- 3) выделить анимируемую часть документа (математические области не обязательно должны целиком попадать в выделение);
- 4) в диалоговом окне указать начальное (From) и конечное (To) значения переменной FRAME и количество кадров в секунду;
- 5) запустить процесс анимации с помощью кнопки Animate в диалоговом окне.

$t := 0.1 \cdot \text{FRAME}$

$f(x) := x^2 + t$

$f(x)$

$x$

FRAME = 27

Select an area of your worksheet whose contents are based on the FRAME variable, enter starting and ending FRAME values, and choose Animate.

## Лабораторная работа № 1

1. Запустите MathCad. Ознакомьтесь с меню приложения и панелями инструментов.
2. Наберите произвольное арифметическое выражение (знак умножения вставляется клавишей \*, знак деления /). Вычислите его значение, нажав клавишу =, когда курсор находится в области выражения.
3. Откройте арифметическую палитру (кнопка с изображением калькулятора). Ознакомьтесь с ее командами.
4. Введите следующие выражения (с помощью клавиатуры или арифметической палитры) и вычислите их значения:

$$3+16.7 \quad 3 \cdot 79 \quad \frac{45}{13} \quad \sqrt{3} \quad \sqrt[3]{7} \quad 1.5^3 \quad \frac{15.1 - 2.4^2}{\sqrt{2 - \sqrt{6}}}$$

$$\pi \cdot \sqrt{47} \quad \left| \ln\left(\frac{7}{13} - 5\right) \right| \quad \frac{3 \cdot \pi}{2^4} + e \cdot 3$$

5. Ознакомьтесь с командами палитр «Высшая математика» и «Греческие буквы». С их помощью введите в документ следующие выражения:
  - неопределенный интеграл функции  $\sin(x)$
  - определенный интеграл функции  $f(x) = x^2+3$  (пределы интегрирования укажите произвольно)
  - производная функции  $f(x) = x^3+3x^2+4x-2$ ; вторая производная той же функции
  - синус разности  $\alpha-\beta$
  - натуральный логарифм числа  $\alpha$ .
  - модуль суммы чисел  $\lambda$  и  $\mu$ .

Можно ли вычислить значения этих выражений численно? символично? Выполните символичные преобразования для тех выражений, для которых это возможно.

6. Вставьте в документ несколько пустых строк и ниже скопируйте выражения из предыдущего задания. Перед добавленными копиями выражений задайте значения переменных, входящих в выражения, с помощью команды присваивания; вычислите значения выражений численно.
7. Измените значения переменных; обратите внимание на пересчет результатов.
8. Задайте функцию  $f(t)=t^2$ . Попробуйте вывести на экран значение  $f(t)$ . Объясните полученный результат.
9. Задайте значение дискретного аргумента  $t$ , принимающего значения от 0 до 5 с шагом 1. В каком месте документа должна располагаться команда присваивания, чтобы MathCad мог вычислить значения  $f(t)$ ?
10. Постройте график функции  $f(t)$ . Измените минимальные и максимальные отображаемые значения по обеим осям. Объясните полученный вид графика.
11. С помощью двойного щелчка в области графика активизируйте диалоговое окно свойств графика. Ознакомьтесь с его вкладками. На вкладке Traces измените цвет и толщину линии графика.
12. В той же области графика постройте график функции  $f(w)$  (значения переменной  $w$  не определяйте). Обратите внимание на различие графиков (различные области определения и разная точность построения точек)!
13. Перейдите в верхнюю часть документа MathCad. Вставьте текстовый заголовок «MathCad» (для этого просто начните набирать текст; после ввода первого пробела шрифт изменит вид. Можно также воспользоваться командой меню Insert | Text Region или нажать на клавиатуре клавишу с изображением кавычки). Введите подзаголовок «Выражения и функции» (Если русский текст не читается,

выделите его и с помощью панели инструментов установите для выделенного фрагмента шрифт MS Sans Serif).

14. Расположите области в документе так, чтобы они не перекрывали друг друга.

## Лабораторная работа №2

### Графики. Символьные преобразования

1. Постройте графики функций  $f(x)$ ,  $g(x)$ ,  $f(g(x))$  на одном чертеже. Установите формат осей «пересечение в начале координат», одинаковый масштаб по оси абсцисс и оси ординат, вспомогательные линии (с интервалом, равным единичному отрезку). Измените цвета линий графиков функций таким образом, чтобы они не сливались со вспомогательными линиями.

a)  $g(x) = \cos(x)$        $f(x) = \ln(x)$

b)  $g(x) := \frac{1}{x}$        $f(x) := \arccos(x)$

c)  $g(x) := \frac{2 \cdot x}{1+x^2}$        $f(x) := \arcsin(x)$

d)  $g(x) := \frac{1}{\sin(x)}$        $f(x) := \arctan(x)$

e)  $g(x) := x \cdot \sqrt{-x^2}$        $f(x) := \sqrt{x}$

f)  $g(x) := \sin(x)$        $f(x) := \arcsin(x)$

2. Задайте дискретный аргумент  $t$  с начальным значением  $-10$ , конечным значением  $10$  и шагом  $0.1$ ; задайте функции  $x(t)$  и  $y(t)$  и постройте график параметрической кривой (под осью абсцисс задается значение  $x(t)$ , слева от оси ординат -  $y(t)$ ), если:

▪  $x(t)=\sin(t), y(t)=\cos(t)$

▪  $x(t)=\sin(t), y(t)=\sin(t) \cos(t)$

▪  $x(t)=\sin(t), y(t)=\sin^2(t) \cos(t)$

3. Постройте график функции в полярной системе координат (аргумент, т.е. угол, записывается в поле под графиком, значение функции - слева от графика):

$f1(v)=3$ ;       $f2(w)=w/2$ ;       $f3(\theta)=\sin(5\theta/3)$ ;       $f4(\phi)=1+\cos(3\phi)+\sin^2(3\phi)$

4. Упростите с помощью меню «Символика»:

▪  $\frac{a^{-3} \cdot b^{-1} + a^{-1} \cdot b^{-3}}{a^{-4} - b^{-4}} \cdot \frac{2}{(a-b)^{-1}}$

▪  $\frac{m-2}{m \cdot (m-2) + 4} + \frac{8+4 \cdot (1-m) + m^2}{8+m^3} - \frac{1}{2+m}$

▪  $\frac{3 \cdot n + 10}{n + 4} + \left( \frac{n-4}{n+6} \right)^2 \cdot \left( \frac{n+21}{16-8 \cdot n + n^2} - \frac{n+3}{16-n^2} \right)$

▪  $\frac{\frac{a+2}{a-2}}{\frac{6 \cdot a}{a^3-8} + \frac{2 \cdot a}{a^2+2 \cdot a+4} + \frac{1}{2-a}} - \frac{4 \cdot a + 4}{a-2}$

5. Докажите тождества:

▪  $(x^2 + b^2) \cdot (x^2 + y^2) = (x \cdot x + b \cdot y)^2 + (x \cdot y - b \cdot x)^2$ .

▪  $8 \cdot a^3 + 12 \cdot a^2 \cdot b + 12 \cdot a^2 \cdot c + 6 \cdot a \cdot b^2 + 12 \cdot a \cdot b \cdot c + 6 \cdot a \cdot c^2 + b^3 + 3 \cdot b^2 \cdot c + 3 \cdot b \cdot c^2 + c^3 = (2 \cdot a + b + c)^3$

$$\left(\frac{c^2}{b^2} + b^2\right) \cdot (c^6 \cdot b^2 + y^2) = (c^4 + b \cdot y)^2 + \left(\frac{c}{b} \cdot y - b^2 \cdot c^3\right)^2$$

6. Найдите интеграл функции  $y = \frac{x}{\sqrt{x^2-1} \cdot \sqrt{x^2+1}}$  на отрезке [3; 4].

7. Найдите неопределенный интеграл функции  $y = \frac{x}{\sqrt{x^2-1} \cdot \sqrt{x^2+1}}$ .

8. Найдите производную функции  $y = \frac{1}{4} \cdot \ln\left(\frac{x^2-1}{x^2+1}\right)$ . Упростите полученное выражение.

9. Найдите пятую производную функции  $y = 2 \cdot \cos(x) - 2 \cdot \sin(x)$ .

10. Найдите предел функции  $y = 2 \cdot \sqrt{x} \cdot \ln\left(\frac{x-1}{x-2}\right)$  при  $x \rightarrow \infty$

11. Решите следующее уравнение относительно переменной  $x$ :  $\pi \cdot x = e$

12. Решите следующее уравнение относительно переменной  $x$ ; относительно переменной  $a$ ; переменной  $t$ . В последнем случае объясните полученный результат.

$$x \cdot t^2 = \frac{1}{a}$$

13. Решите следующее уравнение относительно переменной  $t$ ; относительно переменной  $x$  (во втором случае обратите внимание на то, что выводится только одно значение):

$$\sqrt{\sin(x) \cdot t^2} = 4$$

14. Решите следующие неравенства относительно переменной  $x$ . Объясните полученные результаты, добавив в документ текстовые комментарии.

$$x^2 + 6 \cdot x - 9 < 0$$

$$x^2 + 6 \cdot x - 9 > 0$$

$$x^2 + 6 \cdot x + 9 < 0$$

$$x^2 + 6 \cdot x + 9 > 0$$

$$x^2 + 6 \cdot x + 10 > 0$$

15. Задайте произвольный многочлен  $f(x)$  3-й или 4-й степени. Найдите его корни с помощью функции `root` (предварительно построив график, чтобы найти приближения корней); с помощью функции `polyroots` (аргументом функции является вектор  $v$ , где  $v_i$  – коэффициент при  $x^i$ ).

16. Решите следующие системы линейных уравнений с помощью операторов `Given` и `Find`. Добавьте в документ текстовые комментарии, содержащие объяснение полученных результатов.

Уравнения систем:

|                   |                    |                   |
|-------------------|--------------------|-------------------|
| $3x + 2y + z = 5$ | $x + y + z = 5$    | $x + y + z = 2$   |
| $x + y - z = 0$   | $x + 3y + 3z = 4$  | $2x - 2y - z = 2$ |
| $4x - y + 5z = 3$ | $x + 5y + 5z = 17$ | $3x - y = 4$      |

17. В третьей системе уравнений из предыдущего задания значения  $y$  и  $z$  при решении выражены через  $x$ . Измените аргументы функции `Find` так, чтобы переменные  $x$  и  $y$  были выражены через  $z$ .

## Лабораторная работа №3

### Матрицы и их использование

1. Задайте произвольную квадратную матрицу  $A$  размера не менее  $3 \times 3$ . Найдите ее определитель (если определитель матрицы равен нулю, измените значения элементов матрицы), транспонированную и обратную матрицу.
2. Задайте матрицу  $B$  того же размера, что и  $A$ . Найдите сумму, разность матриц, произведения  $AB$  и  $BA$ .

3. Используя встроенные функции  
`augment(A,B)` - матрица, сформированная размещением матриц A и B бок о бок (A и B должны иметь одинаковое число строк),  
`stack(A,B)` - матрица, сформированная размещением матрицы A над B (A и B должны иметь одинаковое число столбцов),  
 приклейте матрицу B к матрице A:  
 3.1. справа;  
 3.2. снизу.
4. Задайте матрицу C размера 2x2 так, чтобы ее элементами являлись переменные, значения которых не определены. Вычислите символьно определитель этой матрицы. Сделайте то же с матрицами размера 3x3 и 4x4.
5. Откройте документ MathCad с именем «Matr». Для заданной в нем матрицы M с помощью встроенных операторов и функций найдите:  
 5.1. количество строк и столбцов матрицы, общее количество элементов матрицы;  
 5.2. максимальный элемент матрицы;  
 5.3. максимальный элемент в 4 столбце (для этого с помощью специального оператора  $M^{<v>}$  задайте новую матрицу M1, совпадающую со столбцом матрицы M);
6. Решите первую из следующих систем линейных уравнений:  
 6.1. представив систему в матричном виде:  $AX=B$  (задайте матрицы A и B, найдите вектор X);  
 6.2. с помощью функции `rref`.

Попробуйте решить этими способами вторую и третью системы. Объясните результат.

|                   |                    |                   |
|-------------------|--------------------|-------------------|
| $3x + 2y + z = 5$ | $x + y + z = 5$    | $x + y + z = 2$   |
| $x + y - z = 0$   | $x + 3y + 3z = 4$  | $2x - 2y - z = 2$ |
| $4x - y + 5z = 3$ | $x + 5y + 5z = 17$ | $3x - y = 4$      |

7. Задайте два вектора x и y, элементами которых будут координаты некоторых точек на плоскости. Отобразите эти точки на графике, поместив имена векторов в поля ввода у осей абсцисс и ординат. Выберите наиболее подходящие свойства графика для отображения точек.
8. Задайте матрицу 5x2, первый столбец которой будет содержать абсциссы точек, а второй – их ординаты. Отобразите точки на графике.
9. Задайте матрицу, содержащую координаты вершин произвольного квадрата. Отобразите точки на графике так, чтобы можно было видеть все стороны квадрата.
10. Постройте графики поверхностей (все 6):  
 10.1.  $z = xy$   
 10.2.  $z = x^2 - y^2$   
 10.3.  $z = x^2 + y^2$   
 10.4.  $z = \sqrt{x^2 + y^2}$   
 10.5.  $z = x \cdot y^2$   
 10.6.  $z = \sqrt{1 - x^2 - y^2}$  (график строится с помощью задания матрицы координат точек)

Измените свойства графиков, применяя к ним различное оформление.

## Лабораторная работа №4. Стандартные функции

1. **Комплексные числа.**  
 Задайте произвольное комплексное число z1 и найдите с помощью встроенных функций его действительную часть, мнимую часть, аргумент, модуль и комплексно сопряженное.  
 Задайте второе комплексное число z2 и найдите для z1 и z2 их сумму, разность, произведение.
2. **Вероятность**  
`dbinom(k,n,p)` - вероятность k успехов в серии из n испытаний; p - вероятность успеха в одном испытании

- 2.1. Найти вероятность рождения двух мальчиков в семье из 5 детей.
- 2.2. 70 % поступивших на физмат оканчивают институт через 5 лет. Найти вероятность того, что из 25 первокурсников ровно 20 через 5 лет получают диплом.
- 2.3. В урне находятся 5 белых и 7 черных шаров. Извлекаем 6 шаров по одному с возвратом. Какова вероятность того, что будут извлечены ровно 4 белых шара?

### 3. Случайные числа

С помощью функции `rnd(x)` (случайное число от 0 до x) и `ceil(x)` (округление полученного случайного числа в большую сторону) смоделируйте подбрасывание игральной кости 100 раз (результат i-го подбрасывания будет записан в элемент вектора `v`). Просмотрите полученный результат.

### 4. Линейная регрессия

Функции линейной регрессии возвращают наклон и смещение линии, которая наилучшим образом приближает данные (метод наименьших квадратов). Если поместить значения `x` в вектор `vx`, а значения `y` в вектор `vy`, то линия определится в виде:

$$y = \text{slope}(vx, vy) \cdot x + \text{intercept}(vx, vy),$$

где

`slope(vx, vy)` - наклон линии регрессии,

`intercept(vx, vy)` - смещение линии регрессии по оси ординат.

**Пример** (иллюстрирует также использование функции `rnd`). Вектор `x` формируется следующим образом: чтобы получить элемент `xj`, к порядковому номеру `j` элемента добавляется случайная величина. Аналогично формируется вектор `y`. Вектора `x` и `y` определяют координаты точек, через которые мы будем проводить линию.

```
j := 0 .. 40
```

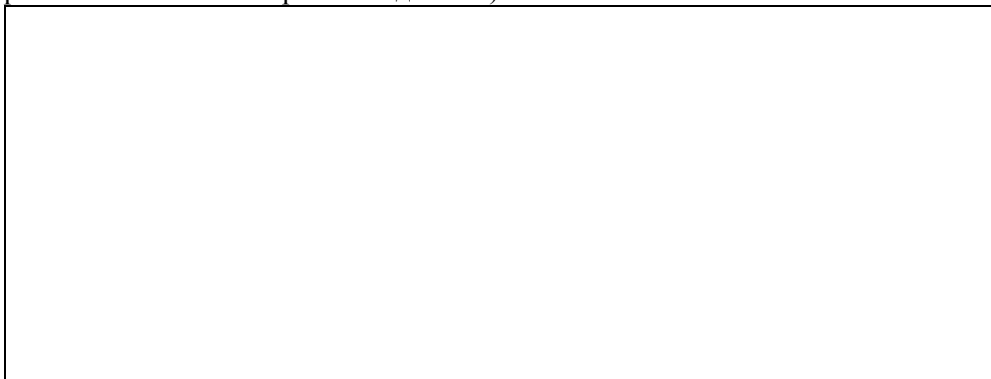
```
xj:=j + rnd(6)
```

```
yj:=j + rnd(5)
```

```
a:=slope(x,y)
```

```
b:=intercept(x,y)
```

На рисунках показано, как выглядит соотношение исходных данных и полученной линии (при различных типах отображения данных).



Задайте значения координат точек, случайным образом отклоняющихся от некоторой прямой. Постройте линию регрессии и с помощью графика проверьте, как она приближает заданные вами данные.

### 4. Анимация

1. Присвойте переменной `a` значение 2. Постройте график функции  $f(x)=x^a$ .
2. Измените формат графика так, чтобы на рисунке отображалась только первая координатная четверть. Попробуйте изменять значение переменной `a` от -3 до 3 (включая 0, 1, а также дробные значения); выясните, при каких значениях свойств графика эти графики будут отображаться наиболее полно, и установите эти значения свойств. Установите одинаковый масштаб по обеим осям. Задайте минимальное и максимальное значение аргумента и функции.
3. Измените значение переменной `a`, выразив ее через встроенную переменную `FRAME` (например, `-3+0.1 FRAME`). Создайте анимацию графика, задав значения переменной `FRAME` так, чтобы показатель степени менялся от -3 до 3.
4. Откройте файлы `ball.avi` и `ball2.avi`. Просмотрите содержащиеся в них анимации; создайте аналогичные анимации, используя графики функций  $f(x)=a \cdot x^2$  и график эллипса, заданного параметрически.



Жохова Елена Юрьевна  
Заводчикова Надежда Ивановна  
Кокорева Ирина Евгеньевна  
Корнилов Петр Анатольевич  
Московская Лина Яковлевна  
Плясунова Ульяна Валерьевна  
Семенова Ольга Геннадьевна

## **Информатика и информационные технологии**

(для студентов специальности 050201.65 – «Математика»)

Редактор

Подписано в печать

Формат

Печ. л. Заказ

Тираж экз.

Редакционно-издательский отдел Ярославского государственного  
педагогического университета имени К.Д.Ушинского (ЯГПУ)  
150000, Ярославль, Республиканская, 108  
ЛР №020080 от 19.12.97

Типография ЯГПУ  
150000, Ярославль, Которосльская наб., 44