

Министерство образования и науки РФ  
Севастопольский государственный университет  
Кафедра информатики и управления в технических системах

## **ОСНОВЫ ЯЗЫКА ПРОГРАММИРОВАНИЯ PASCAL**

**Методические указания**  
к выполнению лабораторных работ  
по дисциплине  
"Алгоритмизация и программирование"  
для студентов очной формы обучения  
по направлениям подготовки  
27.03.04 "Управление в технических системах" и  
09.03.01 " Информатика и вычислительная техника"

Севастополь  
2017

УДК 681.5

**Основы языка программирования Pascal:** Методические указания к выполнению лабораторных работ, входящих в блок № 1 лабораторного практикума по дисциплине «Алгоритмизация и программирование»/ Сост. Д.Н. Старинская, А.А. Кабанов, В.В. Захаров. – Севастополь: Изд-во СевГУ, 2017. – 25 с.

Целью методических указаний является оказание помощи студентам при выполнении лабораторных работ, целью которых является приобретение навыков составления простейших алгоритмов, изучение основ языка Pascal, освоение базовых приемов работы в среде Turbo Pascal и её модификаций Free Pascal, ABC PASCAL и т.п..

Методические указания предназначены для студентов дневной формы обучения по направлениям подготовки 27.03.04 "Управление в технических системах" и 09.03.01 "Информатика и вычислительная техника"

Методические указания рассмотрены и утверждены на заседании кафедры информатики и управления в технических системах (протокол № 7 от 30.09.2017 г.)

Допущено учебно-методическим центром СевГУ в качестве методических указаний.

Рецензент

Кабанов А.А., канд. техн. наук, доцент

## СОДЕРЖАНИЕ

Введение .....	4
1. Лабораторная работа № 1. Программирование линейных алгоритмов на языке Pascal .....	4
1.1. Цель работы .....	4
1.2. Задание на работу .....	4
1.3. Краткие теоретические сведения .....	6
1.3.1. Описание лабораторного стенда .....	6
1.3.2. Стандартные арифметические функции .....	7
1.3.3. Пример программы .....	7
1.4. Содержание отчета и порядок защиты работы .....	9
1.5. Контрольные вопросы .....	10
2. Лабораторная работа № 2. Программирование разветвляющихся алгоритмов на языке Pascal .....	10
2.1. Цель работы .....	10
2.2. Задание на работу .....	10
2.3. Пример программы .....	14
2.4. Содержание отчета и порядок защиты работы .....	15
2.5. Контрольные вопросы .....	16
3. Лабораторная работа № 3. Программирование циклических алгоритмов на языке Pascal .....	16
3.1. Цель работы .....	16
3.2. Задание на работу .....	16
3.3. Краткие теоретические сведения .....	18
3.3.1. Рекуррентные формулы вычисления значений функций .....	18
3.3.2. Пример программы .....	19
3.4. Содержание отчета и порядок защиты работы .....	20
3.5. Контрольные вопросы .....	20
Библиографический список .....	20
Приложение А. Пример оформления титульного листа отчета .....	21
Приложение Б. Пример оформления основной части отчета .....	22
Приложение В. Перечень тем блока № 1 .....	23
Приложение Г. Пример экзаменационного задания по тематике блока № 1 .....	24

## ВВЕДЕНИЕ

Лабораторные работы данного цикла входят в блок №1 дисциплины «Алгоритмизация и программирование». Для успешной сдачи блока № 1 необходимо выполнить и защитить лабораторные работы №№ 1-3 *не позднее 8-ой недели семестра*, а также написать контрольную работу №1 на оценку не менее "удовлетворительно".

Примерный перечень тем, входящих в блок № 1, приведен в приложении В. Пример листа задания контрольной работы представлен в приложении Г.

## 1. ЛАБОРАТОРНАЯ РАБОТА №1. ПРОГРАММИРОВАНИЕ ЛИНЕЙНЫХ АЛГОРИТМОВ НА ЯЗЫКЕ PASCAL

### 1.1. Цель работы

Целью данной работы является исследование линейных алгоритмов и их реализация с помощью операторов языка Pascal.

### 1.2. Задание на работу

1) Ознакомьтесь с теоретическими сведениями об арифметических функциях языка Pascal, приведенными в пункте 1.3.1 методических указаний.

2) Изучите пример программы, приведенный в пункте 1.3.2. Рекомендуем самостоятельно повторить все стадии разработки программы, приведенные в примере.

3) Создайте программу, реализующую вычисление по формулам, указанным в таблице 1.1 в соответствии с номером варианта. Для этого выполните следующие этапы:

- преобразуйте формулы с целью уменьшения количества операций при вычислениях. Упрощение возможно как за счет математических преобразований, так и за счет введения дополнительных переменных для сохранения значений выражений, неоднократно встречающихся в формуле;

- для двух заданных вариантов исходных данных вычислите с помощью калькулятора значения, определяемые формулами (Переведите калькулятор в режим работы с радианами при вычислении тригонометрических функций). При этом окажется, что одна пара исходных значений не входит в область допустимых значений, т.е. приведет к возникновению ошибки;

- составьте алгоритм (схему) программы. В программе должен быть предусмотрен диалоговый ввод значений входных переменных, вывод результатов с пояснениями;

- в соответствии с алгоритмом составьте программу на языке Pascal;

- запустите программу на счет при обоих вариантах значений входных переменных. Проанализируйте полученные результаты и сообщения среды Turbo Pascal.

Таблица 1.1 – Варианты заданий

№ п/п	Типы переменных	Варианты исход- ных данных	Задания
1	$a, b, y, z$ – веще- ственные	а) $a = 5,7; b = 0,08$ б) $a = -0,3; b = 15,1$	$y = \frac{e^{-a} + \frac{z+10^3}{\sin z}}{\cos \pi z + \ln b}, z =  b-15,1 $
2	$a, x, k$ – веществен- ные; $y$ – целая	а) $x=-3,06; y=3$ б) $x=0,215; y=-11$	$a = \frac{e^{-tg \pi k} - \ln  x }{kx + 10^5}, k = \frac{y+11}{2}$
3	$k, x, i$ – веществен- ные; $m$ – целая	а) $x=0,02; m=3$ б) $x=-1,5; m=-10$	$k = \frac{\ln  i + 8 \cdot 10^{-2}  - \ln i}{\cos \pi i + e^x}, i = \frac{m-1}{m+1}$
4	$m, n, p$ – веще- ственные; $r$ – целая	а) $m=716,2; r=1$ б) $m=0,07; r=-11$	$n = e^{\frac{-\sin m + tg \frac{\pi r}{p}}{p}}, p = \ln \left  \frac{r^3 + 10^3}{r^3 - 1} \right $
5	$d, k, z$ – веществен- ные; $x$ – целая	а) $k = -0; 12 x=0$ б) $k = 1,5; x=-11$	$d = \ln \frac{z^2 + 10^{-3}}{z^2 + 1,6 \cdot 10^{-2}}, z = e^{-k} \operatorname{ctg} kx$
6	$m, x, z$ – веще- ственные; $y$ – целая	а) $x = 1,016; y=11$ б) $x=-0,2; y=2$	$m = \sin \operatorname{arctg} \frac{z}{2} - \sin \operatorname{arctg} \frac{\pi}{3}$ $z = \frac{ e^{-xy} + 10^{-3}  + e^{-xy}}{\pi + \ln xy}$
7	$z, x, m$ – веществен- ные; $c$ – целая	а) $x = -0,02; c = -3$ б) $x = -1,1; c = 2$	$z = \operatorname{tg}^2 \frac{\pi m}{m + 10^{-3}}, m = e^{-c x  + \sqrt{cx}}$
8	$K$ – целая $r, t, m$ – веществен- ные	а) $r = 0,07; k = 6$ б) $r = 0,63; k = -15$	$t = \sqrt{\frac{\cos^2 \pi r + k \cdot 10^{-2}}{\cos^2 \pi r + \left  \frac{m}{k} \right }}, m = \ln \cos \pi r$
9	$x, z$ – целые $m, k$ – вещественная	а) $x = -4; z = 12$ б) $x = 2; z = -6$	$m = \frac{\sqrt{\ln k + 10^3} - \sqrt{10^3 - \ln k}}{x^2 + 18x - 40}$ $k = e^{\pi x} \cos 0,01z$
10	$X$ – целая $m, k, z$ – веществен- ные	а) $x = 12; m = 0,51$ б) $x = -7; m = -0,05$	$k = \operatorname{tg}^2 z + \operatorname{ctg}^2 z, z = \frac{e^{\pi x} - e^{-\pi x}}{10^3 + \sqrt{\ln mx}}$
11	$k$ – целая; $x, b, n$ – вещественные	а) $k = 2750 x = 18,1$ б) $k = -124 x = -0,73$	$b = \sqrt{\frac{\sin^2 n + \sin n^2}{ \sin n  + e^{-x}}}, n = \ln \frac{\pi}{kx - 1,6 \cdot 10^3}$
12	$z$ – целая; $m, r, t$ – вещественные	а) $r = 0,12 z = 10$ б) $r = 8,3 z = -116$	$m = 180 \operatorname{arctg} \frac{e^{5t} - e^{-5t}}{e^{zt} - e^{-zt}}, t = \sqrt{\frac{\ln  \pi r }{10^3 + rz}}$
13	$a, b, x, y$ – веще- ственные	а) $x = 1,625; y = 0,825$ б) $x = -2,35; y = 1,115$	$a = \cos^2 \operatorname{tg} \left( \frac{1}{b} \right), b = \frac{x + \frac{y}{5 + \sqrt{x}}}{ y + x  + \sqrt[3]{x}}$
14	$a, b, x, y$ – веще- ственные	а) $x = 47,8; y = 5,5;$ в) $x = 0; y = 2,3$	$a = e^{b-1} + \operatorname{tg}^2 x, b = \frac{2 \cos(x - \pi/6)}{\frac{1}{x} + \sin^2 y}$

### 1.3. Краткие теоретические сведения

#### 1.3.1. Описание лабораторного стенда

Лабораторный стенд представляет собой персональный компьютер с установленной на нем средой программирования Turbo Pascal 7.0. Схема лабораторного стенда дана на рис.1.1.

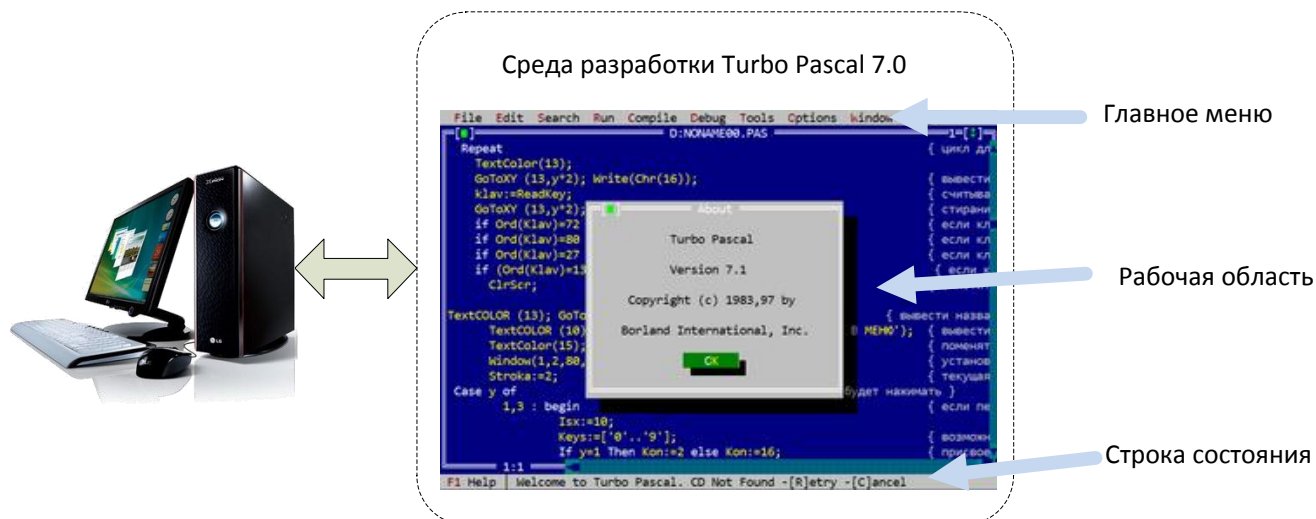


Рисунок 1.1 – Структура лабораторного стенда

Базовыми компонентами системы программирования Turbo Pascal являются компилятор языка Pascal, средства создания и редактирования исходных текстов программ и средства их отладки (поиска и исправления ошибок). Все эти компоненты объединены в единую интегрированную среду разработчика, с которой как раз и работает программист, создавая свои программы.

Основной экран интегрированной среды разработчика Turbo Pascal 7.0 показан на рис.1.1. По функциональному назначению выделяется три области экрана:

- строка меню;
- рабочая область;
- строка состояния.

Строка меню активизируется нажатием клавиши F10. В меню содержатся следующие разделы:

- File. Позволяет выполнять все основные действия с файлами (создание, открытие, сохранение);
- Edit. Позволяет выполнять все основные операции редактирования текста (копирование, вставка, удаление фрагментов, отмена изменений);
- Search. Позволяет осуществлять поиск и замену фрагментов текста;
- Run. Позволяет запускать программу, в том числе в пошаговом режиме;
- Compile. Позволяет осуществлять компиляцию программы;
- Debug. Содержит команды, облегчающие процесс поиска ошибок в программе;
- Tools. Содержит некоторые дополнительные средства Турбо Паскаль;
- Options. Позволяет установить необходимые для работы параметры компилятора и среды разработчика;
- Window. Позволяет выполнять все основные операции с окнами (открывать, закрывать, перемещать, изменять размер);

- Help. Позволяет получить имеющуюся в системе справочную информацию.

Все пункты меню доступны через горячие клавиши. Для этого надо нажать клавишу Alt и ту букву, которая выделена красной в названии пункта меню. Меню также позволяет работать с мышью.

В рабочей области имеется возможность открывать различные окна программы – окна редактируемого текста, окна помощи, отладки и настройки. В вышеприведенном примере открыто только одно окно - окно текста программы. В заголовке окна написано имя файла - исходного текста программы.

Строка состояния демонстрирует некоторые доступные и важные в данный момент операции и соответствующие им комбинации клавиш.

### 1.3.2. Стандартные арифметические функции

Арифметические функции можно использовать только с величинами целого и вещественного типа. Перечень наиболее часто используемых стандартных функций языка Pascal приведен в таблице 1.2.

Таблица 1.2 – Арифметические функции языка Pascal

Функция	Назначение	Тип результата
Abs (x)	Абсолютное значение (модуль) аргумента	Совпадает с типом x
Arctan (x)	Арктангенс аргумента	Вещественный
Cos (x)	Косинус аргумента*	Вещественный
Exp (x)	$e^x$	Вещественный
Ln (x)	Натуральный логарифм	Вещественный
Sin (x)	Синус аргумента	Вещественный
Sqr (x)	Квадрат аргумента	Совпадает с типом x
Sqrt (x)	Квадратный корень аргумента	Вещественный

В языке Turbo Pascal определена функция Pi, возвращающая значение величины  $\pi$ .

В языке Pascal нет функций вычисления тангенса и возведения в степень. Для выполнения таких операций можно воспользоваться следующими математическими преобразованиями:

$$\operatorname{tg} x = \frac{\sin x}{\cos x}, \quad (1.1)$$

$$x^y = e^{y \ln x}, \text{ для } y > 0, \quad (1.2)$$

в частности,

$$\sqrt[y]{x} = e^{\frac{1}{y} \ln x}. \quad (1.3)$$

### 1.3.3. Пример программы

Постановка задачи: Вычислить значение  $a$  по формуле

---

\* Аргументы функций sin(), cos(), а также значение функции arctan() рассматриваются как углы, заданные в радианах

$$a = \frac{\sqrt[3]{|y|}}{(x+1) + (x-1)} + \frac{|\sin x|}{\sqrt[x]{|y|}} \quad (1.4)$$

для  $x = 1,241$ ,  $y = -0,879$ .

Проанализируем формулу (1.4) с целью выявить возможность упростить вычисления. Знаменатель первой дроби можно упростить следующим образом:

$$(x+1) + (x-1) = 2x. \quad (1.5)$$

Чтобы дважды не вычислять  $|y|$ , введем дополнительную переменную

$$z = |y|. \quad (1.6)$$

Тогда с учетом соотношений (1.5), (1.6) алгоритм программы будет иметь вид, показанный на рисунке 1.2.

Предварительный расчет значения  $a$  с помощью калькулятора при заданных значениях  $x$  и  $y$  дает:

$$z = 0,879, \quad \sin x = 0,9461, \\ a = \frac{0,9579}{2,482} + \frac{0,9461}{0,9013} = 0,3859 + 1,0497 = 1,4357. \quad (1.7)$$

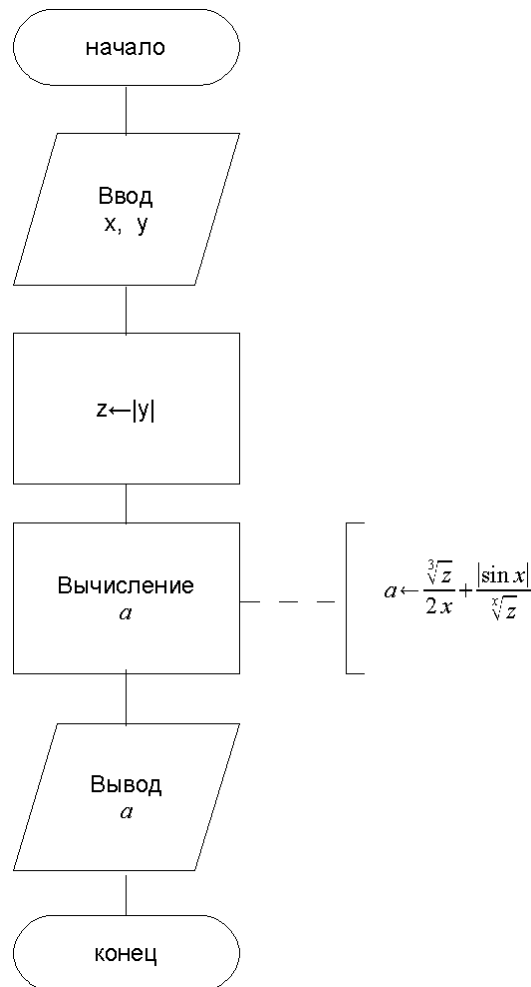


Рисунок 1.2 – Схема программы

Текст программы на языке Pascal будет выглядеть следующим образом:



```

program lr2;
{ФИО:
Вариант:}
var   x, y, z,  a: real;
begin
    read(x, y);
    z  :=  abs(y); {z = |y| - вспомогательная переменная}
    a  :=  exp(ln(z)/3)/2/x + abs(sin(x))/exp(ln(z)/x);
    writeln('Результат: a =', a:5:3); {вывод значения a в 5 позициях,
                                         из них 3 позиции после запятой}
end.

```

При заданных  $x, y$  программа выведет на экран:

Результат:  $a = 1.436$

#### 1.4. Содержание отчета и порядок защиты работы

Выполнение и защита лабораторной работы производится каждым студентом индивидуально. Защита результатов лабораторной работы осуществляется при наличии работающей программы и полностью оформленного отчета.

Отчет должен включать в себя следующие разделы

- титульный лист;
- цель работы,
- постановка задачи. Этот раздел должен состоять из двух подразделов:
  - «Исходные данные», где должны быть приведены номер варианта, формула и значения входных переменных;
  - «Анализ постановки задачи», где должны быть выполнены упрощение вычислений и расчет на калькуляторе с указанием промежуточных результатов;
- схема программы.
- текст программы на языке Pascal;
- результаты работы программы для двух вариантов исходных данных;
- выводы. Этот раздел должен содержать сопоставление результатов вычислений на калькуляторе с результатами работы программы, изложение причины ошибки в случае «плохого» варианта исходных данных, перечень использованных в программе арифметических функций языка Pascal.

Требования к оформлению отчета такие же, как в предыдущей работе.

Защита работы состоит в следующем:

- представление работающей программы на компьютере;
- предъявление отчета, оформленного в соответствии с указанными требованиями;
- ответы на вопросы преподавателя по теоретической и практической части работы. Примеры возможных вопросов приведены в подразделе 1.5.

### 1.5. Контрольные вопросы

- 1) Сформулируйте понятие алгоритма. Перечислите основные требования, предъявляемые к алгоритмам.
- 2) Перечислите основные блоки, применяемые в схемах линейных алгоритмов. Правила изображения схем программ.
- 3) Перечислите разделы программы на языке Pascal.
- 4) Сформулируйте понятие типа данных.
- 5) Приведите основные характеристики типа Integer (типа Real).
- 6) Перечислите виды лексем языка Pascal. Приведите примеры.
- 7) Поясните термин «арифметическое выражение».
- 8) Запишите число 0,000 000 000 043 в экспоненциальном формате по правилам синтаксиса языка Pascal.
- 9) Приоритеты операций при вычислении арифметических выражений?

## 2. ЛАБОРАТОРНАЯ РАБОТА №2. ПРОГРАММИРОВАНИЕ РАЗВЕТВЛЯЮЩИХСЯ АЛГОРИТМОВ НА ЯЗЫКЕ PASCAL

### 2.1. Цель работы

Целью данной работы является исследование разветвляющихся алгоритмов и их программирование с помощью условного оператора языка Pascal. Закрепление навыков программирования ввода и вывода информации. Приобретение начальных навыков тестирования программ.

### 2.2. Задание на работу

- 1) Работа выполняется в среде Turbo Pascal 7.0. Описание лабораторного стенда приведено в методических указаниях к лабораторной работе № 1.
- 2) Изучите пример программы, приведенный в разделе 2.3.
- 2) Создайте программу, реализующую вычисления по формулам в соответствии с номером Вашего варианта (табл. 2.1). Для этого выполните следующие этапы:

– преобразуйте формулы с целью уменьшения количества операций при вычислениях. Упрощение возможно как за счет математических преобразований, так и за счет введения дополнительных переменных для сохранения значений выражений, неоднократно встречающихся в формуле;

– составьте алгоритм (схему) программы. В программе должен быть предусмотрен диалоговый ввод значений входных переменных, вывод результатов с пояснениями. Также должен быть организован вывод значений промежуточных переменных, если таковые имеются;

– **самостоятельно подберите** такие **три варианта исходных данных**, чтобы, запустив программу, можно было проверить правильность выбора каждой из ветвей алгоритма;

– в соответствии с алгоритмом составьте программу на языке Pascal;

– запустите программу на счет при трех вариантах значений входных переменных. Проанализируйте полученные результаты и сообщения среды Turbo Pascal.

Таблица 2.1 – Варианты заданий

№ варианта	Задание
1	$m = \begin{cases} \operatorname{ctg} x^2, & \sin x^2 \geq 0,702, \\ \sqrt{\operatorname{tg} x^2}, & -0,2 < \sin x^2 < 0,702 \\ 1,6 \cdot 10^{-8}, & \sin x^2 \leq -0,2 \end{cases}$
2	$b = \begin{cases} \ln k, & k > \sqrt{2}, \\ -2,6 \cdot 10^3, & -2\sqrt{2} \leq k \leq \sqrt{2} \\ \ln \frac{ k }{\sqrt{2}}, & k < -2\sqrt{2} \end{cases}$
3	$r = \begin{cases} e^{-0,02 m }, & m > \sqrt{ m }, \\ -0,902 \cdot 10^{-12}, & \sqrt{ m } - 12 < m \leq \sqrt{ m }, \\ \sqrt{e^{0,02 m } - \sin \frac{m}{\sqrt{3}}}, & m \leq \sqrt{ m } - 12 \end{cases}$
4	$k = \begin{cases} -0,706 \cdot 10^{-4}, & \operatorname{arctg} z \geq \frac{\pi}{4}, \\ \sin \frac{\operatorname{arctg} z}{2}, & -\frac{\pi}{4} \leq \operatorname{arctg} z < \frac{\pi}{4}, \\ \sin  \operatorname{arctg} z , & \operatorname{arctg} z \leq -\frac{\pi}{4} \end{cases}$
5	$a = \begin{cases} \ln \sqrt{ m - 10^{-6} }, &  m - 10^{-6}  > 0,75, \\ (m - 10^{-6})^4, & 0,2 <  m - 10^{-6}  \leq 0,75, \\  m - 10^{-6}  - 0,3 \cdot 10^{-6}, &  m - 10^{-6}  \leq 0,2 \end{cases}$
6	$t = \begin{cases} -9,001 \cdot 10^3, & j > 0, \\ \sqrt[4]{-j} + \operatorname{tg} j, & -\sqrt{-j} \leq j \leq 0, \\ \sqrt[4]{-j}, & j < -\sqrt{-j}. \end{cases}$
7	$n = \begin{cases} 12,305 \cdot 10^6, & \ln  x  > 10, \\ \frac{\ln^2  x  + \ln  x  + \sqrt{2}}{\sqrt{2} \ln  x  + x^2}, & 1 < \ln  x  \leq 10, \\ \sqrt{\ln  x  + 1}, & \ln  x  \leq 1. \end{cases}$

Таблица 2.1 – Варианты заданий

№ варианта	Задание
8	$i = \begin{cases} -2 \sin^2 x, & \sin^2 x \geq \frac{1}{2}, \\ -1,6 \cdot 10^{-6}, & \frac{1}{4} < \sin^2 x < \frac{1}{2}, \\ \operatorname{tg}^2 x, & \sin^2 x \leq \frac{1}{4}. \end{cases}$
9	$z = \begin{cases} \sqrt{2} \sin^2 n, & \operatorname{arctg}^2 n > 1, \\ -0,068 \cdot 10^{-10}, & 10^{-5} \leq \operatorname{arctg}^2 n \leq 1, \\ \operatorname{tg} \operatorname{arctg}^2 n, & \operatorname{arctg}^2 n < 10^{-5}. \end{cases}$
10	$d = \begin{cases} \sin \pi r^2, & \sqrt{r^2 + 0,6} > 2, \\ -1,22 \cdot 10^{-5}, & \sqrt{1,2} < \sqrt{r^2 + 0,6} \leq 2, \\ \frac{\sqrt{r^2 + 0,6} - \sqrt{0,6 - r^2}}{ r + 0,6  +  r - 0,6 }, & \sqrt{r^2 + 0,6} \leq \sqrt{1,2} \end{cases}$
11	$b = \begin{cases} e^{0,6 \operatorname{arctg} k - \frac{k}{\sqrt{2}}}, &  \operatorname{arctg} k  \geq 1, \\ \operatorname{tg}  \operatorname{arctg} k , & \frac{\pi}{6} \leq  \operatorname{arctg} k  < 1, \\ -10^{-5}, &  \operatorname{arctg} k  < \frac{\pi}{6} \end{cases}$
12	$k = \begin{cases} \sin^2 a + \frac{1}{ a  + 1}, & \sin a + \cos a > 1,6, \\  \cos^2 a - 1  + \sqrt{2} \cos a, & 0 \leq \sin a + \cos a \leq 1,6, \\ 1,825 \cdot 10^{-12}, & \sin a + \cos a < 0. \end{cases}$
13	$f = \begin{cases} \sqrt{1 + \cos\left(\frac{\pi}{2} - x\right)}, & \cos\left(\frac{\pi}{2} - x\right) > 0,5, \\ 0,25 \cdot 10^{-2}, & 0 < \cos\left(\frac{\pi}{2} - x\right) \leq 0,5, \\ \operatorname{tg}^2 \cos\left(\frac{\pi}{2} - x\right), & \cos\left(\frac{\pi}{2} - x\right) \leq 0. \end{cases}$
14	$b = \begin{cases} 2\sqrt{a} + 0,5a, & a > 0, \\ -0,4 \cdot 10^{-5}, & -\sqrt{ a } < a \leq 0, \\ a^5 - 1,3, & -\sqrt{ a } \geq a. \end{cases}$

### 2.3. Пример программы

Условный оператор позволяет проверить некоторое условие и в зависимости от результатов проверки выполнить то или иное действие. Таким образом, условный оператор – это средство ветвления вычислительного процесса.

Рассмотрим следующую постановку задачи: составить программу вычисления кусочно-заданной функции

$$y = \begin{cases} 10^{-8} \sqrt{b^2 + 1}, & \text{при } \sqrt{b^2 + 1} > 2, \\ 0,5 \sqrt{b^2 + 1} & \text{при } \sqrt{b^2 + 1} \leq 2. \end{cases} \quad (2.1)$$

Проанализируем формулу (2.1) с целью выявить возможность упростить вычисления. Чтобы многократно не вычислять  $\sqrt{b^2 + 1}$ , введем дополнительную переменную

$$z = \sqrt{b^2 + 1}. \quad (2.2)$$

Тогда с учетом соотношения (2.2), алгоритм программы будет иметь вид, показанный на рисунке 2.1.

Выберем два таких варианта значений  $b$ , чтобы, произведя вычисления на ЭВМ, можно было проверить правильность выбора каждой из ветвей алгоритма. Произведем предварительный расчет значений  $y$  с помощью калькулятора:

- 1)  $b = 2$ ,  $z = \sqrt{2^2 + 1} = \sqrt{5} = 2,2361 > 2$ ,  $y = 2,2361 \cdot 10^{-8}$ ;
- 2)  $b = 0$ ,  $z = \sqrt{0^2 + 1} = 1 \leq 2$ ,  $y = 0,5$ .

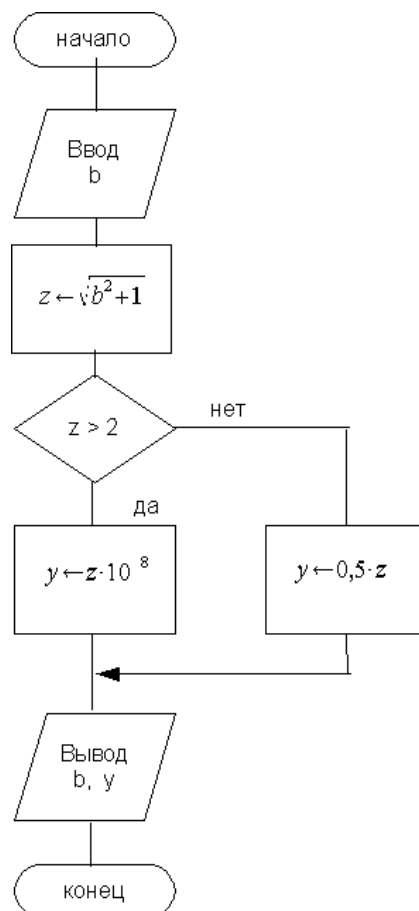


Рисунок 2.1 – Схема программы

Текст программы на языке Pascal будет выглядеть следующим образом:

```

program lr3;
{
ФИО:
Вариант:
}
var b, y, z : real;
begin
  write('b='); readln(b);
  z := sqrt(sqr(b)+1.0);
  if z>2 then
    y:=1e-8*z
  else
    y:=0.5*z;
  writeln(' b=',b:5,' y=',y:10)
end.

```

Результаты работы программы для двух вариантов входных данных:

```

b = 2.0E+00      y =2.236E-08
b = 0.0E+00      y =5.000E-01

```

## 2.4. Содержание отчета и порядок защиты работы

Выполнение и защита лабораторной работы производится каждым студентом индивидуально. Защита результатов лабораторной работы осуществляется при наличии работающей программы и полностью оформленного отчета.

Отчет должен включать в себя следующие разделы

- титульный лист;
- цель работы,
- постановка задачи. Этот раздел должен состоять из двух подразделов:
  - «Вариант задания», где должны быть приведены номер варианта и формула для проведения вычислений;
  - «Выбор исходных данных», где должны быть приведены три варианта значений переменной и обоснование их выбора;
- схема программы;
- текст программы на языке Pascal;
- результаты работы программы для трех вариантов исходных данных;
- выводы.

Защита работы состоит в следующем:

- представление работающей программы на компьютере;
- предъявление отчета, оформленного в соответствии с указанными требованиями;
- ответы на вопросы преподавателя по теоретической и практической части работы. Примеры возможных вопросов приведены в подразделе 2.5.

## 2.5. Контрольные вопросы

- 1) Назовите основные характеристики типа данных `Boolean`.
- 2) Поясните термин «логическое выражение».
- 3) Перечислите операции отношения.
- 4) Изобразите таблицу истинности для логических операций языка `Pascal`.
- 5) Перечислите основные блоки, применяемые в схемах разветвляющихся алгоритмов.
- 6) Изобразите синтаксическую диаграмму условного оператора.
- 7) Изложите алгоритм работы условного оператора.
- 8) Что представляет собой составной оператор?
- 9) Каковы синтаксис и алгоритм работы оператора `case`?
- 10) Сколько будет « $2 \times 2 = 4$ »? :)

## 3. ЛАБОРАТОРНАЯ РАБОТА №3. ПРОГРАММИРОВАНИЕ ЦИКЛИЧЕСКИХ АЛГОРИТМОВ НА ЯЗЫКЕ PASCAL

### 3.1. Цель работы

Целью данной работы является исследование циклических алгоритмов и программ, выполняющих вычисления по рекуррентным формулам, а также закрепление навыков работы в среде `Turbo Pascal`, навыков тестирования программ.

### 3.2. Задание на работу

Работа выполняется в среде `Turbo Pascal 7.0`. Описание лабораторного стенда приведено в методических указаниях к лабораторной работе № 1.

Перед началом работы ознакомьтесь с теоретическими сведениями о рекуррентных формулах арифметических корней, приведенными в пункте 3.3.1 методических указаний.

В лабораторной работе требуется написать *две программы*.

*Первая программа* должна осуществлять вычисление арифметического корня по рекуррентной формуле для одного значения  $x$ .

Программа должна представлять собой циклический алгоритм, в теле которого пересчитывается по рекуррентной формуле значение  $y_k$  при  $k = 1, 2, 3, \dots, n$ . Согласно теории рекуррентных вычислений с ростом  $k$  значение  $y_k$  приближается к точному значению  $y$ . В программе пересчет  $y_k$  должен начинаться с  $y_0 = x$  и повториться  $n$  раз. Также должен быть реализован расчет точного значения функции  $y_{\text{точное}}$  с помощью средств языка `Pascal` (функций `sqrt()`, `exp()`, `ln()`). Кроме того, в программе должен быть предусмотрен диалоговый ввод значений  $n$  и  $x$ . На каждом шаге цикла должен осуществляться вывод  $y_k$  и ошибки вычисления по рекуррентной формуле  $\varepsilon = |y_{\text{точное}} - y_k|$ .

Запустите программу на счет при трех значениях  $x$  из диапазона  $[x_{\min}, x_{\max}]$ , заданного в таблице 3.1. Для каждого значения  $x$  постройте и проанализируйте гра-



фики последовательных приближений значений  $y_k$ , вычисленных первой программой, к точному значению  $y$ .

Вторая программа должна вычислять арифметический корень по рекуррентной формуле при всех значениях  $x$ , изменяющихся с шагом  $\Delta x$  в пределах от  $x_{\min}$  до  $x_{\max}$ , указанных в таблице вариантов заданий. Таким образом, вторая программа представляет собой двойной цикл, причем ее внутренний цикл совпадает с составленным в первой программе, а внешний изменяет значения  $x$ . Программа должна осуществлять вывод точного значения функции  $y_{\text{точное}}$  и приближенного  $y_n$ , а также ошибки  $\varepsilon = |y_{\text{точное}} - y_n|$  для каждого значения  $x$ . Промежуточные значения  $y_k$  выводить не следует. Запустите программу на счет при значениях  $x$ ,  $n$ ,  $x_{\min}$ ,  $x_{\max}$ , заданных в таблице 3.1.

Таблица 3.1 – Варианты заданий

№ варианта	Функция	Рекуррентная формула	Исходные данные			
			$n$	$x_{\min}$	$x_{\max}$	$\Delta x$
1	$y = \frac{1}{\sqrt{x}}$	$y_k = 0,5 \cdot y_{k-1} (3 - xy_{k-1}^2)$	12	0,6	2,1	0,5
2	$y = \sqrt[4]{x^3}$	$y_k = -0,25y_{k-1} + (y_{k-1} + x^3/4y_{k-1}^3)$	12	0,2	2,6	0,8
3	$y = \sqrt[3]{x}$	$y_k = 1/3 \cdot (2y_{k-1} + x/y_{k-1}^2)$	12	0,7	1,6	0,3
4	$y = \sqrt[3]{x^2}$	$y_k = y_{k-1} + 1/3 \cdot (x^2/y_{k-1}^2 - y_{k-1})$	11	0,9	2,4	0,5
5	$y = \sqrt{x}$	$y_k = y_{k-1} (3/2 - y_{k-1}^2/2x)$	12	0,8	2,4	0,4
6	$y = \sqrt{x}$	$y_k = \frac{y_{k-1}(y_{k-1}^2 + 3x)}{3y_{k-1}^2 + x}$	10	0,5	2,1	0,4
7	$y = \sqrt[3]{x}$	$y_k = \frac{1}{2} y_{k-1} + \frac{1,5x}{2y_{k-1}^2 + x/y_{k-1}}$	10	0,1	2,5	0,6
8	$y = \sqrt[3]{x}$	$y_k = y_{k-1} + 1/3 \cdot (x/y_{k-1}^2 - y_{k-1})$	11	0,7	2,2	0,5
9	$y = \frac{1}{\sqrt[3]{x}}$	$y_k = y_{k-1} + 1/3 \cdot (y_{k-1} + xy_{k-1}^4)$	10	0,8	2,0	0,2
10	$y = \frac{1}{\sqrt[3]{x}}$	$y_k = 1/3 \cdot y_{k-1} (4 - xy_{k-1}^3)$	12	0,9	3,0	0,7
11	$y = \sqrt[4]{x}$	$y_k = y_{k-1} - 0,25(y_{k-1} - x/y_{k-1}^3)$	10	1,1	2,3	0,3
12	$y = \frac{1}{\sqrt[4]{x}}$	$y_k = y_{k-1} + 0,25(y_{k-1} - xy_{k-1}^5)$	12	0,9	3,0	0,7
13	$y = \sqrt[5]{x}$	$y_k = y_{k-1} - \frac{y_{k-1}^5 - x}{5y_{k-1}^4}$	10	0,5	1,2	0,1
14	$y = \sqrt[5]{x}$	$y_k = 0,2(4y_{k-1} + x/y_{k-1}^4)$	12	0,6	2,0	0,2

Во всех вариантах начальное приближение  $y_0 = x$ .

### 3.3. Краткие теоретические сведения

#### 3.3.1. Рекуррентные формулы вычисления значений функций

В различных алгоритмах численного решения задач часто используются итерационные методы. **Метод итераций (метод последовательных приближений)** – метод решения математических задач с помощью построения последовательности, сходящейся к искомому решению, начиная с некоторого начального приближения. При этом члены последовательности вычисляются повторным применением какой-либо операции (итерациями).

Иными словами, итерационные методы – это методы, в которых точное решение может быть получено лишь в результате бесконечного повторения единообразных (как правило, простых) действий.

В итерационных алгоритмах часто применяются так называемые рекуррентные формулы. Рекуррентные формулы служат для получения последовательности значений. При этом каждое следующее значение вычисляется по предшествующим значениям с помощью одной и той же формулы.

*Рекуррентная формула* — формула вида  $a_n = f(n, a_{n-1}, a_{n-2}, \dots, a_{n-p})$ ,  $n \geq p + 1$ , выражающая каждый член последовательности  $a_n$  ( $n \in \mathbb{N}$ ) через  $p$  предыдущих членов.

В качестве конкретного примера применения такого подхода приведем *правило Ньютона вычисления арифметических корней*.

Пусть  $x$  – данное положительное число, а  $n \geq 2$  – данный натуральный показатель корня. Ставится задача вычислить вещественное значение

$$y = \sqrt[n]{x} \quad (3.1)$$

путем последовательных приближений с помощью арифметических операций.

Согласно правилу Ньютона процесс приближений к  $\sqrt[n]{x}$  определяется формулой

$$y_{k+1} = y_k - \frac{y_k^n - x}{n y_k^{n-1}} \quad (3.2)$$

или в другом виде

$$y_{k+1} = \frac{1}{n} \left[ (n-1)y_k - \frac{x}{y_k^{n-1}} \right], \quad (3.3)$$

где  $k = 0, 1, 2, \dots$ , а  $y_0 > 0$  задается.

Еще из глубокой древности известен частный случай правила Ньютона – *процесс Герона*

$$y_{k+1} = \frac{1}{2} \left( y_k + \frac{x}{y_k} \right), \quad (3.4)$$

применяемый для извлечения квадратных корней

$$y = \sqrt{x}. \quad (3.5)$$

### 3.3.2. Пример программы

Необходимо составить программу для приближенного вычисления значения функции  $y = \sqrt{x}$  с помощью рекуррентной формулы  $y_k = \frac{1}{2} \left( y_{k-1} + \frac{x}{y_{k-1}} \right)$ , для заданного  $x$ . Алгоритм решения задачи можно представить с помощью схемы, показанной на рисунке 3.1.

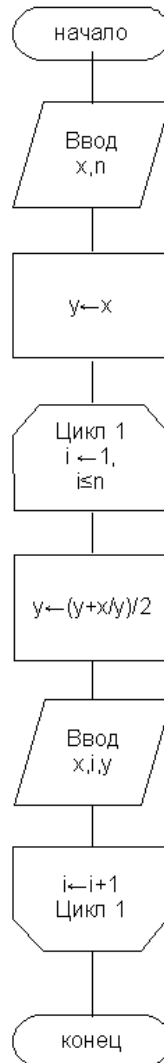


Рисунок 3.1 – Схема программы

Тогда текст программы будет иметь вид:

```

programm lr4;
var x, y : real; i, n : integer;
begin
  write('x='); readln(x);
  write('n='); readln(n);
  i:=1; y:=x;
  while i <= n do
    begin
      y:=(y+x/y)/2;
      writeln('x=',x, ' i=', i:2, ' y=', y); i := i + 1;
    end;end.

```

### 3.4. Содержание отчета и порядок защиты работы

Выполнение и защита лабораторной работы производится каждым студентом индивидуально. Защита результатов лабораторной работы осуществляется при наличии работающей программы и полностью оформленного отчета.

Отчет должен включать в себя следующие разделы

- титульный лист;
- цель работы,
- постановка задачи – этот раздел должен содержать вариант задания в соответствии с таблицей 3.1;
- схемы программ;
- тексты программ на языке Pascal;
- результаты работы программ – этот раздел должен состоять из двух подразделов. Подраздел «Результаты работы первой программы» должен содержать таблицы значений результатов и графики для трех вариантов  $x$  в соответствии с заданием из подраздела 3.2 настоящих методических указаний. Подраздел «Результаты работы второй программы» должен содержать таблицу значений результатов в соответствии с заданием из подраздела 3.2 настоящих методических указаний.
- выводы.

Защита работы состоит в следующем:

- представление работающей программы на компьютере;
- предъявление оформленного в соответствии требованиями отчета;
- ответы на вопросы преподавателя по теоретической и практической части работы. Примеры возможных вопросов приведены в подразделе 3.5.

### 3.5. Контрольные вопросы

- 1) Опишите синтаксис и алгоритм работы оператора цикла с предусловием в языке Pascal.
- 2) Опишите синтаксис и алгоритм работы оператора цикла с постусловием в языке Pascal.
- 3) Опишите синтаксис и алгоритм работы оператора цикла с параметром в языке Pascal.
- 4) От каких исходных данных зависит время выполнения составленных в лабораторной работе программ, и каким образом?
- 5) . Какие блоки в двойном циклическом алгоритме выполняются наибольшее число раз при любых исходных данных?

### БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Пильщиков В.Н. Сборник упражнений по языку Pascal: Учебное пособие для вузов / В.Н. Пильщиков. – М.: Наука, 1989г. – 160с.
2. Павловская Т. А. Паскаль: программирование на языке высокого уровня: практикум [Текст] : учеб. пособие для студ. вузов / Т. А. Павловская. – М. и др.: Питер, 2006. – 408 с.

3. Павловская Т. А. Паскаль: программирование на языке высокого уровня [Текст] : учеб. для студ. вузов / Т. А. Павловская. – М. и др. : Питер, 2006. – 400 с.

**ПРИЛОЖЕНИЕ А.**  
**ПРИМЕР ОФОРМЛЕНИЯ ТИТУЛЬНОГО ЛИСТА ОТЧЕТА**  
(справочное)

Министерство образования и науки РФ  
Севастопольский государственный университет  
Кафедра информатики и управления в технических системах

**ОТЧЕТ**  
по лабораторной работе №1  
**ПРОГРАММИРОВАНИЕ ЛИНЕЙНЫХ АЛГОРИТМОВ**  
**НА ЯЗЫКЕ PASCAL**

Выполнил: студент гр. УТС/б-11о  
Иванов И.И.

Проверил: ст. преподаватель  
кафедры ИУТС  
Захаров В.В.

Севастополь

2017

## ПРИЛОЖЕНИЕ Б. ПРИМЕР ОФОРМЛЕНИЯ ОСНОВНОЙ ЧАСТИ ОТЧЕТА (справочное)

### 1 ЦЕЛЬ РАБОТЫ

Целью данной работы является изучить и практически освоить базовые приемы работы в среде Turbo Pascal.

### 2 ПОСТАНОВКА ЗАДАЧИ

Написать программу вывода на экран инициалов с помощью символов \*.

### 3 СХЕМА ПРОГРАММЫ

Схема программы приведена на рисунке 3.1.

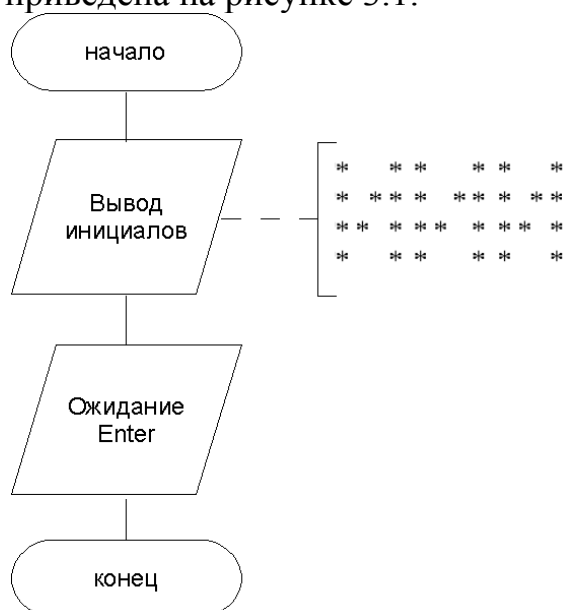


Рисунок 3.1 – Схема программы

### 4 ТЕКСТ ПРОГРАММЫ

```

Program MyName;
begin
  writeln('* * * * *');
  writeln('* * * * *');
  writeln('* * * * *');
  writeln('* * * * *');
  readln;
end.
  
```

### 5 РЕЗУЛЬТАТЫ РАБОТЫ ПРОГРАММЫ

Программа выводит на экран изображение, показанное на рисунке 5.1.

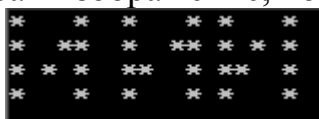


Рисунок 5.1 – Результат работы программы

### ВЫВОДЫ

В рамках лабораторной работы разработана программа вывода на экран инициалов с помощью символов \*. .....

**ПРИЛОЖЕНИЕ В.**  
**ПЕРЕЧЕНЬ ТЕМ БЛОКА №1**  
(справочное)

1. Понятие алгоритма. Свойства и классификация алгоритмов.
2. Общая схема работы ЭВМ. Архитектура фон Неймана.
3. Основные этапы программирования.
4. Единицы измерения информации: бит, байт и пр. Перевод десятичных чисел в двоичную систему и наоборот.
5. Простейшая программа на языке Pascal. Структура программы. Трассировка программы.
6. Переменные и константы в языке Pascal. Разделы описания переменных и констант. Понятие типа данных.
7. Понятие типа данных. Данные целого типа в языке Pascal.
8. Понятие типа данных. Данные вещественного типа в языке Pascal. Особенности машинной арифметики.
9. Раздел операторов в программе на языке Pascal. Оператор присваивания.
10. Арифметические выражения. Арифметические операции. Математические функции языка Pascal.
11. Введение в синтаксис языка Pascal. Лексемы и разделители.
12. Правила составления синтаксических диаграмм. Примеры синтаксических диаграмм.
13. Разветвляющиеся алгоритмы, условный оператор в языке Pascal. Составной оператор.
14. Данные логического типа в языке Pascal. Логические операции и выражения в языке Pascal.
15. Оператор выбора варианта case.
16. Циклические алгоритмы. Оператор цикла с предусловием в языке Pascal. Пример.
17. Циклические алгоритмы. Оператор цикла с постусловием в языке Pascal. Пример.
18. Циклические алгоритмы. Оператор цикла с параметром в языке Pascal. Пример.
19. Циклические алгоритмы. Пример цикла с заданным числом повторений.
20. Процедуры Break и Continue в Turbo Pascal.
21. Оператор goto. Использование меток. Пустой оператор.

**ПРИЛОЖЕНИЕ Г.**  
**ПРИМЕР ЗАДАНИЯ ПО ТЕМАТИКИ БЛОКА №1**  
(справочное)

**Задание №1** (5 баллов)

**В условном операторе после ключевого слова `if` может находиться:**

- 1) Любой оператор языка Pascal;
- 2) Вложенный условный оператор;
- 3) Выражение логического типа;
- 4) Раздел описания переменных;
- 5) Оператор присваивания.

**Номер правильного ответа:** \_\_\_\_\_

**Задание №2** (5 баллов)

**Какое из нижеследующих утверждений неверно:**

- 1) Результат операции сложения целых чисел – целый;
- 2) Результат вычисления стандартных тригонометрических функций с аргументами целого типа – вещественный;
- 3) К величинам типа Boolean применимы операции `not` и `or`;
- 4) В разделе описания констант можно описать и константы, и переменные;
- 5) Результат операции деления целых чисел – вещественный.

**Номер правильного ответа:** \_\_\_\_\_

**Задание №3** (5 баллов)

**Чему будет равно значение переменной `m` после выполнения программы?**

```
Program Task3;  
Var k, m: Integer;  
Begin  
  k:=0; m:=0;  
  while k<3 do  
    k:=k+1;  
    m:=m+1;  
  end.
```

- 1) 0;
- 2) 3;
- 3) 4
- 4) 2;
- 5) 1.

**Номер правильного ответа:** \_\_\_\_\_

**Задание №4**

(Максимальная оценка 15 баллов)



Отметьте (обведите, подчеркните) синтаксическую ошибку (или ошибки) в приведенной программе. Напишите тот вариант программы, который, по Вашему мнению, будет правильным.

```

Program 4,
Var a:=2, b:=3 : Real;
Begin
  If b <= 4 then
    c := 0;
  else
    c := 1;
  write(c );
end;

```

### Задание №5

(Максимальная оценка 20 баллов)

Для вводимых с клавиатуры вещественных  $x, a, b, c$  найти значение следующей функции  $f$ , принимающей вещественные значения:

$$f = \begin{cases} \frac{1}{ax} - b & \text{при } x+5 < 0 \text{ и } c = 0 \\ \frac{x-a}{x} & \text{при } x+5 > 0 \text{ и } c \neq 0 \\ \frac{10x}{c-4} & \text{в остальных случаях} \end{cases}$$