**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«Сибирский государственный университет науки и технологий**

**имени академика М.Ф. Решетнева»**

институт заочного обучения

институт/ факультет/ подразделение

ИУС

кафедра

**Отчёт по лабораторная работа №1**

По дисциплине Технология программирования

Выполнил студент группы БИМВ 23-01

Очно-заочной формы обучения

Сибгатулин Р.Р.

Руководитель:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата сдачи: \_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2025 г.

Дата защиты: \_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2025 г.

Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

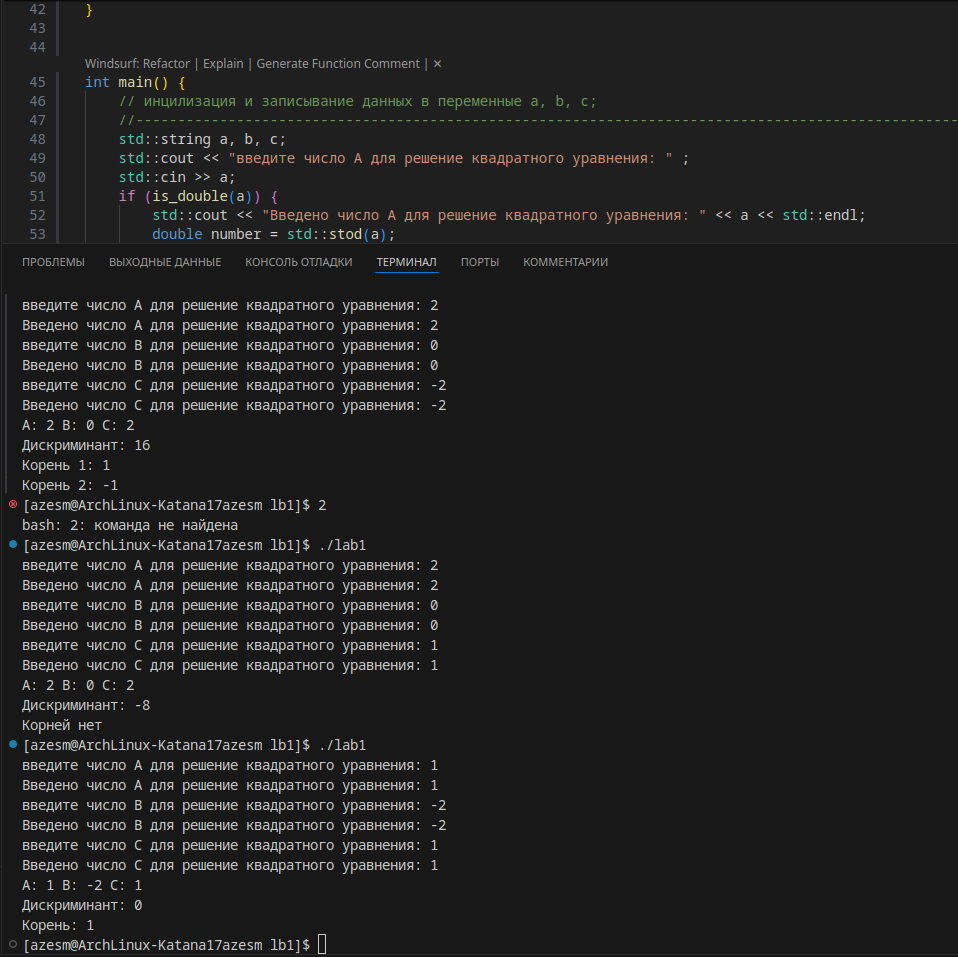
(подпись руководителя)

Красноярск 2025 г.

**Цель лабораторной работы:** Изучение основ работы в среде программирования: возможности текстового редактора; Средства отладки и тестирования приложения.

**Индивидуальные задания**

**Дано:** **А**, **B**, **С** – коэффициенты квадратного уравнения **АХ2+BХ+С=0**. Разработать программу нахождения его действительных корней, работающую в консольном режиме.



**Вывод:** Изучил основы работы в средах програмиирования: возможности текстового редактора; Средства отладки и тестирования приложения.

## Контрольные вопросы

1. Как выполнить трассировку без захода в процедуру?

Чтобы выполнить трассировку без захода в процедуру, используйте команду "Обход" (Step Over) — она выполнит вызов процедуры целиком и перейдёт к следующей строке, не заходя внутрь.

1. Как определить текущее значение переменной в ходе трассировки?

Во время трассировки текущее значение переменной можно определить, наведя курсор на переменную или просмотрев её в окне переменных (watch/locals) в отладчике. Также можно ввести имя переменной в командную строку отладчика.

1. Как сохранить исходный файл?

Чтобы сохранить исходный файл, нужно:

* В графическом редакторе/IDE: нажать Ctrl + S или выбрать "Файл → Сохранить".
* В текстовом редакторе (например, nano, vim): использовать соответствующую команду (например, :w в vim).

1. Как прочитать исходный файл?

используйте редактор или команду cat, less, nano и т. д., чтобы прочитать содержимое файла.

1. Как получить исполняемый файл?

используйте компилятор (например, gcc, fpc, javac) для создания исполняемого файла.

1. Сформулировать правила составления протокола?

протокол должен содержать дату, место, участников, повестку, обсуждение, решения и подписи.

1. Сформулировать назначение исходного файла программы.

Назначение исходного файла программы — хранить текст программы на языке программирования, предназначенный для чтения, редактирования и компиляции.

Он содержит алгоритмы, описанные в виде команд, которые позже преобразуются в исполняемый файл.

1. Сформулировать назначение исполняемого файла программы.

Назначение исполняемого файла программы — запуск и выполнение программы на компьютере без необходимости компиляции.

Он содержит машинный код, понятный процессору.

1. В приведенном ниже примере протокола прохождения первого контрольного примера исправьте ошибки (несоответствие результатам трассировки программы, отсутствие или ошибочность описания операции, выполняемой программой, отсутствие или ошибочность описания результатов выполнения операции):
   1. Чтение с клавиатуры значений переменных **A=2, B=0, C=-2**;
   2. Вычисление **D=Sqr(B)-4\*A\*C=16.0**;
   3. Проверка **D<0**, условие не выполняется
   4. Проверка **D=0**;
   5. Вычисление ;



* 1. Печать сообщения «**X1=-2, X2=2**» ;
  2. Задержка, управляемая пользователем, ввод «Enter»;
  3. Конец сеанса работы программы.

а. Чтение с клавиатуры значений переменных: A = 2, B = 0, C = -2.  
(Верно. Пользователь вводит значения. Программа сдержанно принимает их без возмущения.)

b. Вычисление дискриминанта:  
D = Sqr(B) - 4\*A\*C = 0² - 4\*2\*(-2) = 16.0.  
(Верно. Всё по учебнику, никакой магии — чистая арифметика.)

c. Проверка условия D < 0:  
 Условие не выполняется, так как D = 16 > 0.  
(Корректно. Программа — не драматург, отрицательных поворотов не будет.)

d. Проверка условия D = 0:  
 Условие не выполняется, так как D ≠ 0.  
(Добавим явное указание результата проверки — ясность прежде всего.)

e. Вычисление корней квадратного уравнения по формуле:  
X1 = (-B + √D) / (2\*A) = (0 + 4) / 4 = 1,  
X2 = (-B - √D) / (2\*A) = (0 - 4) / 4 = -1.  
 🛑 Ошибка в оригинале: корни не равны -2 и 2. Это совершенно иные сущности. Подозреваю, их звали на собеседование в другую программу.

f. Печать сообщения:  
"X1 = 1.0, X2 = -1.0".  
(Исправлено согласно реальному результату вычислений.)

g. Задержка, управляемая пользователем — программа ожидает нажатия Enter.  
(Ничего страшного, просто кофе допить.)

h. Завершение сеанса работы программы.  
 \*(Вежливо попрощалась и ушла в память.)