

## **ОТЧЕТ О ВЫПОЛНЕНИИ ЛАБАРАТОРНОЙ РАБОТЫ**

«Линейные процессы обработки данных»

студента 2 курса, КС-191 группы

Агапов Евгений Викторович

Направление 09.02.03 – «Программирование в компьютерных системах»

Руководитель:  
преподаватель информационных дисциплин  
М.В. Валеев

---

Работа защищена

« \_\_\_\_\_ »  
“ \_\_\_\_\_ ” 20 \_\_\_\_ г.

## СОДЕРЖАНИЕ

1 Задача 1 .....	4
1.1 Постановка задачи .....	4
1.2 Структурные требования .....	4
1.3 Описание работы .....	4
1.3.1 Используемые переменные .....	4
1.3.2 Псевдокод программы .....	4
1.3.4 Листинг программной реализации .....	5
1.3.5 Результаты .....	6
1.3.6 Таблица .....	6
2 Задача 2 .....	7
2.3.3 Блок-схема алгоритма .....	8
2.2.4 Листинг программной реализации .....	9
2.3.5 Результаты .....	9
2.3.6 Таблица .....	10
3 Задача 3 .....	11
3.1 Постановка задачи .....	11
3.2 Структурные требования .....	11
3.3 Описание работы .....	11
3.3.1 Используемые переменные .....	11
3.3.2 Псевдокод программы .....	12
3.3.4 Листинг программной реализации .....	14
3.3.5 Результаты .....	14
3.3.6 Таблица .....	15
4 Задача 4 .....	16
4.1 Постановка задачи .....	16
4.2 Структурные требования .....	16
4.3 Описание работы .....	16
4.3.1 Используемые переменные .....	16
4.3.2 Псевдокод программы .....	16
4.3.3 Блок-схема алгоритма .....	17
4.3.4 Листинг программной реализации .....	18
4.3.5 Результат .....	18
4.3.6 Таблица .....	19
5 Задача 5 .....	20

5.1 Постановка задачи.....	20
5.2 Структурные требования .....	20
5.3 Описание работы .....	20
5.3.1 Используемые переменные .....	20
5.3.2 Псевдокод программы.....	20
5.3.3 Блок-схема алгоритма .....	21
5.3.4 Листинг программной реализации .....	22
5.3.5 Результат .....	22
5.3.6 Таблица .....	23
6 Задача 6 .....	24
6.1 Постановка задачи.....	24
6.2 Структурные требования .....	24
6.3 Описание работы .....	24
6.3.1 Используемые переменные .....	24
6.3.2 Псевдокод программы .....	24
6.3.3 Блок-схема алгоритма.....	25
6.3.4 Листинг программной реализации.....	25
6.3.5 Результат .....	26
6.3.6 Таблица.....	26
7 Задача 7 .....	27
7.1 Постановка задачи.....	27
7.2 Структурные требования .....	27
7.3 Описание работы .....	27
7.3.1 Используемые переменные .....	27
7.3.2 Псевдокод алгоритма.....	27
7.3.3 Блок-схема алгоритма.....	27
7.3.4 Листинг программной реализации.....	28
7.3.5 Результат .....	29
7.3.6 Таблица.....	30

# 1 Задача 1

## 1.1 Постановка задачи

Вычислите значение функции  $y = (e^{-x_1} + e^{-x_2})/2$

## 1.2 Структурные требования

Особых структурных требований нет.

## 1.3 Описание работы

### 1.3.1 Используемые переменные

$x_1$  – первая степенная переменная

$x_2$  – вторая степенная переменная

$y$  – результат

### 1.3.2 Псевдокод программы

Начало

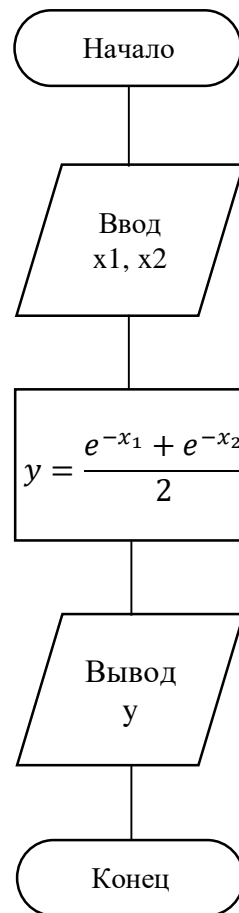
Ввод переменных  $x_1$  и  $x_2$

$$y = (e^{-x_1} + e^{-x_2})/2$$

Вывод  $y$

Конец

### 1.3.3 Блок-схема алгоритма

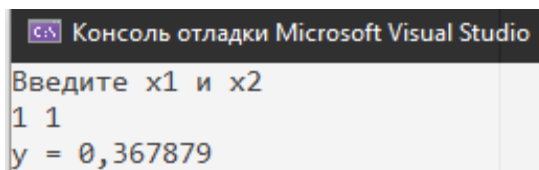


### 1.3.4 Листинг программной реализации

```
#include <stdio.h>
#include <math.h>
#include <conio.h>
#include <locale>

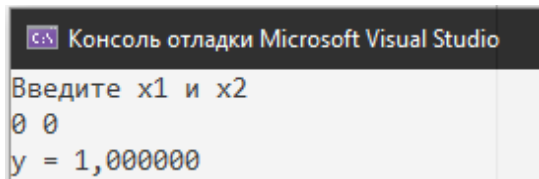
int main()
{
    setlocale(0, "rus");
    float x1, x2, y;
    printf("Введите x1 и x2\n");
    scanf_s("%f%f", &x1, &x2);
    y=(exp(-x1) + exp(-x2))/2;
    printf("y=%f", y);
    getch();
    return 0;
}
```

### 1.3.5 Результаты



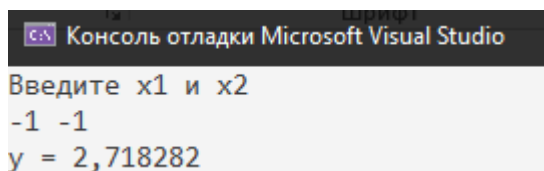
```
Консоль отладки Microsoft Visual Studio
Введите x1 и x2
1 1
y = 0,367879
```

Рисунок 1 - Положительные значения x



```
Консоль отладки Microsoft Visual Studio
Введите x1 и x2
0 0
y = 1,000000
```

Рисунок 2 - Нейтральные значения x



```
Консоль отладки Microsoft Visual Studio
Введите x1 и x2
-1 -1
y = 2,718282
```

Рисунок 3 - Отрицательные значения x

### 1.3.6 Таблица

№	Переменные		Программный расчет	Ручной Расчет
	X1	X2		
1	1	1	0.367878	1/e
2	0	0	1.000000	1
3	-1	-1	2.718282	e

1. Возьмем значение  $x1=1$ , а значение  $x2=1$ , результат программы будет равен 0.367878, результат ручного расчета равен  $1/e$ .
2. Возьмем значения  $x1=0$ , а значение  $x2=0$ , результат программы будет равен 1 и результат ручного расчета 1.
3. Возьмем значения  $x1=-1$ , а значение  $x2=-1$ , результат программы будет равен 2.718282, а результат ручного расчета e.

## 2 Задача 2

### 2.1 Постановка задачи

Вычислить площадь треугольника со сторонами  $a$ ,  $b$ ,  $c$  по формуле Герона

$$s = \sqrt{p * (p - a) * (p - b) * (p - c)}$$
$$p = (a + b + c)/2$$

### 2.2 Структурные требования

$a < b + c$ ,  $b < a + c$ ,  $c < a + b$ ,  $a \geq 1$ ,  $b \geq 1$ ,  $c \geq 1$ .

### 2.3 Описание работы

#### 2.3.1 Используемые переменные

$a$  – Сторона треугольника  
 $b$  – Сторона треугольника  
 $c$  – Сторона треугольника  
 $p$  – Периметр треугольника  
 $s$  – Площадь треугольника

#### 2.3.2 Псевдокод программы

Начало

Ввод переменных  $a$ ,  $b$ ,  $c$ ,  $p$ ,  $s$

Если  $a > 0$  и  $a < b + c$  и  $b > 0$  и  $b < a + c$  и  $c > 0$  и  $c < a + b$

То  $p = (a + b + c)/2$ .

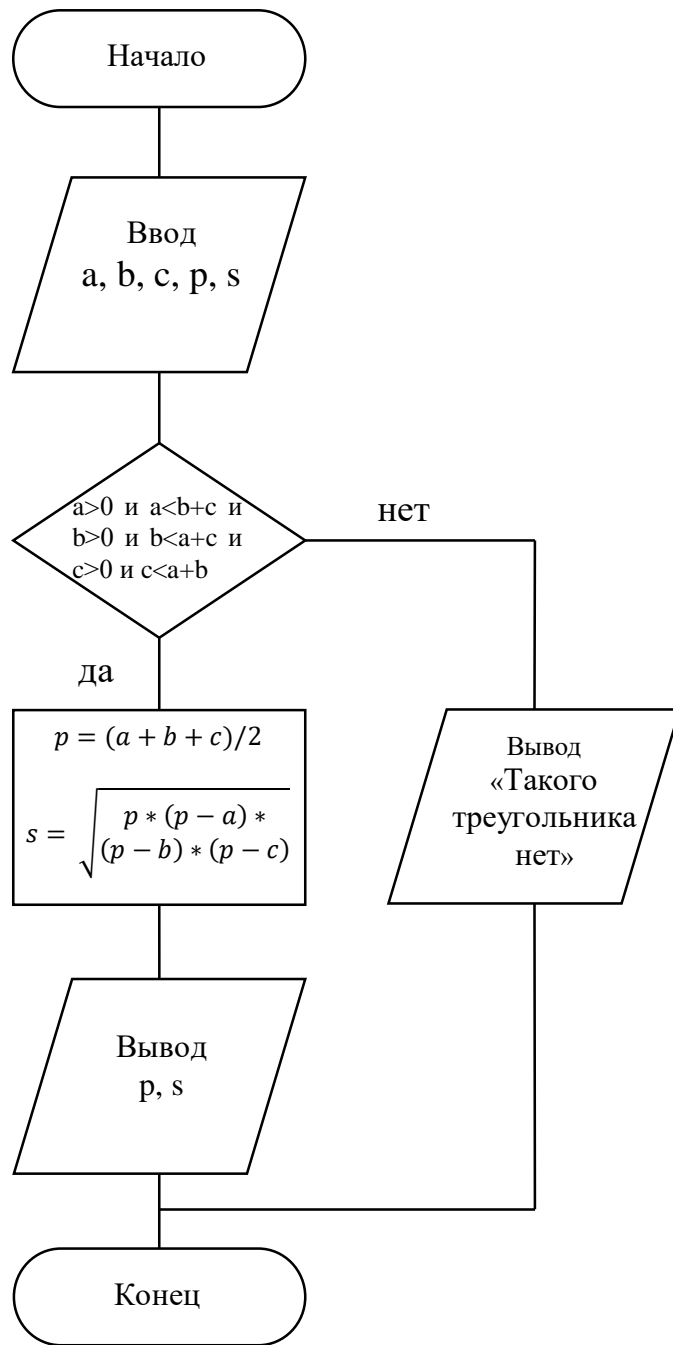
$$s = \sqrt{p * (p - a) * (p - b) * (p - c)},$$

Вывод  $p$ ,  $s$

Иначе вывод «Такого треугольника нет»

Конец

### 2.3.3 Блок-схема алгоритма





#### 2.2.4 Листинг программной реализации

```
#include <stdio.h>
#include <conio.h>
#include <clocale>
#include <math.h>

int main()
{
    setlocale(0, "rus");
    float a, b, c, s, p;
    printf("Введите стороны треугольника\n");
    scanf_s("%f%f%f", &a, &b, &c);
    if (a <= 0 || a >= b + c || b <= 0 || b >= a + c || c >= 0 || c >= a + b)
        printf("Такого треугольника нет\n");
    else
    {
        p = (a + b + c) / 2;
        s = sqrt(p * (p - a) * (p - b) * (p - c));
        printf("p=%f\ns=%f", p, s);
    }
    _getch();
    return 0;
}
```

#### 2.3.5 Результаты

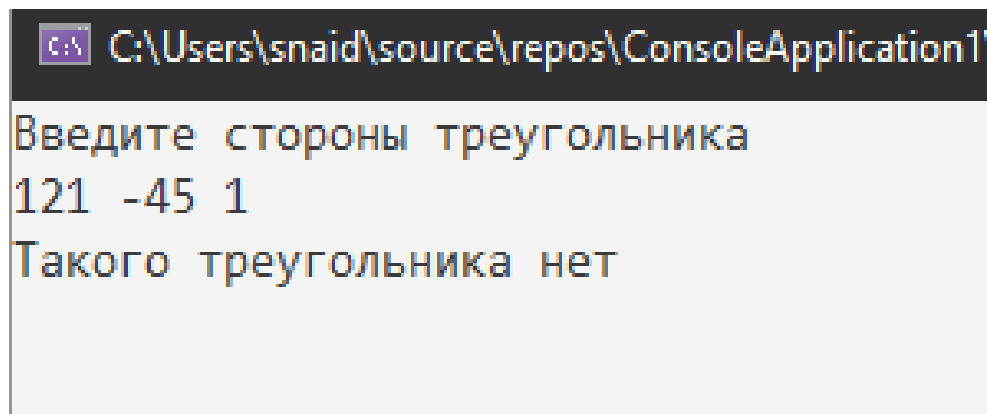


Рисунок 4 - Одна сторона с отрицательным значение

```

C:\Users\snaid\source\repos\ConsoleApplica
Введите стороны треугольника
5 22 4
Такого треугольника нет

```

Рисунок 5 - Неверное соотношение сторон

```

Введите стороны треугольника
18 9 14
p=20,500000; s=61,894569

```

Рисунок 6 - Верный результат

### 2.3.6 Таблица

№	Переменные			Программный	Ручной
	a	b	c	Расчет	Расчет
1	121	-4	1	Такого треугольника нет	Такого треугольника нет
2	5	22	4	Такого треугольника нет	Такого треугольника нет
3	18	9	14	p=20,500000; s= 61,894569	p=20; s= 62

1. Возьмем значение переменных a=121, b=-4, c=1 неверные значение переменных.
2. Возьмем значение переменных a=5, b=22, c=4 неверные значения переменных.
3. Возьмем значение переменных a=18, b=9, c=14 результат программного расчета p=20,500000; s= 61,894569 результат ручного расчета равен p=20; s= 62

### 3 Задача 3

#### 3.1 Постановка задачи

Вычислить высоты треугольника со сторонами  $a$ ,  $b$ ,  $c$  по формулам

$$s = \sqrt{p * (p - a) * (p - b) * (p - c)},$$

$$p = (a + b + c)/2,$$

$$h_a = 2 * s/a,$$

$$h_b = 2 * s/b,$$

$$h_c = 2 * s/c.$$

#### 3.2 Структурные требования

$a > 0$  и  $a < b + c$  и  $b > 0$  и  $b < a + c$  и  $c > 0$  и  $c < a + b$

#### 3.3 Описание работы

##### 3.3.1 Используемые переменные

$a$  – сторона треугольника

$b$  – сторона треугольника

$c$  – сторона треугольника

$p$  – периметр треугольника

$s$  – площадь треугольника

$h_a$  – высота  $a$

$h_b$  – высота  $b$

$h_c$  – высота  $c$

### 3.3.2 Псевдокод программы

Начало

Ввод переменных  $a, b, c, h_a, h_b, h_c$

Если  $a > 0$  и  $a < b + c$  и  $b > 0$  и  $b < a + c$  и  $c > 0$  и  $c < a + b$

То  $p = (a + b + c)/2,$

$$s = \sqrt{p * (p - a) * (p - b) * (p - c)}$$

$$h_a = 2 * s / a$$

$$h_b = 2 * s / b$$

$$h_c = 2 * s / c$$

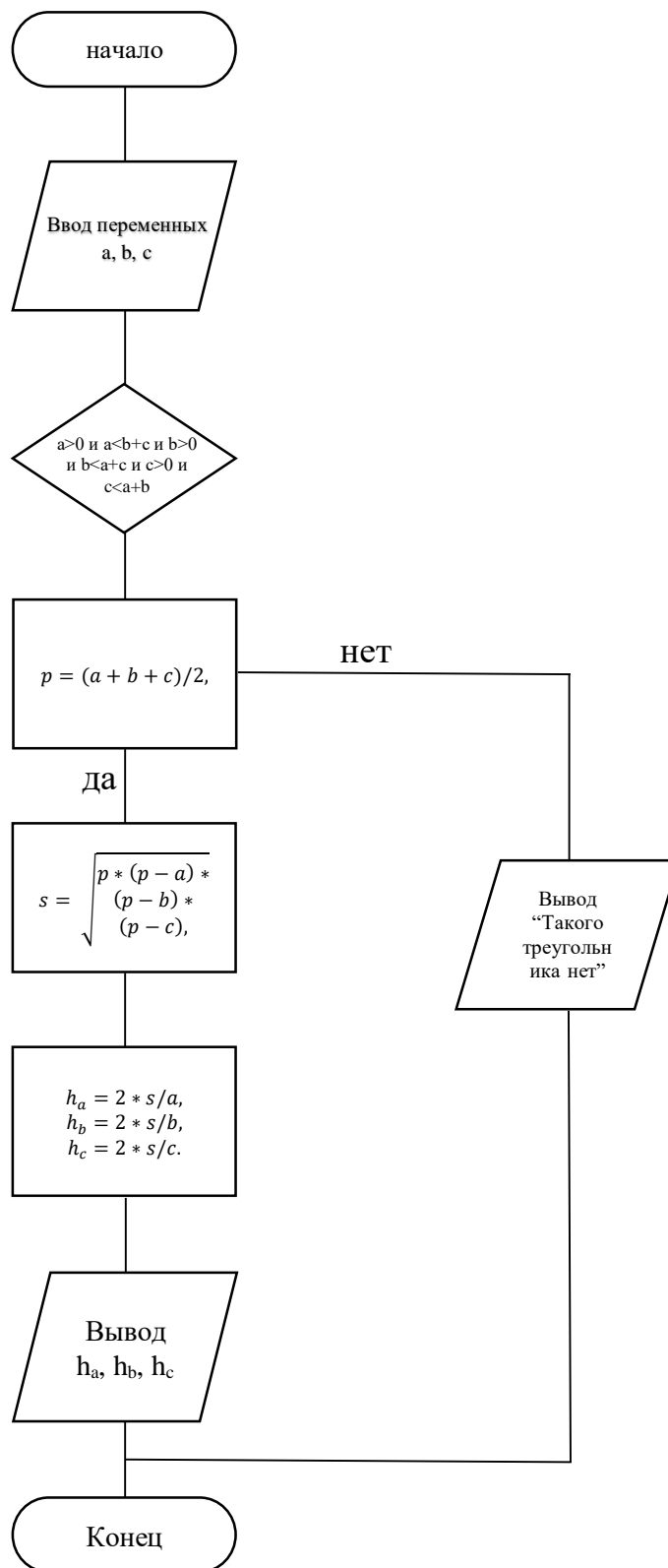
Вывод  $h_a, h_b, h_c$

Иначе

Вывод «Такого треугольника нет»

Конец

## Блок-схема алгоритма



### 3.3.4 Листинг программной реализации

```
#include <stdio.h>
#include <math.h>
#include <conio.h>
#include <clocale>

int main()
{
    setlocale(0, "rus");
    float a, b, c, s, p, ha, hb, hc;
    printf("Введите стороны треугольника\n");
    scanf_s("%f%f%f", &a, &b, &c);
    if (a > 0 && a < b + c && b > 0 && b < a + c && c > 0 && c < a + b)
    {


$p = (a + b + c) / 2;$


        s = sqrt(p * (p - a) * (p - b) * (p - c));
        ha = 2 * (s / a);
        hb = 2 * (s / b);
        hc = 2 * (s / c);
        printf("Высоты треугольника равны %f %f %f", ha, hb, hc);
    }
    else printf("Такого треугольника нет\n");
    _getch();
    return 0;
}
```

### 3.3.5 Результаты

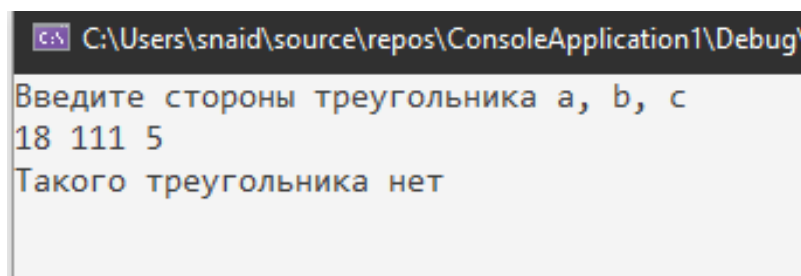


Рисунок 7 - Неверное соотношение сторон.

```

C:\Users\snaid\source\repos\ConsoleApplica
Введите стороны треугольника а, b, с
55 11 -5
Такого треугольника нет

```

Рисунок 8 - Одно значение отрицательное.

```

C:\Users\snaid\source\repos\ConsoleApplication1\Debug\ConsoleApplication1
Введите стороны треугольника а, b, с
3 5 6
Высоты треугольника равны 4,988877 2,993326 2,494438

```

Рисунок 9 - Верные значения.

### 3.3.6 Таблица

№	Переменные			Программный расчет	Ручной расчет
	a	b	c		
1	18	111	5	Такого треугольника нет	Такого треугольника нет
2	55	11	-5	Такого треугольника нет	Такого треугольника нет
3	3	5	6	a = 4.988877 b = 2.993326 c = 2.494438	a = 5 b = 3 c = 2.5

1. Возьмем переменные a=18, b=111, c=5, неверные значения треугольника.
2. Возьмем переменные a=55, b=11, c=-5, неверные значения треугольника.
3. Возьмем переменные a=3, b=5, c=6, результаты программного расчета  
a = 4.988877, b = 2.993326, c = 2.494438, результат ручного расчета a = 5,  
b = 3, c = 2.5

## 4 Задача 4

### 4.1 Постановка задачи

Вычислите площадь поверхности и объем усеченного конуса, используя функцию возведения в квадрат.

$$s = \pi * (R + r) * l + \pi * R^2 + \pi * r^2,$$

$$v = (1/3) * (R^2 + r^2 + R * r) * h * \pi$$

### 4.2 Структурные требования

$$R > r \text{ и } R > 0 \text{ и } r > 0 \text{ и } h > 0$$

### 4.3 Описание работы

#### 4.3.1 Используемые переменные

R – Радиус верхнего основания

r – Радиус нижнего основания

h – Высота

l – образующая

s – площадь

v – объем

#### 4.3.2 Псевдокод программы

Начало

Ввод переменных R, r, h,  $\pi = 3.141592$ , l, s, v

Если

$R > r$  и  $R > 0$  и  $r > 0$  и  $h > 0$

Условие

$$l = \sqrt{h^2 + (R - r)^2}$$

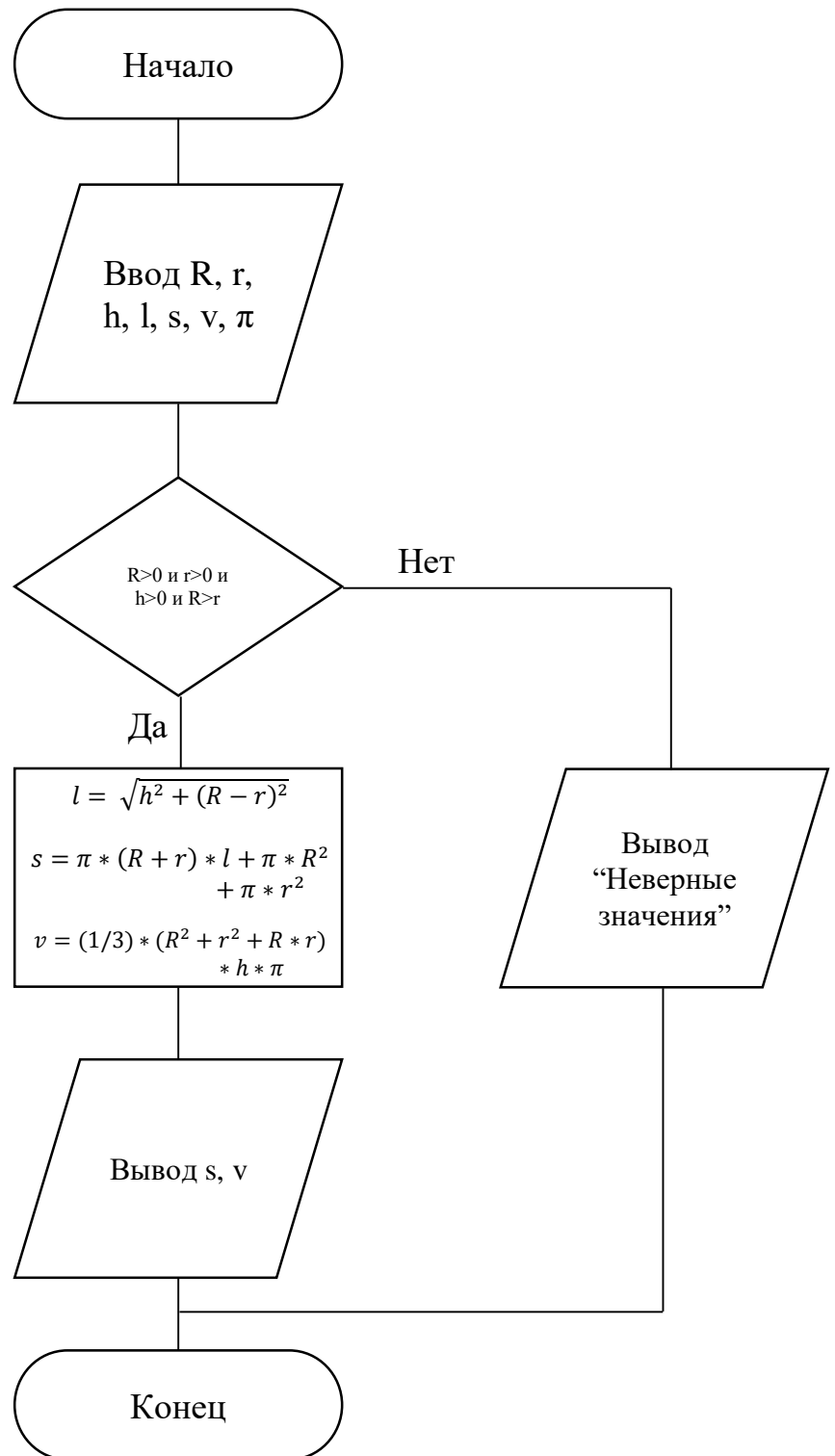
$$s = \pi * (R + r) * l + \pi * R^2 + \pi * r^2,$$

$$v = (1/3) * (R^2 + r^2 + R * r) * h * \pi$$



Вывод  $s$  и  $v$   
 Иначе  
 Вывод (“Неверные параметры”)  
 Вывод  $s$  и  $v$   
 Конец

#### 4.3.3 Блок-схема алгоритма

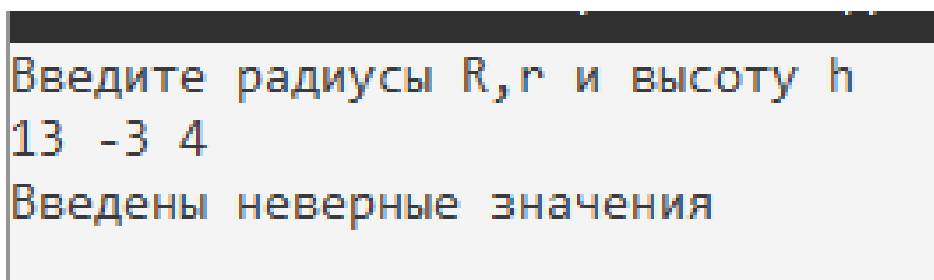


#### 4.3.4 Листинг программной реализации

```
#include <stdio.h>
#include <math.h>
#include <conio.h>
#include <clocale>

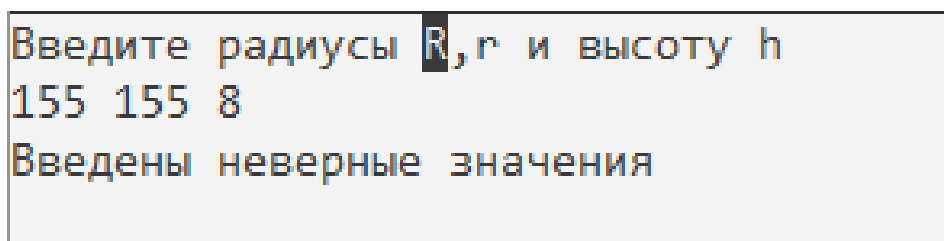
int main()
{
    setlocale(0, "rus");
    float pi = 3.141592, s, R, r, v, h, l;
    printf("Введите радиусы R,r и высоту h\n");
    scanf_s("%f%f%f", &R, &r, &h);
    if (R > r && R > 0 && r > 0 && h > 0)
    {
        l = sqrt(pow(h, 2) + (pow(R, 2) - 2 * R * r + pow(r, 2)));
        s = pi * (R + r) * l + pi * pow(R, 2) + pi * pow(r, 2);
        v = (1.0 / 3) * (pow(R, 2) + pow(r, 2) + R * r) * h * pi;
        printf("s = %f; v = %f", s, v);
    }
    else
        printf("Введены неверные значения ");
    _getch();
    return 0;
}
```

#### 4.3.5 Результат



```
Введите радиусы R,r и высоту h
13 -3 4
Введены неверные значения
```

Рисунок 10 – Неверные значения (Отрицательное значение)



```
Введите радиусы R,r и высоту h
155 155 8
Введены неверные значения
```

Рисунок 11 – Неверные значения (R=r)

Введите радиусы R, r и высоту h  
 16 15 13  
 $s = 2780,907471$ ;  $v = 9815,380859$

Рисунок 12 – Верный результат

#### 4.3.6 Таблица

№	Переменные			Программный расчет	Ручной Расчет
	r	R	h		
1	13	-3	4	Введены неверные значения	Неверные значения
2	155	155	8	Введены неверные значения	Неверные значения
3	16	15	13	$s = 2780,907471$ $v = 9815,380859$	$s = 2781$ $v = 9815$

1. Возьмем переменные  $r = 13$ ,  $R = -3$ ,  $h = 4$ , неверные значения.
2. Возьмем переменные  $r = 155$ ,  $R = 155$ ,  $h = 8$ , неверные значения.
3. Возьмем переменные  $r = 16$ ,  $R = 15$ ,  $h = 13$ , результат программного расчета  $s = 2780,907471$ ,  $v = 9815,380859$ , результат ручного расчета  $s = 2781$ ,  $v = 9815$ .

## 5 Задача 5

### 5.1 Постановка задачи

Вычислите значение функции, используя функцию возведения  $e$  в заданную степень EXP и функцию SIN.

$$y = a * e^{-a*\omega*x} \sin(\omega * x)$$
$$x = (\pi/2 - \varphi)$$

### 5.2 Структурные требования

$a, \omega, \varphi (0;1)$

### 5.3 Описание работы

#### 5.3.1 Используемые переменные

$a$  – параметр.  
 $\omega$  – сферический координат.  
 $\varphi$  – сферический координат.  
 $x$  – декардовый координат.  
 $y$  – декардовый координат.  
 $\pi$  – постоянная величина.

#### 5.3.2 Псевдокод программы

Начало

Ввод переменных  $a, \omega, \varphi, x, y, \pi$

Если  $a, \omega, \varphi (0;1)$

Условие

$$x = \pi/2 - \varphi$$

$$y = a * e^{-a*\omega*x} \sin(\omega * x)$$

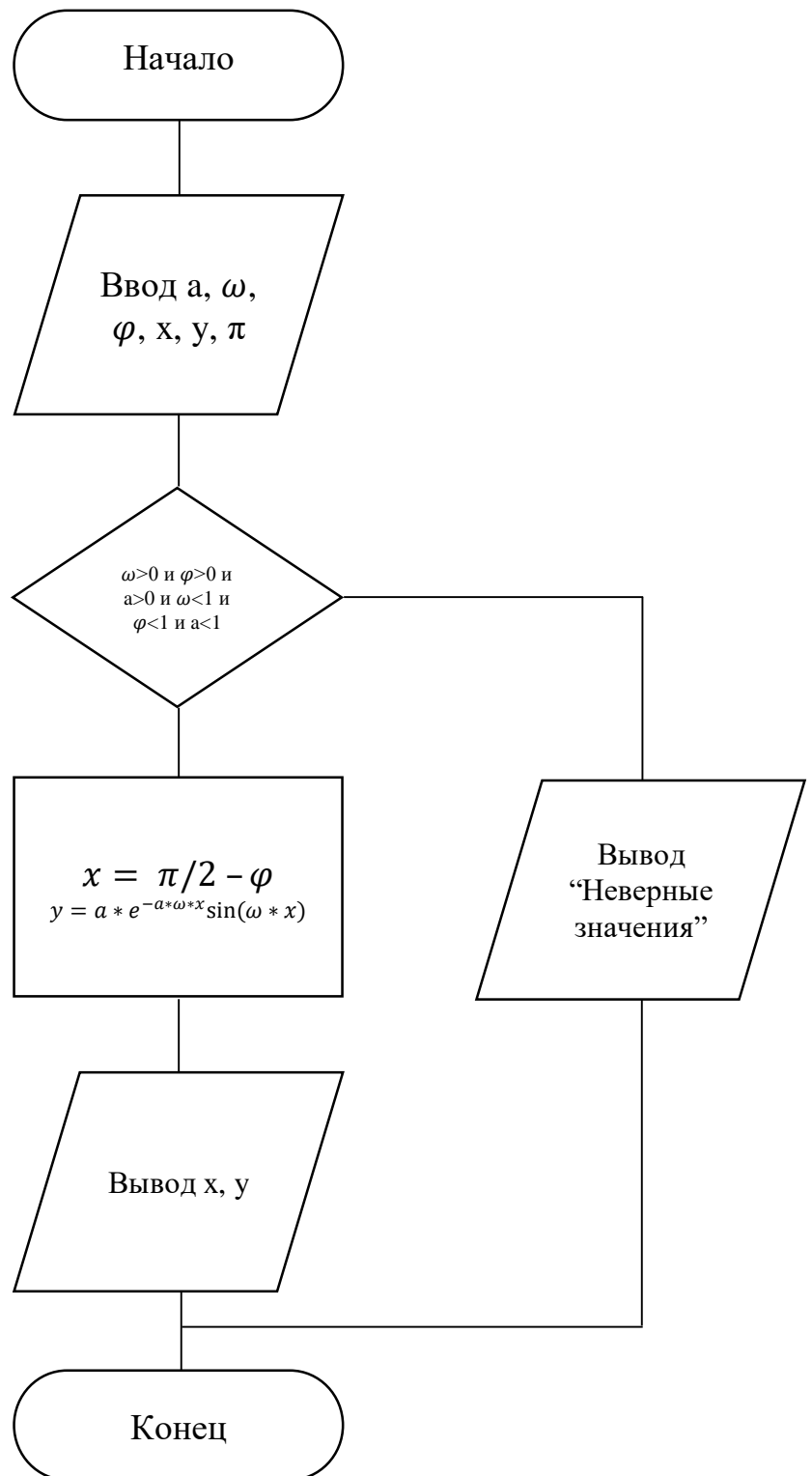
Вывод  $x, y$

Иначе

Вывод “Неверные значения”

Конец

### 5.3.3 Блок-схема алгоритма

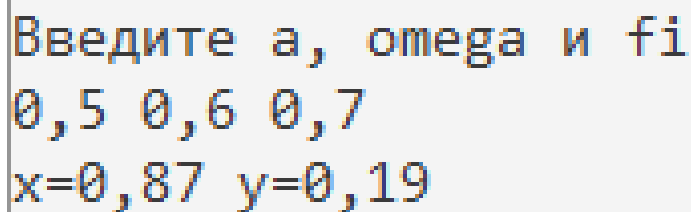


### 5.3.4 Листинг программной реализации

```
#include "stdafx.h"
#include "conio.h"
#include "math.h"
#include "clocale"

int main()
{
    setlocale(0, "rus");
    float a, om, fi, Pi = 3.141592, x, y;
    printf("Введите a, omega и fi\n");
    scanf_s("%f%f%f", &a, &om, &fi);
    if ((om > 0) && (fi > 0) && (a > 0) && (om < 1) && (fi < 1) && (a < 1))
    {
        x = Pi / 2 - fi;
        y = a*exp(-a*om*x)*sin(om*x);
        printf("x=%.2f y=%.2f", x, y);
    }
    else printf("Неверные значения");
    _getch();
    return 0;
}
```

### 5.3.5 Результат



```
Введите a, omega и fi
0,5 0,6 0,7
x=0,87 y=0,19
```

Рисунок 13 – Верные значения

```

Введите а, omega и fi
0,5 0,7 0,4
x=1,17 y=0,24

```

Рисунок 14 – Верные значения

```

Введите а, omega и fi
-0,8 0,5 0,6
Неверные значения

```

Рисунок 15 – Введенные значения неверны

### 5.3.6 Таблица

№	Переменные			Программный расчет	Ручной Расчет
	а	$\omega$	$\varphi$		
1	0,5	0,6	0,7	x=0,87 y=0,24	x=0,8 y=0,2
2	0,5	0,7	0,4	x=1,17 y=0,24	x=1,2 y=0,24
3	-0,8	0,5	0,6	Неверные значения	Неверные значения

1. Возьмем переменные  $a=0,5$ ,  $\omega=0,6$ ,  $\varphi=0,7$ , программный расчет  $x=0,87$ ;  $y=0,24$
2. Возьмем переменные  $a=0,5$ ,  $\omega=0,7$ ,  $\varphi=0,4$ , программный расчет  $x=1,17$ ;  $y=0,24$
3. Возьмем переменные  $a=-0,8$ ,  $\omega=0,5$ ,  $\varphi=0,6$ , программный расчет неверные значения.

## 6    Задача 6

### 6.1 Постановка задачи

Выведите третий справа разряд целого числа, содержащего не менее трёх разрядов.

### 6.2 Структурные требования

Число должно быть целое и не менее трех разрядов.

### 6.3    Описание работы

#### 6.3.1 Используемые переменные

а – переменная, которая вводится пользователем

sot – окончательное значение

#### 6.3.2 Псевдокод программы

Начало

    Ввод а, sot

    Если

        а<-99 или а>99

    То

        sot = abs(а % 1000) / 100

    Вывод sot

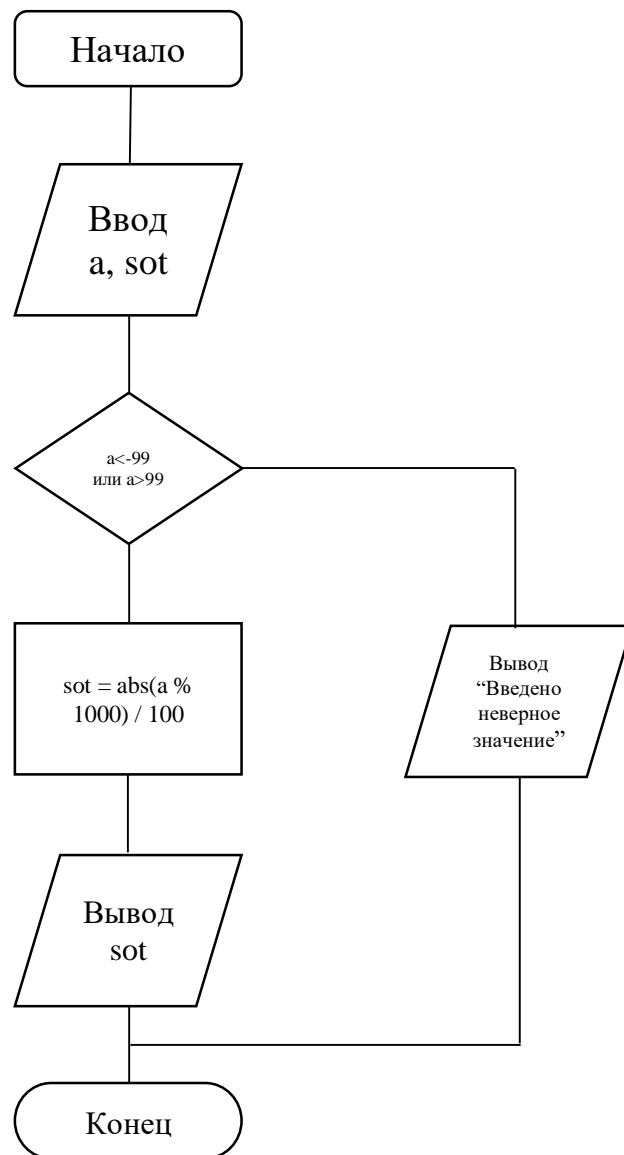
    Иначе

        Вывод “Введено неверное значение”

Конец



### 6.3.3 Блок-схема алгоритма



### 6.3.4 Листинг программной реализации

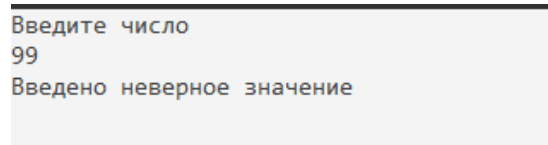
```
#include <stdio.h>
#include <math.h>
#include <conio.h>
#include <locale.h>
int main()
{
    setlocale(0, "rus");
    int a, sot;
    printf("Введите число\n");
    scanf_s("%d", &a);
    if ((a < -99) || (a > 99))
    {
```

```

        sot = abs((a % 1000) / 100);
        printf("%d", sot);
    }
    else printf("Введено неверное значение");
    _getch();
    return 0;
}

```

### 6.3.5 Результат

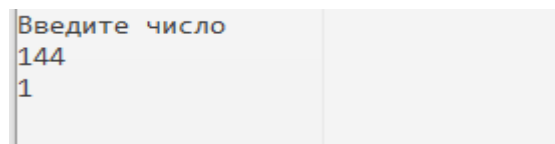


```

Введите число
99
Введено неверное значение

```

Рисунок 16 – Неверное значение

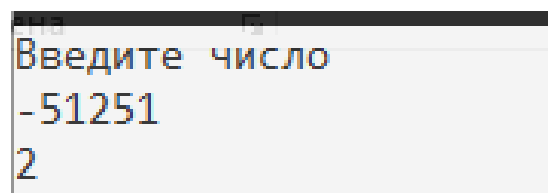


```

Введите число
144
1

```

Рисунок 17 – Верное значение (Положительное)



```

Введите число
-51251
2

```

Рисунок 18 – Верное значение (Отрицательное)

### 6.3.6 Таблица

№	Переменные	Программный расчет	Ручной Расчет
	a		
1	99	Неверное значение	Неверное значение
2	144	1	1
3	-51251	2	2

1. Возьмем переменную a=99, неверное значение
2. Возьмем переменную a=144, программный расчет 1, ручной расчет 1
3. Возьмем переменную a=-51251, программный расчет 2, ручной расчет 2

## 7    Задача 7

### 7.1 Постановка задачи

Выведите первый после десятичной точки разряд дробной части действительного числа, содержащего не менее одного разряда в дробной части.

### 7.2 Структурные требования

$a > 0$  и  $b > 0$

### 7.3    Описание работы

#### 7.3.1 Используемые переменные

a – числитель

b – знаменатель

sot – первое значение

sot2 – результат

#### 7.3.2 Псевдокод алгоритма

Начало

    Ввод a, b, sot, sot2

    Если

$a > 0$  и  $b > 0$

    То

$sot = (((a * 1.) / (b * 1.) * 10))$

$sot2 = sot \% 10$

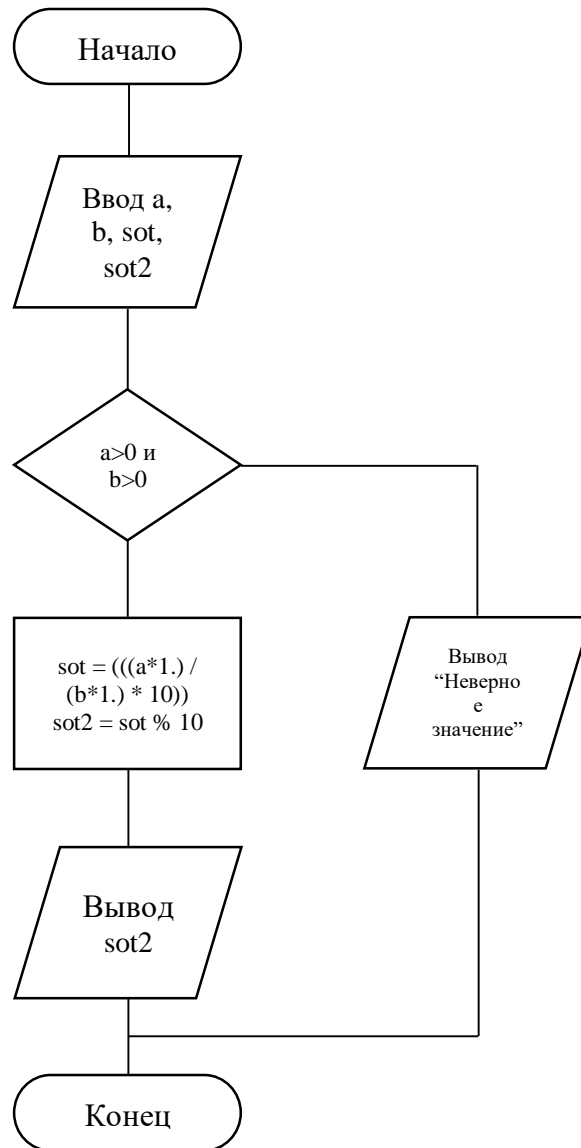
    Вывод sot2

    Иначе

        Вывод “Неверное значение”

Конец

#### 7.3.3 Блок-схема алгоритма



#### 7.3.4 Листинг программной реализации

```

#include <stdio.h>
#include <math.h>
#include <conio.h>
#include <clocale>

int main()
{
    setlocale(0, "rus");
    int a, b, sot;
    float sot2;
    printf("Введите числитель и знаменатель\n");
    scanf_s("%d%d", &a, &b);
    if ((a > 0) && (b > 0))

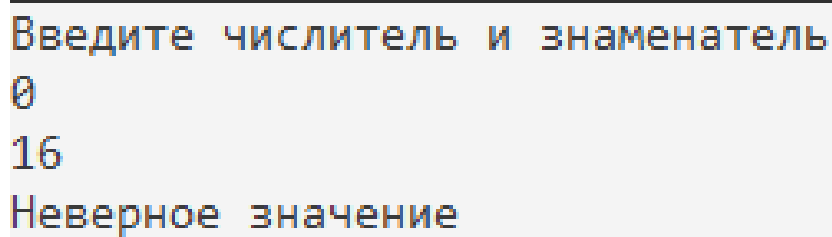
```

```

{
    sot = (((a*1.) / (b*1.) * 10));
    sot2 = sot % 10;
    printf("%.0f", sos2);
}
else
printf("Неверное значение");
_getch();
return 0;
}

```

### 7.3.5 Результат

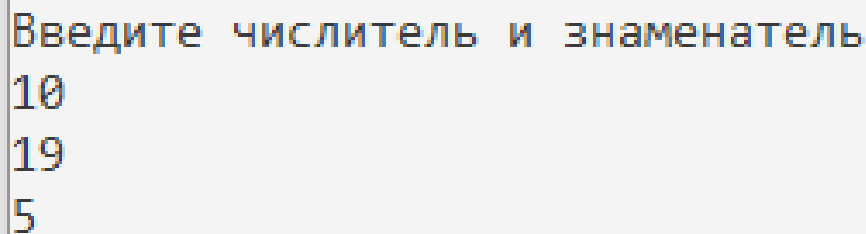


```

Введите числитель и знаменатель
0
16
Неверное значение

```

Рисунок 19 – Неверное значение

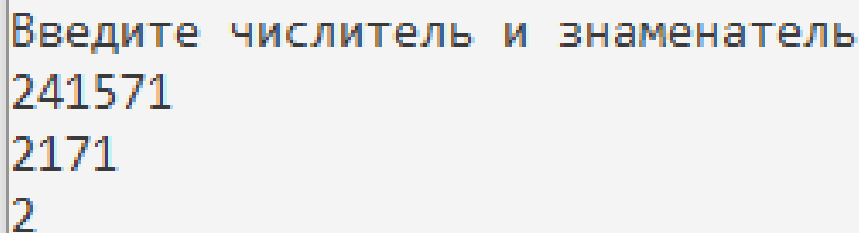


```

Введите числитель и знаменатель
10
19
5

```

Рисунок 20 – Верные значения



```

Введите числитель и знаменатель
241571
2171
2

```

Рисунок 21 – Верные значения

### 7.3.6 Таблица

№	Переменные		Программный расчет	Ручной Расчет
	a	b		
1	0	16	Неверные значения	Неверные значения
2	10	19	5	5
3	241571	2171	2	2

1. Возьмем переменные  $a=0$ ,  $b=16$ , значения неверные.
2. Возьмем переменные  $a=10$ ,  $b=19$ , результат программного расчета 5, результат ручного расчета 5.
3. Возьмем переменные  $a=241571$ ,  $b=2171$ , результат программного расчета 2, результат ручного расчета 2.