



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО  
ОБРАЗОВАНИЯ РФ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ»

Факультет управления и информатики в технологических системах  
Кафедра информационной безопасности  
Направление подготовки (специальность) 10.05.03 Информационная  
безопасность автоматизированных систем

**Отчет**  
по практической работе № 1  
по дисциплине Алгоритмы и структуры данных

Выполнил студент гр. УБ-31

Кувшинова Анастасия Алексеевна  
(ФИО)

237162  
(номер зачетки)

\_\_\_\_\_  
(подпись)

Проверил:

Доцент каф ИБ Зиновьева В.В.  
(должность руководителя от университета) (ф.и.о)

\_\_\_\_\_  
(оценка)

\_\_\_\_\_  
(подпись)

. .2024  
(дата)

## 1 ЦЕЛИ

Целями выполнения практической работы №1 Алгоритм следования являются:

- 1 Изучить ГОСТ 19.701-90;
- 2 Разработать графический алгоритм;
- 3 Разработать программу вычисления на языке C++.

## 2 ЗАДАНИЕ №1. ВАРИАНТ 1

### 2.1 Формулировка задания

Напишите алгоритм и программу для расчета  $Z$

$$Z = \frac{b + \sqrt{b^2 - 4ac}}{2a + 5b}$$

### 2.2 Таблица

Построим таблицу с различными значениями переменных  $a$ ,  $b$ ,  $c$  так, чтобы они давали разный результат при выводе

	$a$	$b$	$c$	$Z$
<b>1 вариант</b>	1	2	3	Корень отрицательный
<b>2 вариант</b>	0	0	1	Знаменатель. равен 0
<b>3 вариант</b>	1	20	2	0.39

В первом варианте значений переменных мы получим отрицательный корень, так как:

$$Z = \frac{b + \sqrt{b^2 - 4ac}}{2a + 5b} = \frac{2 + \sqrt{2^2 - 4 * 1 * 3}}{2 * 1 + 5 * 2} = \frac{2 + \sqrt{4 - 12}}{12} = \text{отрицат. корень}$$

Во втором варианте значений переменных мы получим знаменатель, который равен 0, так как:

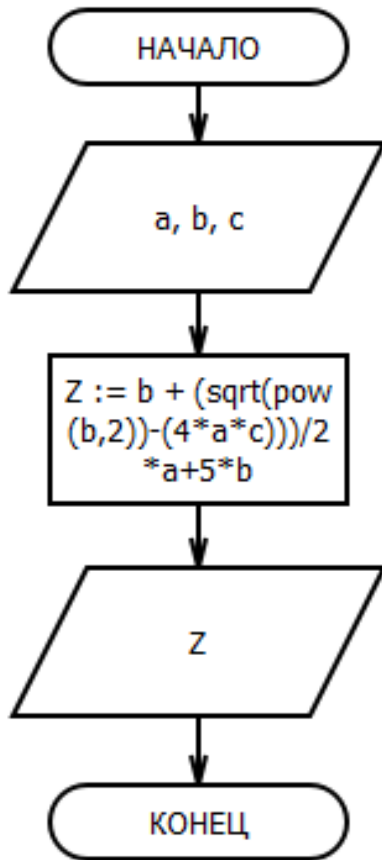
$$Z = \frac{0 + \sqrt{0^2 - 4 * 0 * 1}}{2 * 0 + 5 * 0} = \frac{0}{0} = \text{знам. равен 0}$$

В третьем варианте значений переменных мы получим ответ, так как корень положительный и знаменатель не равен нулю:

$$Z = \frac{20 + \sqrt{20^2 - 4 * 1 * 2}}{2 * 1 + 5 * 20} = \frac{20 + \sqrt{400 - 8}}{2 + 100} = \frac{40}{102} = 0.39$$

### 2.3 Код 1. Без цикла

Блок-схема:



Код:

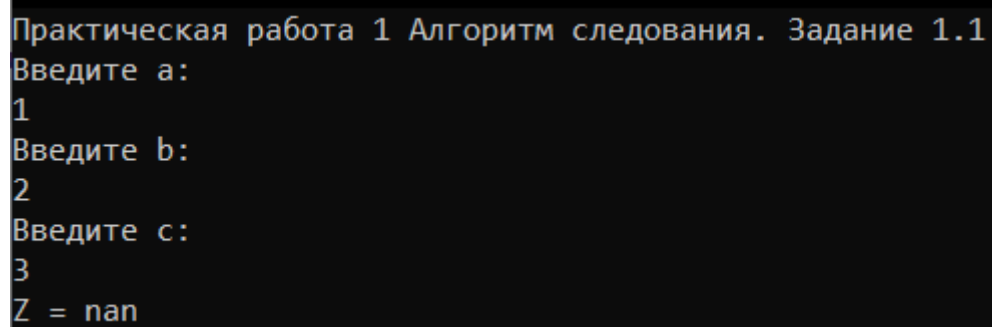
```
#include <iostream>
#include <locale>
#include <cmath>
using namespace std;

int main()
{
    setlocale(LC_ALL, "Russian");
    cout << "Практическая работа 1 Алгоритм следования. Задание 1.1 " <<
endl;
    float a, b, c;
    cout << "Введите a: " << endl;
    cin >> a;
    cout << "Введите b: " << endl;
    cin >> b;
    cout << "Введите c: " << endl;
    cin >> c;
    double Z;
    Z = (b + (sqrt(pow(b,2)-(4*a*c))))/(2*a+5*b);
    cout << "Z = " << Z << endl;
```

```
    return 0;  
}
```

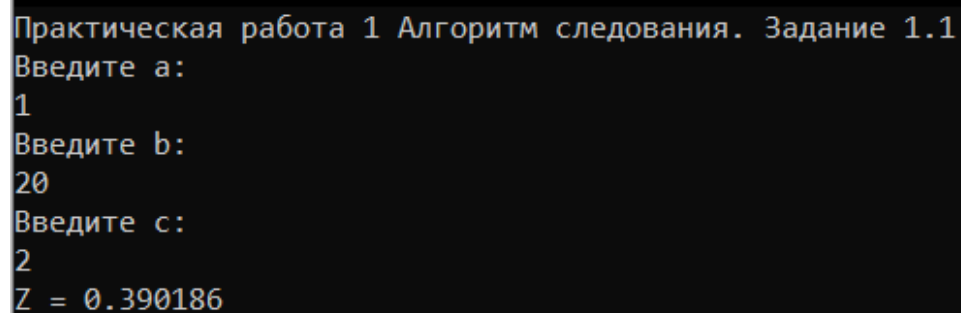
### 2.3.1 Скрины работы программы

Если ввести значения, подставив которые мы получим отрицательный корень или ноль в знаменателе, то результат выйдет следующий:



```
Практическая работа 1 Алгоритм следования. Задание 1.1  
Введите a:  
1  
Введите b:  
2  
Введите c:  
3  
Z = nan
```

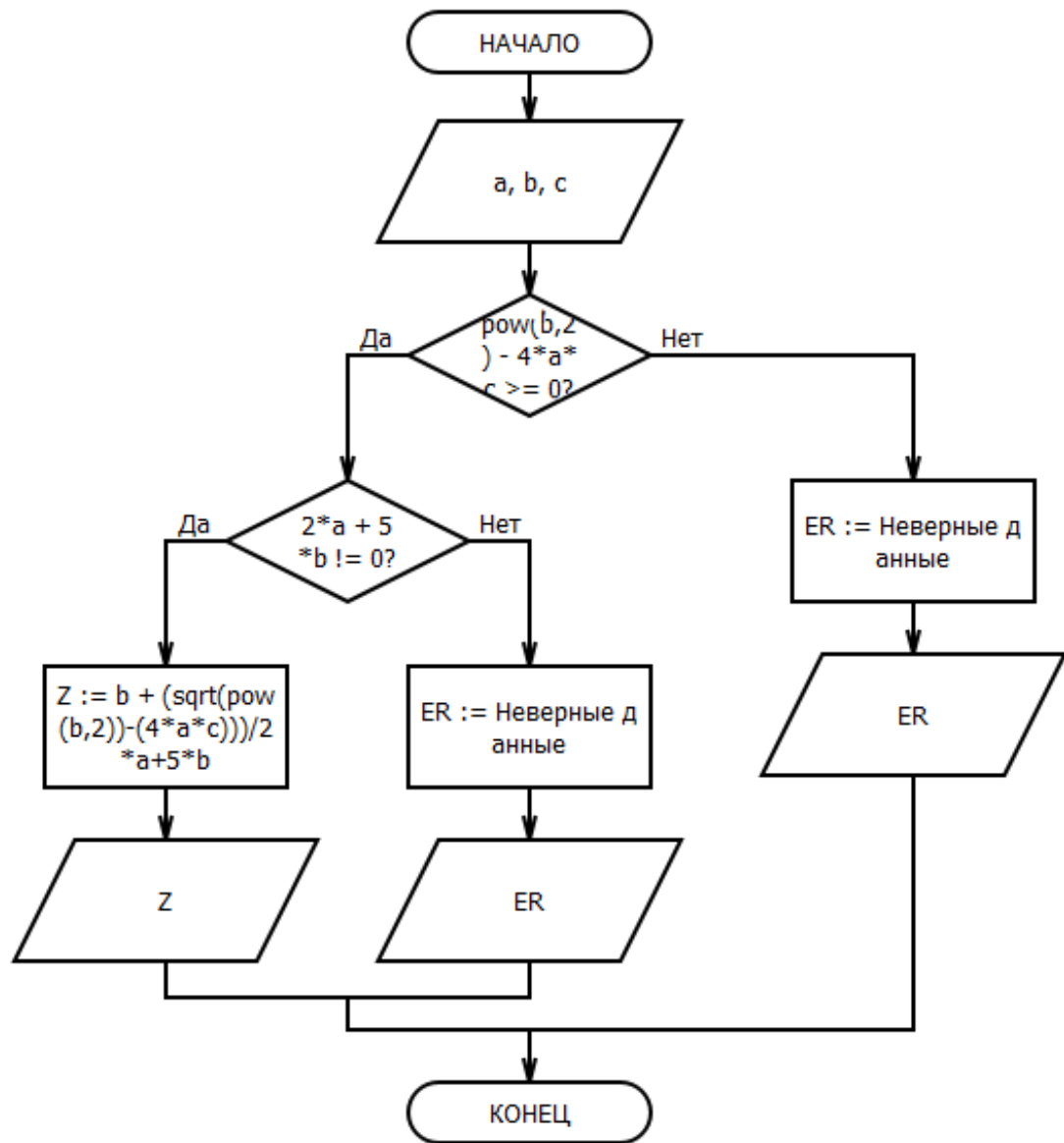
Иначе программа выведет верное значение Z:



```
Практическая работа 1 Алгоритм следования. Задание 1.1  
Введите a:  
1  
Введите b:  
20  
Введите c:  
2  
Z = 0.390186
```

### 2.4 Код 2. Цикл с предусловием

Блок-схема:



Код:

```
#include <iostream>
#include <locale>
#include <cmath>
using namespace std;
```

```
int main()
{
    setlocale(LC_ALL, "Russian");
    cout << "Практическая работа 1 Алгоритм следования. Задание 1.2 " <<
endl;
    float a, b, c;
    cout << "Введите a: " << endl;
    cin >> a;
    cout << "Введите b: " << endl;
```

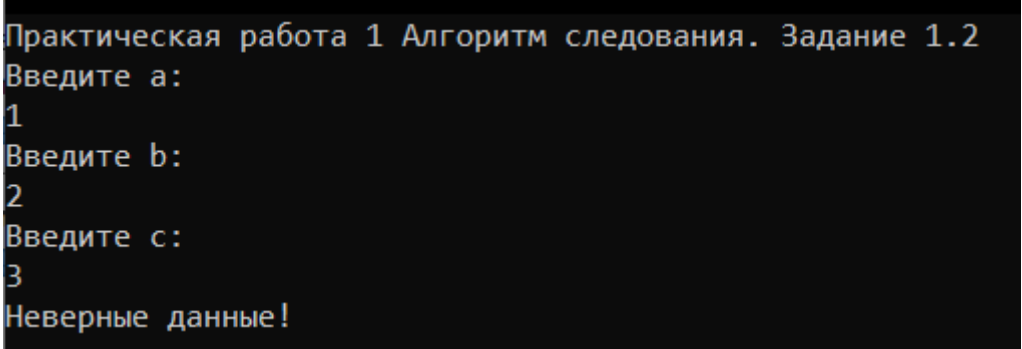
```

cin >> b;
cout << "Введите c: " << endl;
cin >> c;
double Z;
if ((pow(b,2)-4*a*c)>=0 && 2*a+5*b!=0) {
    Z = (b + (sqrt(pow(b,2)-(4*a*c))))/(2*a+5*b);
    cout << "Z = " << Z << endl;
} else {
    cout << "Неверные данные!" << endl;
}
return 0;
}

```

#### 2.4.1 Скрины работы программы

Введем значения, чтобы корень получился отрицательным (либо знаменатель равен 0) и увидим:

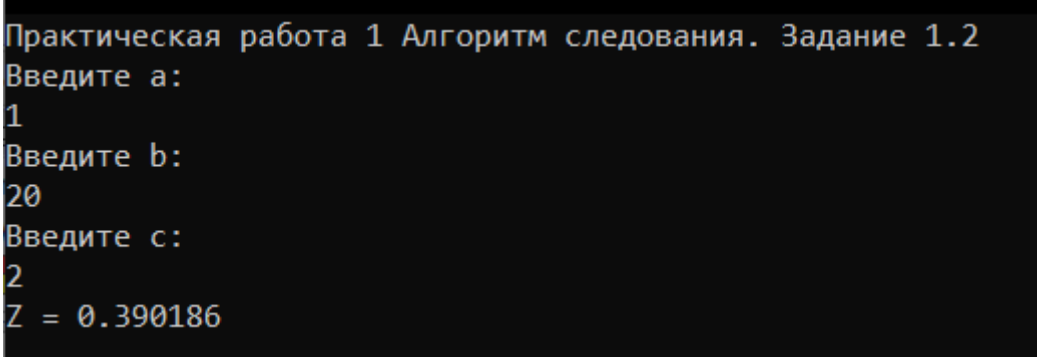


```

Практическая работа 1 Алгоритм следования. Задание 1.2
Введите a:
1
Введите b:
2
Введите c:
3
Неверные данные!

```

Введем корректные данные и получим ответ:



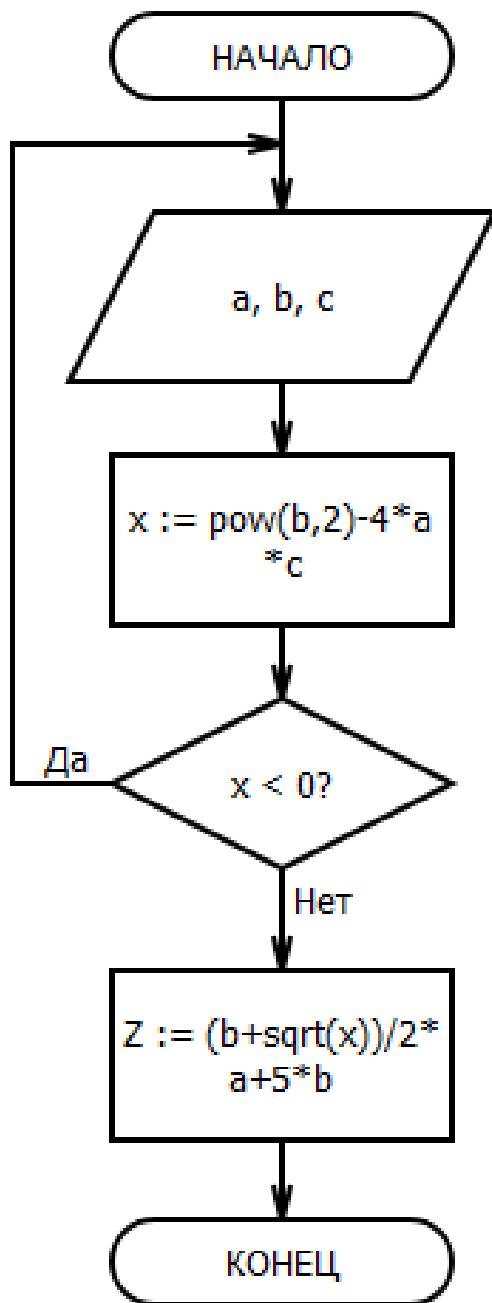
```

Практическая работа 1 Алгоритм следования. Задание 1.2
Введите a:
1
Введите b:
20
Введите c:
2
Z = 0.390186

```

#### 2.5 Код 3. Цикл с постусловием

Блок-схема:



Код:

```

#include <iostream>
#include <locale>
#include <cmath>
using namespace std;

int main()
{
    setlocale(LC_ALL,"Russian");
    cout << "Практическая работа 1 Алгоритм следования. Задание 1.3 " <<
endl;
    float a, b, c, x1, x2;
  
```



```

double Z;
do {
    cout << "Введите a: " << endl;
    cin >> a;
    cout << "Введите b: " << endl;
    cin >> b;
    cout << "Введите c: " << endl;
    cin >> c;
    x1 = pow(b,2) - 4*a*c;
    x2 = 2*a+5*b;

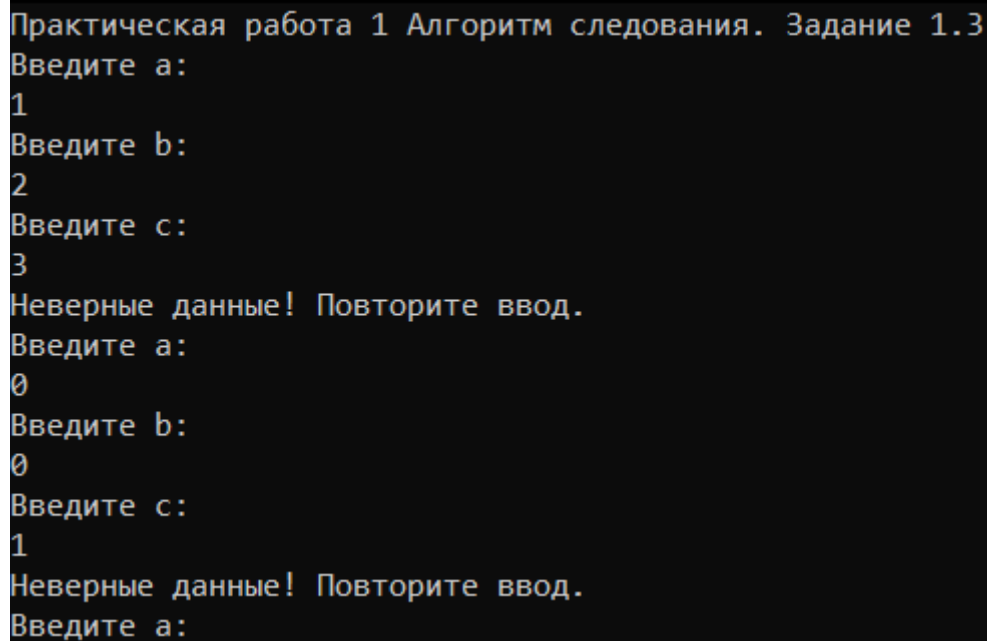
    if (x1<0 || x2==0) {
        cout << "Неверные данные! Повторите ввод." << endl;
    }
} while (x1<0 || x2==0);
Z = (b + (sqrt(pow(b,2)-(4*a*c))))/(2*a+5*b);
cout << "Z = " << Z << endl;

return 0;
}

```

### 2.5.1 Скрины работы программы

Снова введем такие значения, которые дадут отрицательный корень или ноль в знаменателе:



```

Практическая работа 1 Алгоритм следования. Задание 1.3
Введите a:
1
Введите b:
2
Введите c:
3
Неверные данные! Повторите ввод.
Введите a:
0
Введите b:
0
Введите c:
1
Неверные данные! Повторите ввод.
Введите a:

```

Мы можем наблюдать, что программа не дает нам выйти из цикла, пока мы не введем корректные значения. Попробуем ввести такие числа, которые не будут давать отрицательный корень либо 0 в знаменателе:

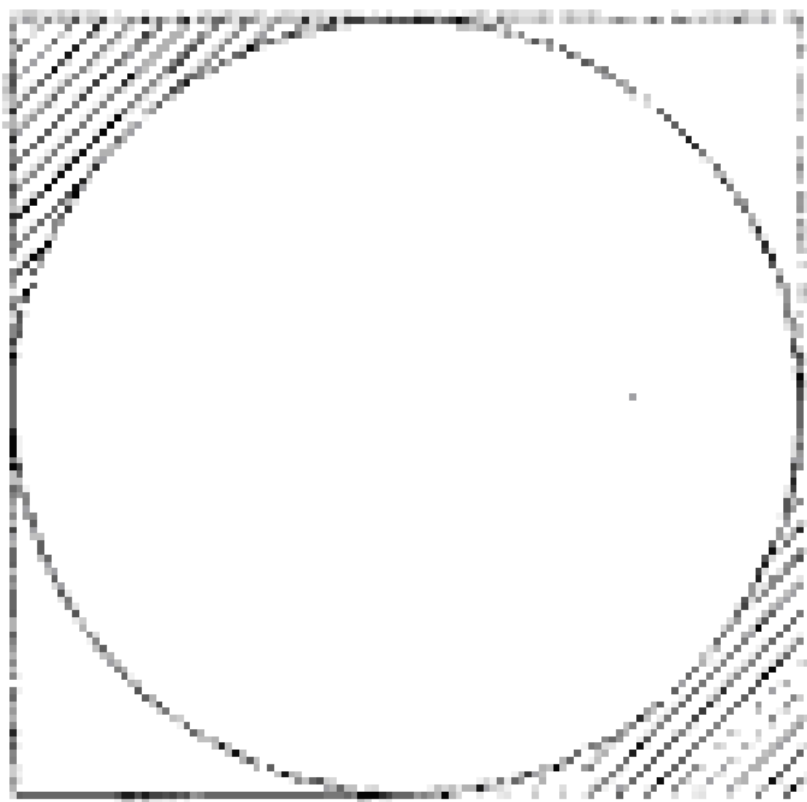
```
Введите a:  
1  
Введите b:  
20  
Введите c:  
2  
Z = 0.390186
```

Таким образом, можно сделать вывод, что программа работает корректно.

### 3 ЗАДАНИЕ №2. ВАРИАНТ 7

#### 3.1 Формулировка задания

В квадрат вписана окружность. Определите площадь заштрихованной части фигуры, если известен радиус окружности.



#### 3.2 Таблица

Для того, чтобы определить площадь заштрихованной части фигуры, нам нужно:

1. Определить, чему равна площадь квадрата
2. Определить, чему равна площадь круга
3. Вычесть площадь круга из площади квадрата
4. Разделить полученную разность на 2 (тк нам нужны лишь два “треугольника” в углах квадрата, а не четыре)

Построим таблицу с различными значениями переменных  $a$ ,  $b$ ,  $c$  так, чтобы они давали разный результат при выводе

	$a$	$S_{\text{кв}}$	$S_{\text{окр}}$	$S_{\text{ф}}$
<b>1 вариант</b>	2	4	3.14	0.43
<b>2 вариант</b>	5	25	19.625	2.6875

Рассмотрим вариант 1. Ввели значение  $a = 2$ .

Зная значение  $a$ , мы можем узнать площадь квадрата:

$$S_{\text{кв}} = a^2 = 2^2 = 4$$

Зная, что:

$$R = \frac{a}{2}$$

Преобразуем формулу площади окружности:

$$S_{\text{окр}} = \pi * R^2 = \frac{\pi * a^2}{4} = \frac{3.14 * 2^2}{4} = 3.14$$

Для того, чтобы узнать площадь заштрихованной части фигуры, необходимо вычесть  $S_{\text{окр}}$  из  $S_{\text{кв}}$  и взять половину от полученного результата:

$$S_{\phi} = \frac{S_{\text{кв}} - S_{\text{окр}}}{2} = \frac{4 - 3.14}{2} = \frac{0.86}{2} = 0.43$$

Менее подробно рассмотрим вариант 2. Ввели значение  $a = 5$ .

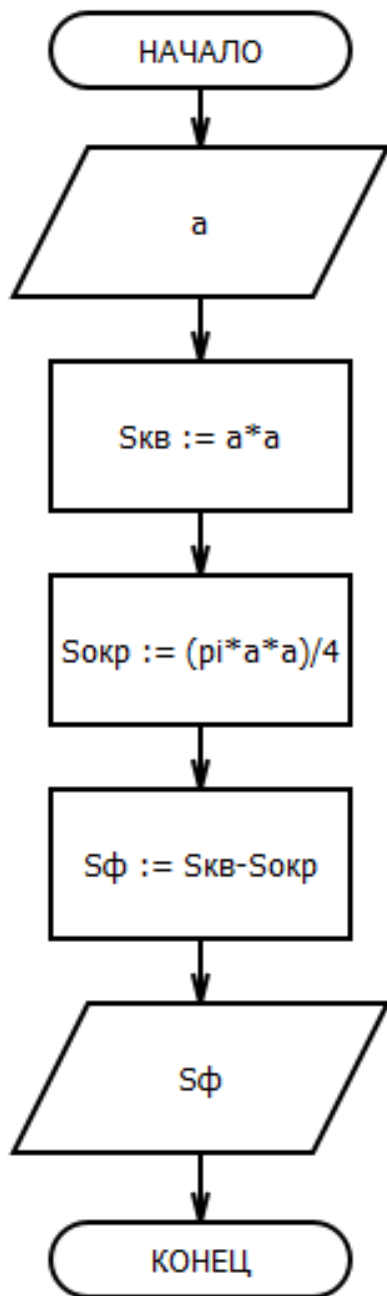
$$S_{\text{кв}} = a^2 = 25$$

$$S_{\text{окр}} = \frac{\pi * a^2}{4} = 19.625$$

$$S_{\phi} = \frac{25 - 19.625}{2} = 2.6875$$

### 3.3 Код 1. Без цикла

Блок-схема:



Код:

```
#include <iostream>
#include<locale>
#include<cmath>
using namespace std;
```

```
int main()
{
    setlocale(LC_ALL, "Russian");
    cout << "Практическая работа 1 Алгоритм следования. Задание
2.1 " << endl;
```

```

double const pi = 3.14;
float a;
cout << "Введите сторону квадрата: " << endl;
cin >> a;
double S_f, S_okr, S_kv;
S_kv = a*a;
S_okr = (pi * a * a)/4;
S_f = (S_kv - S_okr)/2;

cout << "Площадь закрашенной фигуры: " << S_f << endl;
return 0;
}

```

### 3.3.1 Скрины работы программы

Если ввести отрицательное значение a, то результат выйдет следующий:

```

Практическая работа 1 Алгоритм следования. Задание 2.1
Введите сторону квадрата:
-2
Площадь закрашенной фигуры: 0.43

```

Как можно заметить, программа не обрабатывает ошибку и выводит результат, который получается, если бы a было положительным.

Можем убедиться в этом:

```

Практическая работа 1 Алгоритм следования. Задание 2.1
Введите сторону квадрата:
2
Площадь закрашенной фигуры: 0.43

```

Введем a=5

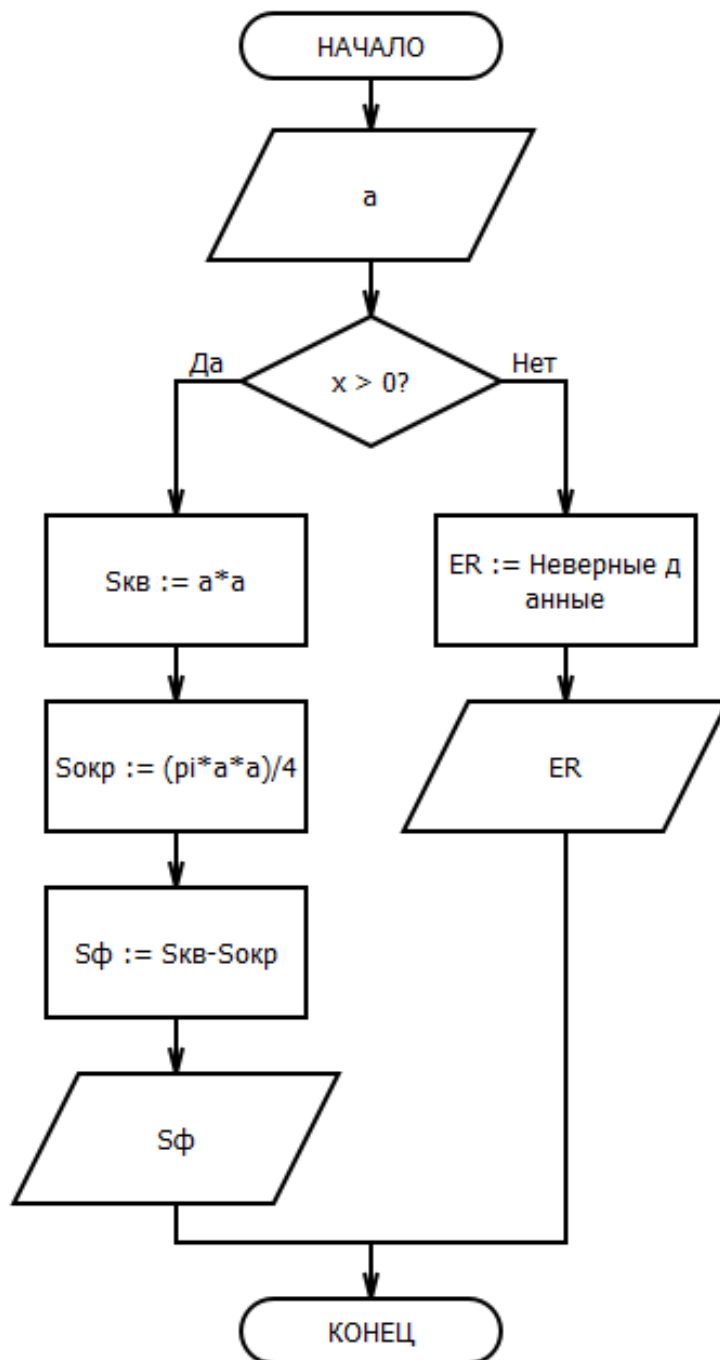
```

Практическая работа 1 Алгоритм следования. Задание 2.3
Введите сторону квадрата:
5
Площадь закрашенной фигуры: 2.6875

```

### 3.4 Код 2. Цикл с предусловием

Блок-схема:



Код:

```
#include <iostream>
#include<locale>
#include<cmath>
using namespace std;
```

```

int main()
{
    setlocale(LC_ALL, "Russian");
    cout << "Практическая работа 1 Алгоритм следования. Задание 2.2 " <<
endl;
    double const pi = 3.14;
    float a;
    cout << "Введите сторону квадрата: " << endl;
    cin >> a;
    double S_f, S_okr, S_kv;
    S_kv = a*a;
    S_okr = (pi * a * a)/4;
    S_f = (S_kv - S_okr)/2;

    if (a>0) {
        cout << "Площадь закрашенной фигуры: " << S_f << endl;
    } else {
        cout << "Неверные данные!" << endl;
    }

    return 0;
}

```

### 3.4.1 Скрины работы программы

Снова введем отрицательное  $a$  и увидим, что программа выводит сообщение об ошибке:

```

Практическая работа 1 Алгоритм следования. Задание 2.2
Введите сторону квадрата:
-2
Неверные данные!

```

Введем корректные данные и получим ответ:

```

Практическая работа 1 Алгоритм следования. Задание 2.2
Введите сторону квадрата:
2
Площадь закрашенной фигуры: 0.43

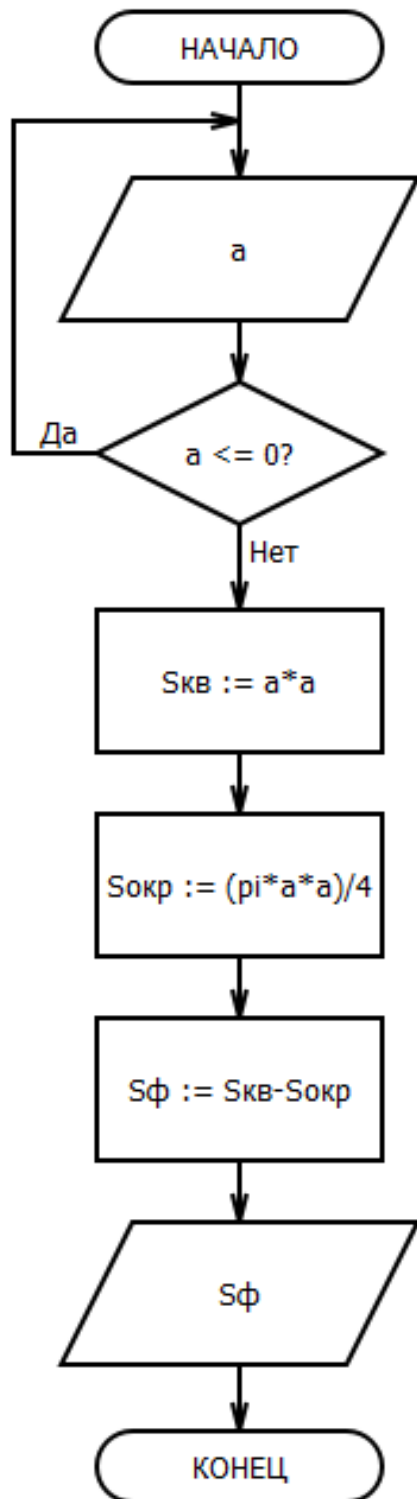
```



Практическая работа 1 Алгоритм следования. Задание 2.2  
Введите сторону квадрата:  
5  
Площадь закрашенной фигуры: 2.6875

### 3.5 Код 3. Цикл с постусловием

Блок-схема:



Код:

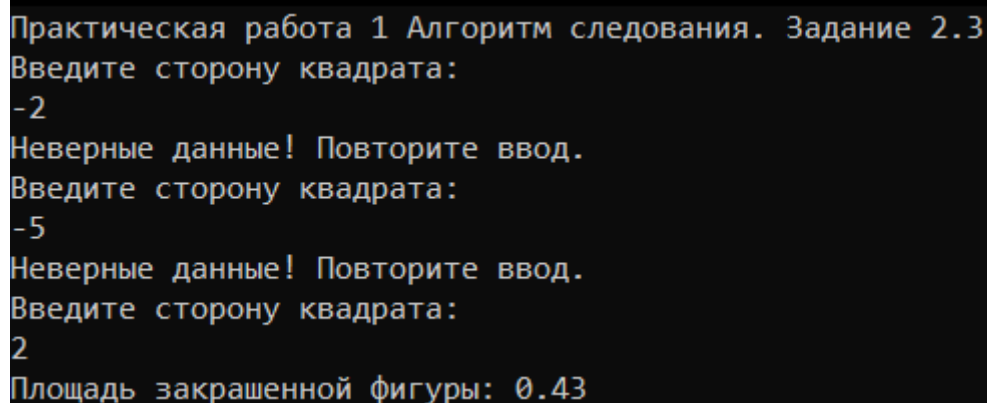
```
#include <iostream>
#include <locale>
#include <cmath>
using namespace std;

int main()
{   setlocale(LC_ALL, "Russian");
    cout << "Практическая работа 1 Алгоритм следования. Задание 2.3 " <<
endl;
    double const pi = 3.14;
    float a, x;
    double S_f, S_okr, S_kv;
    do {
        cout << "Введите сторону квадрата: " << endl;
        cin >> a;

        if (a<0) {
            cout << "Неверные данные! Повторите ввод." << endl;
        }
    } while (a<0);
    S_kv = a*a;
    S_okr = (pi * a * a)/4;
    S_f = (S_kv - S_okr)/2;
    cout << "Площадь закрашенной фигуры: " << S_f << endl;
    return 0;
}
```

### 3.5.1 Скрины работы программы

Мы можем наблюдать, что программа вновь и вновь будет просить ввести значение стороны квадрата, пока мы не введем положительное а:



```
Практическая работа 1 Алгоритм следования. Задание 2.3
Введите сторону квадрата:
-2
Неверные данные! Повторите ввод.
Введите сторону квадрата:
-5
Неверные данные! Повторите ввод.
Введите сторону квадрата:
2
Площадь закрашенной фигуры: 0.43
```

```
Практическая работа 1 Алгоритм следования. Задание 2.3
Введите сторону квадрата:
-65
Неверные данные! Повторите ввод.
Введите сторону квадрата:
0
Неверные данные! Повторите ввод.
Введите сторону квадрата:
5
Площадь закрашенной фигуры: 2.6875
```

Таким образом, можно сделать вывод, что программа работает корректно.