**МИНОБРНАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «ТУЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

Институт прикладной математики и компьютерных наук

Кафедра информационной безопасности

**ЯЗЫКИ ПРОГРАММИРОВАНИЯ**

Отчет по выполнению лабораторной работы № 1

Выполнила \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ст. гр.230711 Павлова Виктория Сергеевна

Проверила \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

доц. каф. ИБ Басалова Галина Валерьевна

Тула 2022

# Оглавление

[Оглавление 2](#_Toc96166223)

[Лабораторная работа № 1. Составление программ на языке Паскаль с использованием различных управляющих операторов 3](#_Toc96166224)

[Цель работы: 3](#_Toc96166225)

[Задания на работу: 3](#_Toc96166226)

[Ход работы: 3](#_Toc96166227)

[7](#_Toc96166228)

[Вывод: 7](#_Toc96166229)

# Лабораторная работа № 1. Составление программ на языке Паскаль с использованием различных управляющих операторов

## Цель работы:

Повторение принципов использования различных управляющих конструкций языка Паскаль.

## Задания на работу:

1. Изучить теоретические положения и конспект лекций по данной теме.

2. Запустить интегрированную систему PascalABC или FreePascal.

3. Набрать текст программы из *примера 1*. Сохранить созданную программу. Запустить на выполнение созданную программу. Посмотреть и проверить результаты работы программы.

4. Для заданного варианта для каждой задачи: разработать алгоритм решения задачи. Составить программу, реализующую разработанный алгоритм. Сохранить файл, проверить работу программы.

5. Показать результаты работы программы преподавателю.

6. Составить отчет по работе.

Для каждой задачи по своему варианту: описание разработанного алгоритма (постановка задачи, исходные данные, требуемые результаты, схема алгоритма, используемые переменные и структуры данных); текст созданной программы; результаты выполнения разработанной программы для разных наборов исходных данных.

## Ход работы:

1. Проверю работу программы из *примера 1.*

Входные данные: N – количество членов последовательности, x1,x2 – их значения.  
Выходные данные: вид последовательности.

|  |  |  |
| --- | --- | --- |
| **Тест программы №1** Входные данные: N=5, нулевая последовательность  Ожидаемый результат: не возрастает Результат работы программы: | **Тест программы №1** Входные данные: последовательность [0, 2, 4, 6, 8, 10, 12 ], N=7  Ожидаемый результат: возрастает Результат работы программы: | **Тест программы №1** Входные данные: N=5, смешанная последовательность [ 0, 1, 0, 1, 0 ]  Ожидаемый результат: не возрастает Результат работы программы: |

Программа из примера 1 работает корректно.  
  
2. Задания по варианту:

**Задача №19**: Дано: натуральные числа n (n<1000000) и m (m<10). Проверить, есть ли в записи числа n цифра m.

|  |  |
| --- | --- |
| Начало  Ввод n,m  y = n  n > 0  x = остаток от деления n на 10;  n = целое от деления n на 10;  Цифра m ЕСТЬ в записи числа y  x = m  да  n ≤ 0  да  Цифра m НЕТ в записи числа y  Конец | Код программы:  **var** m,n,y,x:integer;  **begin**  readln(n,m);  y:=n;  **while** n>0 **do**  **begin**  x:=n **mod** 10;  n:=n **div** 10;  **if** x=m **then**  **begin**  writeln('Цифра ',m,' ЕСТЬ в записи числа ',y);  **break**;  **end**;  **end**;  **if** n<=0 **then** writeln('Цифры ',m,' НЕТ в записи числа',y);  **end**. |

|  |  |  |
| --- | --- | --- |
| **Тест программы №1:** Входные данные: целые n=12123, m=5.  Ожидаемый результат: цифры m нет в записи числа n  Результат работы программы: | **Тест программы №2:** Входные данные: целые n=452, m=2.  Ожидаемый результат: цифра m есть в записи числа n  Результат работы программы: | **Тест программы №3:** Входные данные: целые n=999999, m=9.  Ожидаемый результат: цифра m есть в записи числа n  Результат работы программы: |

**Задача №35**: Представить в виде обыкновенной несократимой дроби сумму заданных А (А<=1000) обыкновенных дробей. Входные данные: целое число А, затем А числителей и знаменателей дробей-слагаемых. Каждый числитель – целое число, не превосходящее по модулю 1000, знаменатель – натуральное число, не большее 1000. Выходные данные: числитель и знаменатель суммы.

|  |  |
| --- | --- |
| Схема:  Начало  Ввод a, x, y  sum=x1,  znam=y1  i=1  x1 = x,  y1 = y  Ввод x,y  от i до a-1    Конец  p=sum, q=znam  Дробь до сокращения  r = ост. от ; p = q, q = r  r=0  нет  да  Дробь  после сокр-я  sum=sum+x  sum=sum\*y + + x\*y1  znam=znam\*y; x1=x; y1=znam;  да  y=y1  нет | Код программы:  **var** a,x,y,i,x1,y1,sum,znam,p,q,r:integer;  **begin**  readln(a);  write('числитель:'); readln(x);  write('знаменатель:'); readln(y);  x1:=x; y1:=y;  sum:=x1; znam:=y1;  **for** i:=1 **to** a-1 **do**  **begin**  write('числитель:');  readln(x);  write('знаменатель:');  readln(y);  **if** y=y1 **then**  **begin**  sum:=sum+x;  **end**  **else begin**  sum:=sum\*y+x\*y1;  znam:=znam\*y;  x1:=x;  y1:=znam;  **end**;  **end**;  writeln(sum,'/',znam);  p:=sum;  q:=znam;  **repeat**  r:=p **mod** q;  p:=q; q:=r;  **until** r=0;  writeln(sum **div** p,'/',znam **div** p);  **end**. |

|  |  |  |
| --- | --- | --- |
| **Тест программы №1:**  Входные данные: a – кол-во дробей (a=4); целые x – их числители, не превосходящие по модулю 1000, а натуральные y – их знаменатели, не большие 1000: .  Ожидаемый результат: сокращённый числитель и знаменатель дроби, равной сумме введённых дробей: .  Результат работы программы: | **Тест программы №2:**  Входные данные: a – кол-во дробей (a=4); целые x – их числители, не превосходящие по модулю 1000, а натуральные y – их знаменатели, не большие 1000: .  Ожидаемый результат: сокращённый числитель и знаменатель дроби, равной сумме введённых дробей: .  Результат работы программы: | **Тест программы №3:**  Входные данные: a – кол-во дробей(a=3); целые x – их числители, не превосходящие по модулю 1000, а натуральные y – их знаменатели, не большие 1000: .  Ожидаемый результат: сокращённый числитель и знаменатель дроби, равной сумме введённых дробей: .  Результат работы программы: |

**Задача №66**: Дано натуральное число n (n < 100). Определить число способов

выплаты суммы n рублей с помощью монет достоинством 1, 2, 5 рублей и

бумажных купюр достоинством 10 рублей. Вывести на экран все способы

выплаты.

Код программы:

**var** n,x10,x5,x2,x1,c: integer;

**begin**

readln(n); //cумма

c:=0;

writeln('Варианты: ');

**for** x10:=0 **to** (n **div** 10) **do**

**for** x5:=0 **to** (n - x10\*10) **div** 5 **do**

**for** x2:=0 **to** (n-x10\*10 - x5\*5) **div** 2   
 **do begin**

inc(c);

x1:=n-x10\*10-x5\*5-x2\*2;

writeln('10\*', x10, ' + 5\*', x5, ' + 2\*', x2, ' + 1\*', x1);

**end**;

writeln('Количество вариантов: ', c);

**end**.

|  |  |
| --- | --- |
| Схема программы:  c=c+1;  Выв. ‘+2\*’ +1\*  Начало  Ввод n  с = 0, =0, =0, =0;  oт до целого от n/10  oт до целого  от (n-\*10)/5  oт до целого  от (n-\*10-  \*5)/2  Вывод кол-ва  вариантов  Конец | **Тест программы №1:**  Входные данные: n – целое число рублей, n<100, n = 5. Выходные данные: c = 4 – количество способов выплаты суммы (и сами способы): 1) 1+1+1+1+1=5\*1; 3) 5;  2) 2+1+1+1=2+1\*3; 4) 2+2+1=2\*2+1 Результат работы программы:  **Тест программы №2:** Входные данные: n – целое число рублей, n<100, n = 10. Выходные данные: c = 11 – количество способов выплаты суммы (и сами способы): 1) 1+1+1+1+1+1+1+1+1+1=10\*1;  2) 2+1+1+1+1+1+1+1+1=2\*1+1\*8;  3) 2+2+1+1+1+1+1+1=2\*2+1\*6;  4) 2+2+2+1+1+1+1=12\*3+1\*4;  5) 2+2+2+2+1+1=2\*4+1\*2;  6) 2+2+2+2+2=2\*5;  7) 5+1+1+1+1+1=5\*1+1\*5  8) 5+2+1+1+1=5\*1+2\*1+1\*3  9) 5+2+2+1=5\*1+2\*2+1\*1  10) 5+5=5\*2; 11) 10;  Результат работы программы: |

## Вывод:

Я повторила принципы использования различных управляющих конструкций языка Паскаль.