**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-41

*Крылов НикитаРоманович*

*(ФИО)*

*237162*

*(номер зачетки)*

*(подпись)*

Проверил:

Доцент каф ИБ Зиновьева В.В.

*(должность руководителя от университета) (ф.и.о)*

*(оценка) (подпись)*

*. .2025*

*(дата)*

Воронеж – 2024

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 С++.
4. ЗАДАНИЕ №1. ВАРИАНТ 1
   1. Формулировка задания

Напишите алгоритм и программу для расчета Z

* 1. Таблица

𝑍=

)

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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | ***a*** |  |  | ***Z*** |
| **1 вариант** | 1 |  |  | Корень  отрицательный |
| **2 вариант** | 0 |  |  | Знаменатель.  равен 0 |
| **3 вариант** | 1 |  |  | 0.39 |

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

𝑍 =

𝑏 + √𝑏2 − 4𝑎𝑐 2𝑎 + 5𝑏

2 + √22 − 4 ∗ 1 ∗ 3

=

2 ∗ 1 + 5 ∗ 2

2 + √4 − 12

=

12

= отрицат. корень

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

𝑍 =

0 + √02 − 4 ∗ 0 ∗ 1 0

= 

2 ∗ 0 + 5 ∗ 0 0

= знам. равен 0

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

𝑍 =

20 + √202 − 4 ∗ 1 ∗ 2

2 ∗ 1 + 5 ∗ 20

20 + √400 − 8

=

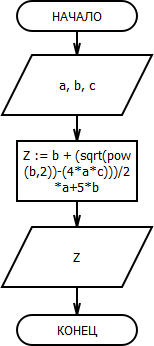
2 + 100

40

=

102

= 0.39

* 1. Код 1. Без цикла Блок-схема:

Код:

#include <iostream> #include <clocale> #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;

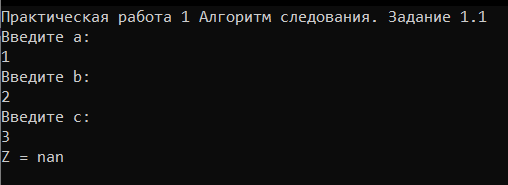
Z = (b + (sqrt(pow(b,2)-(4\*a\*c))))/(2\*a+5\*b); cout << "Z = " << Z << endl;

return 0;

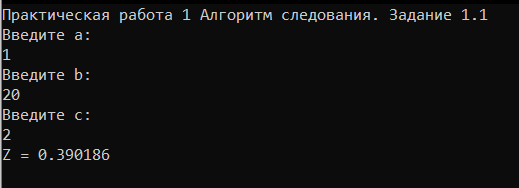
}

* + 1. Скрины работы программы

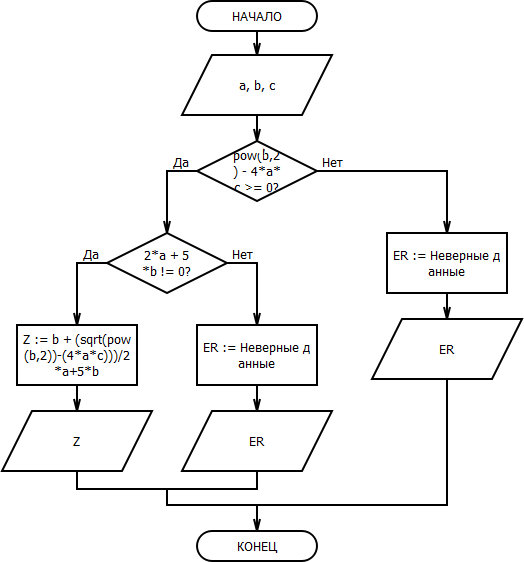
Если ввести значения, подставив которые мы получим отрицательный корень или ноль в знаменателе, то результат выйдет следующий:



Иначе программа выведет верное значение Z:



* 1. Код 2. Цикл с предусловием Блок-схема:



Код:

#include <iostream> #include <clocale> #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 {

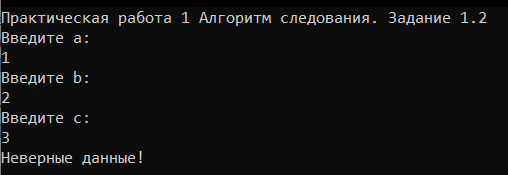
cout << "Неверные данные!" << endl;

}

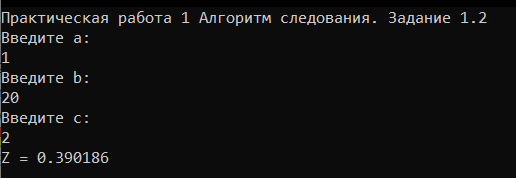
return 0;

}

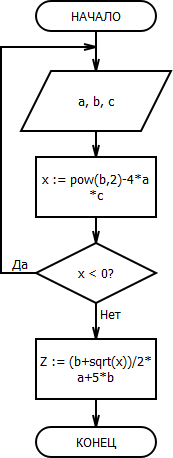
* + 1. Скрины работы программы

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

Введем корректные данные и получим ответ:



* 1. Код 3. Цикл с постусловием Блок-схема:



Код:

#include <iostream> #include <clocale> #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;

}

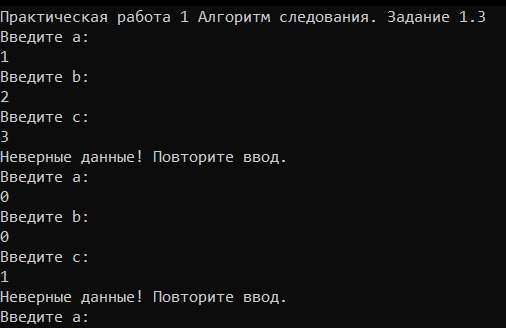
} while (x1<0 || x2==0);

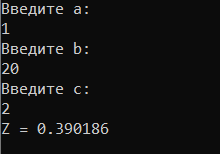
Z = (b + (sqrt(pow(b,2)-(4\*a\*c))))/(2\*a+5\*b); cout << "Z = " << Z << endl;

return 0;

}

* + 1. Скрины работы программы

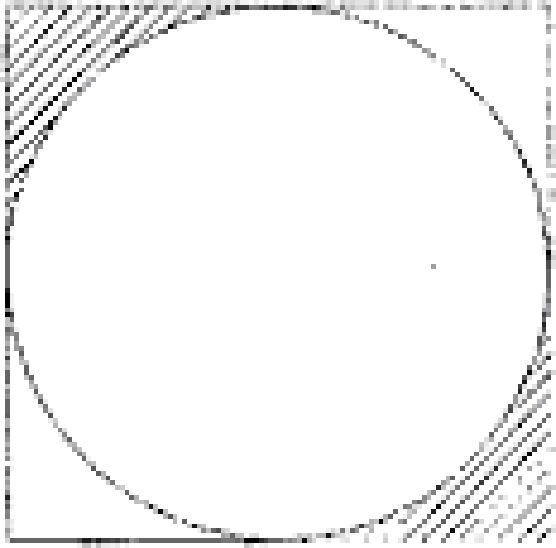
Снова введем такие значения, которые дадут отрицательный корень или ноль в знаменателе:

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:

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

1. ЗАДАНИЕ №2. ВАРИАНТ 7
   1. Формулировка задания

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



* 1. 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** | 𝑺кв | 𝑺окр | 𝑺ф |
| **1 вариант** | 2 | 4 | 3.14 | 0.43 |
| **2 вариант** | 5 | 25 | 19.625 | 2.6875 |

Рассмотрим вариант 1. Ввели значение а = 2.

Зная значение a, мы можем узнать площадь квадрата:

𝑆кв = 𝑎2 = 22 = 4

Зная, что:

𝑎

𝑅 = 

2

Преобразуем формулу площади окружности:

𝑆окр = 𝜋 ∗ 𝑅2 =

𝜋 ∗ 𝑎2 4

3.14 ∗ 22

=

4

= 3.14

Для того, чтобы узнать площадь заштрихованной части фигуры, необходимо вычесть Sокр из Sкв и взять половину от полученного результата:

𝑆ф =

𝑆кв − 𝑆окр

=

2

4 − 3.14

=

2

0.86

= 0.43

2

Менее подробно рассмотрим вариант 2. Ввели значение а = 5.

𝑆кв = 𝑎2 = 25

𝜋 ∗ 𝑎2

𝑆окр =

= 19.625

4

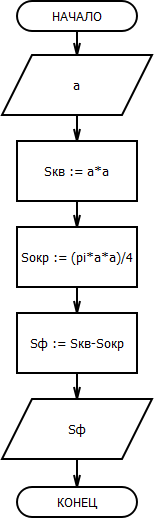
𝑆ф =

25 − 19.625

= 2.6875

2

* 1. Код 1. Без цикла Блок-схема:



Код:

#include <iostream> #include<clocale> #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;

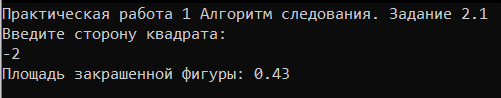
S\_okr = (pi \* a \* a)/4; S\_f = (S\_kv - S\_okr)/2;

cout << "Площадь закрашенной фигуры: " << S\_f << endl; return 0;

}

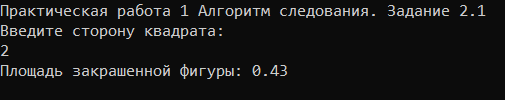
* + 1. Скрины работы программы

Если ввести отрицательное значение a, то результат выйдет следующий:

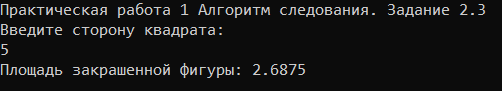


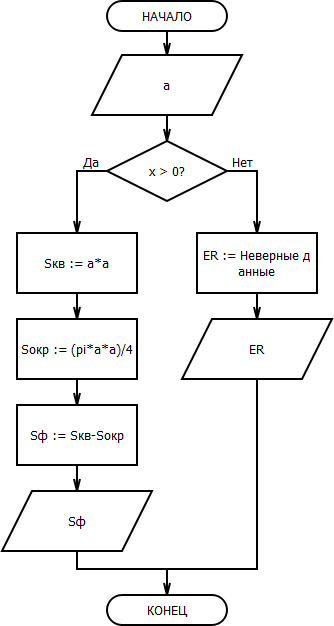
Как можно заметить, программа не обрабатывает ошибку и выводит результат, который получается, если бы а было положительным.

Можем убедиться в этом:



Введем а=5



* 1. Код 2. Цикл с предусловием Блок-схема:

Код:

#include <iostream> #include<clocale> #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 {

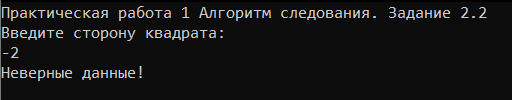
cout << "Неверные данные!" << endl;

}

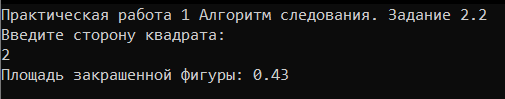
return 0;

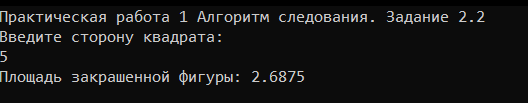
}

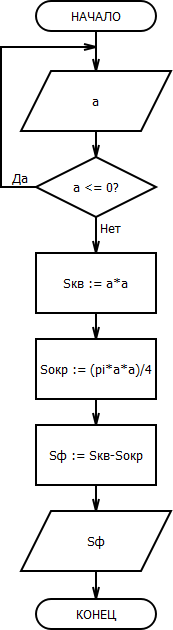
* + 1. Скрины работы программы

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

Введем корректные данные и получим ответ:





* 1. Код 3. Цикл с постусловием Блок-схема:

Код:

#include <iostream> #include <clocale> #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;

}

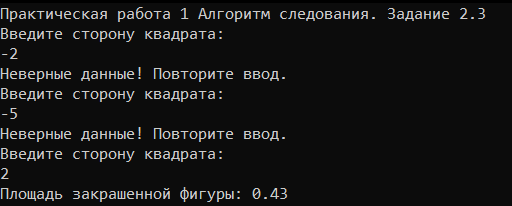
} while (a<0); S\_kv = a\*a;

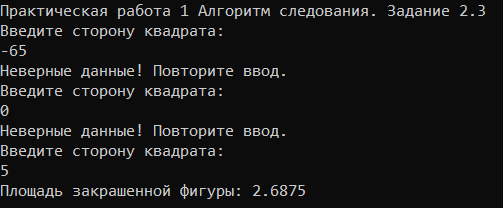
S\_okr = (pi \* a \* a)/4; S\_f = (S\_kv - S\_okr)/2;

cout << "Площадь закрашенной фигуры: " << S\_f << endl; return 0;

}

* + 1. Скрины работы программы

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



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