МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «КубГУ»)

Факультет компьютерных технологий и прикладной математики
Кафедра вычислительных технологий

ЛАБОРАТОРНАЯ РАБОТА №1
Дисциплина: Платформо-независимое программирование

Работу выполнил: _____ А. А. Костров

Направление подготовки: 02.03.02 Фундаментальная информатика и информационные технологии

Преподаватель: _____ В. И. Шиян

Краснодар
2026

Тема. Работа в командной строке, компиляция и запуск на выполнение.

Цель. Освоить основы работы в командной строке Java. Изучить структуру и синтаксис простой программы. Посмотреть возможности языка.

1. Напишем, скомпилируем и выполним программу *Калькулятор* и примеры из командной строки (рисунки 1-14).

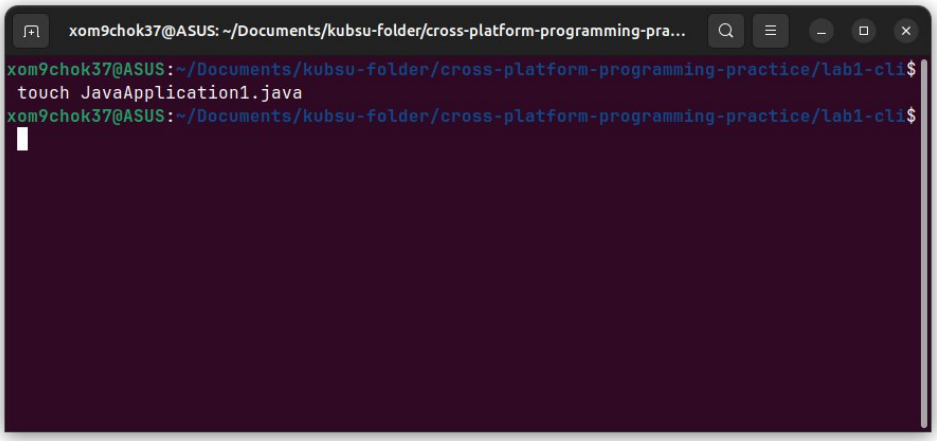

Рисунок 1 — Создадим файл *JavaApplication1.java*

Рисунок 2 — Напишем код программы *Калькулятор*

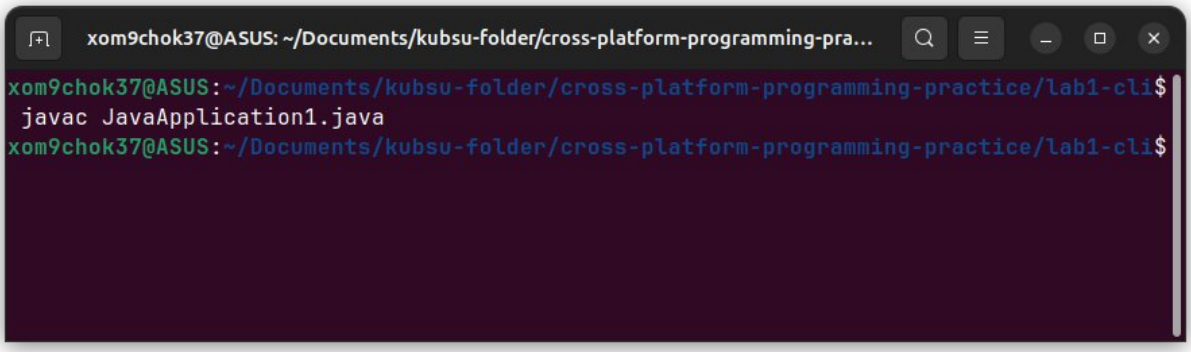

Рисунок 3 — Скомпилируем программу

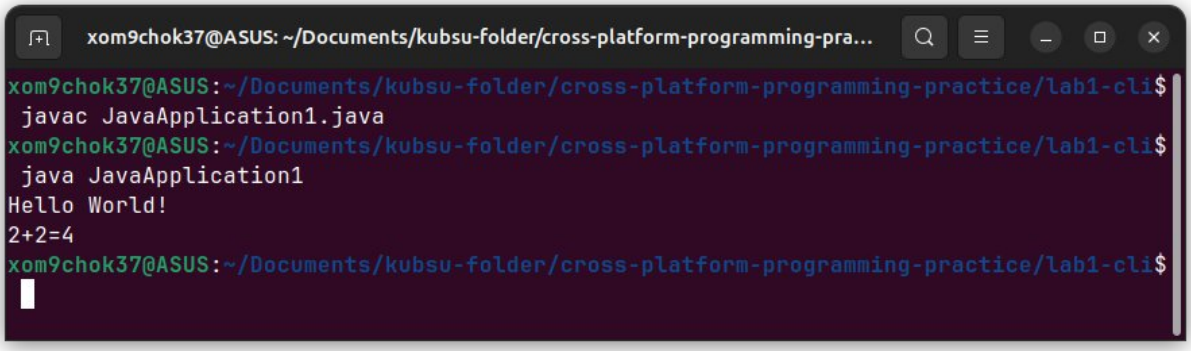

Рисунок 4 — Выполним программу *Калькулятор*

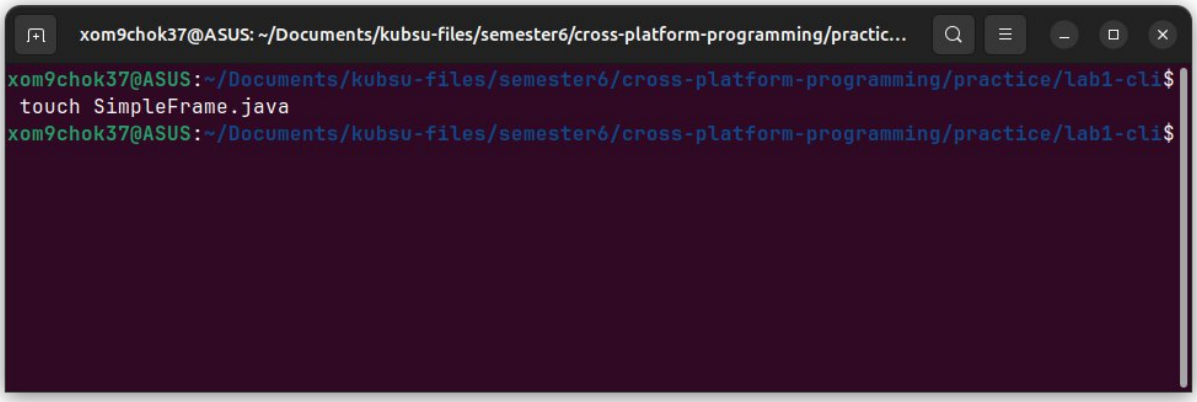

Рисунок 5 — Создадим файл *SimpleFrame.java*

Open — SimpleFrame.java
~/Documents/kubsu-files/semester6/cross-platform-programming/practice/lab1-cli

```
import java.awt.*;
import java.awt.event.*;

// Создаем наш класс, наследуясь от стандартного класса Frame
class SimpleFrame extends Frame {
  SimpleFrame(String s) {
    super(s);
    // Передаем название окна в родительский класс
    setSize(400, 150); // Устанавливаем размер окна
    setVisible(true); // Окно становится видимым
    // Добавляем слушателя, который будет реагировать на кнопку
    // закрытия окна
    addWindowListener(new WindowAdapter() {
      public void windowClosing(WindowEvent ev) {
        dispose(); // Закрываем окно
        System.exit(0); // Выходим из программы
      }
    });
    addWindowListener(new WindowAdapter() {
      public void windowClosing(WindowEvent ev) {
        dispose();
        System.exit(0);
      }
    });
  }
  // Создаем экземпляр нашего класса в главном классе
  public static void main(String[] args) {
    new SimpleFrame(" Моя программа");
  }
}
```

Рисунок 6 — Напишем код программы

xom9chok37@ASUS: ~/Documents/kubsu-files/semester6/cross-platform-programming/practic...

```
xom9chok37@ASUS:~/Documents/kubsu-files/semester6/cross-platform-programming/practice/lab1-cli$
 javac SimpleFrame.java
xom9chok37@ASUS:~/Documents/kubsu-files/semester6/cross-platform-programming/practice/lab1-cli$
```

Рисунок 7 — Скомпилируем пример 1.1

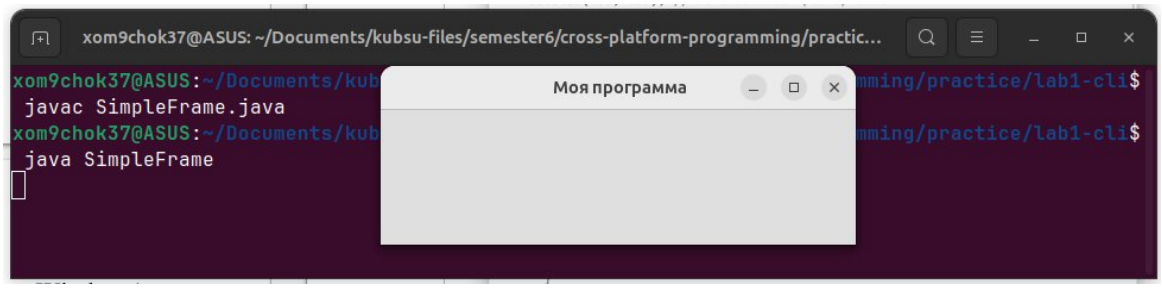

Рисунок 8 — Запустим пример 1.1

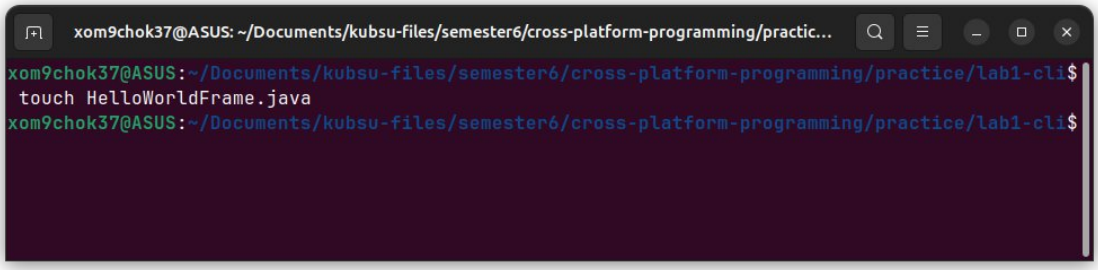

Рисунок 9 — Создадим файл *HelloWorldFrame.java*

Рисунок 10 — Напишем код программы из примера 1.2

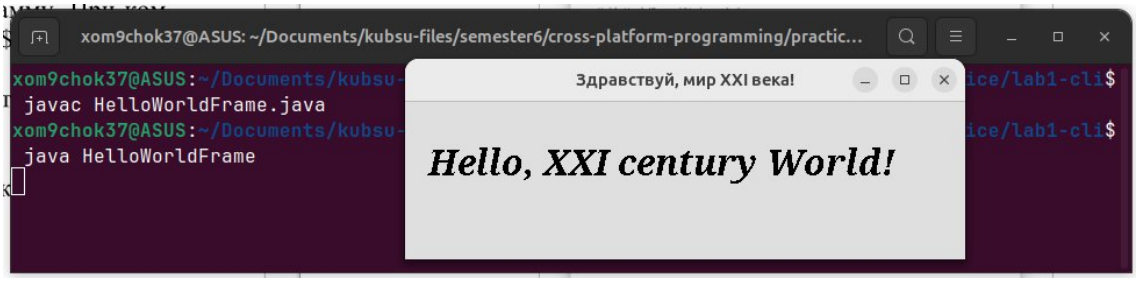

Рисунок 11 — Скомпилируем и запустим программу

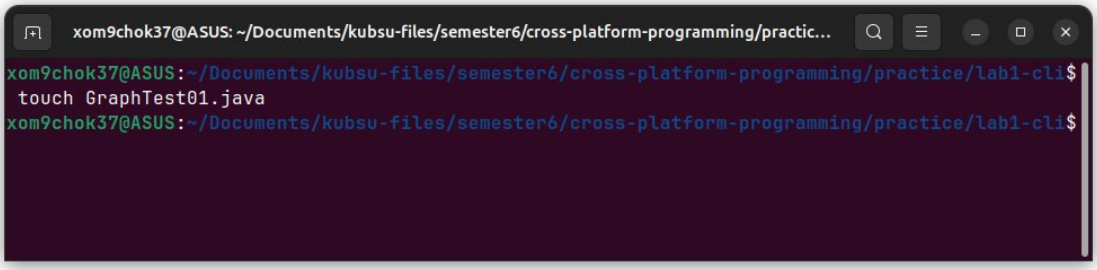

Рисунок 12 — Создадим файл *GraphTest01.java*

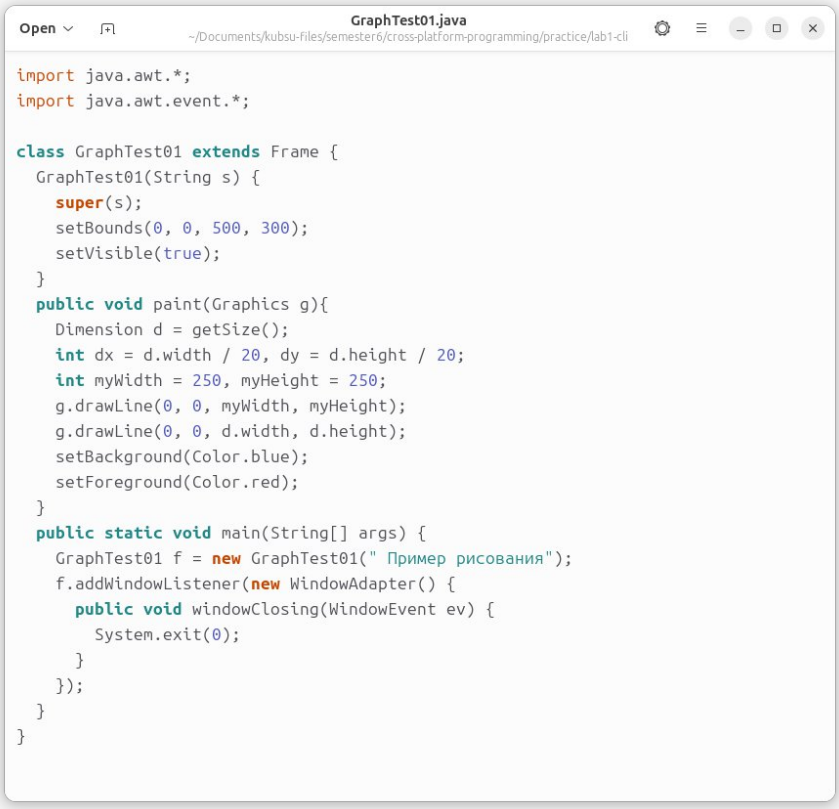

Рисунок 13 — Напишем код программы из примера 1.3

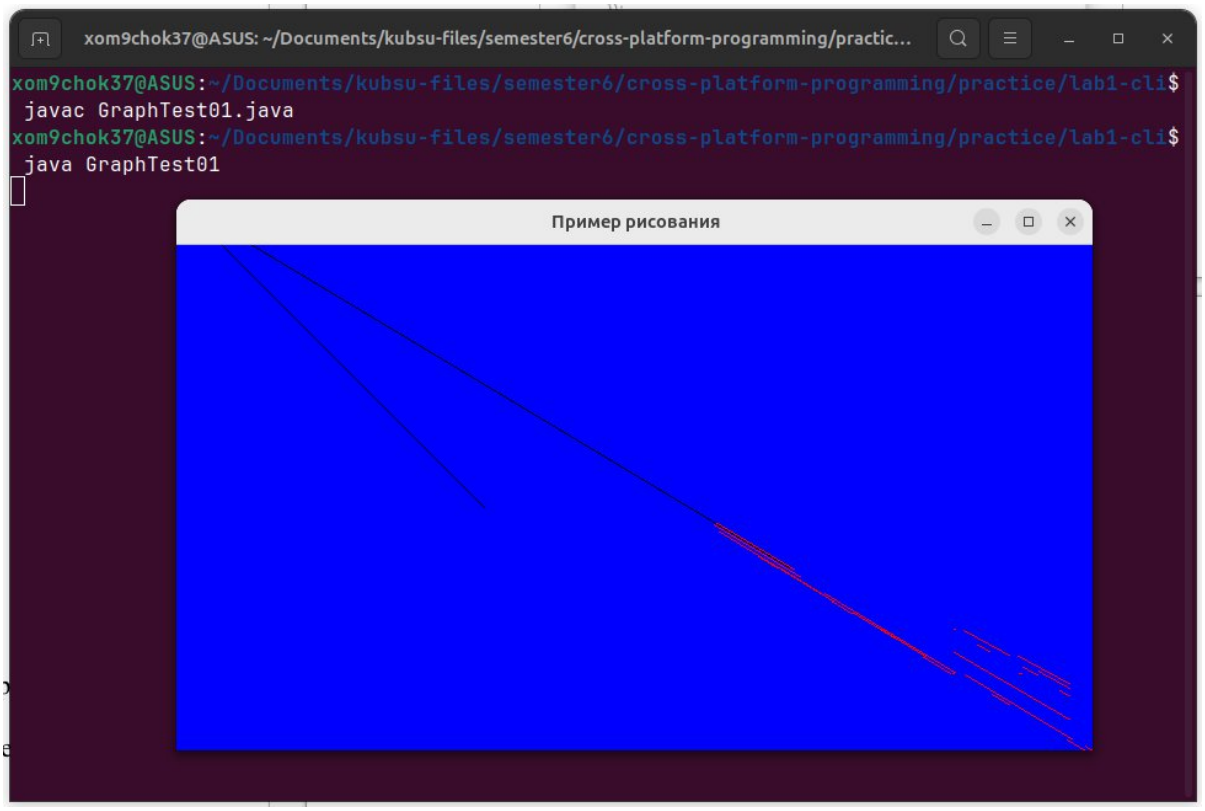

Рисунок 14 — Скомпилируем и запустим программу

2. Напишем программу, рисующую картинку из примитивных фигур (рисунок 15).

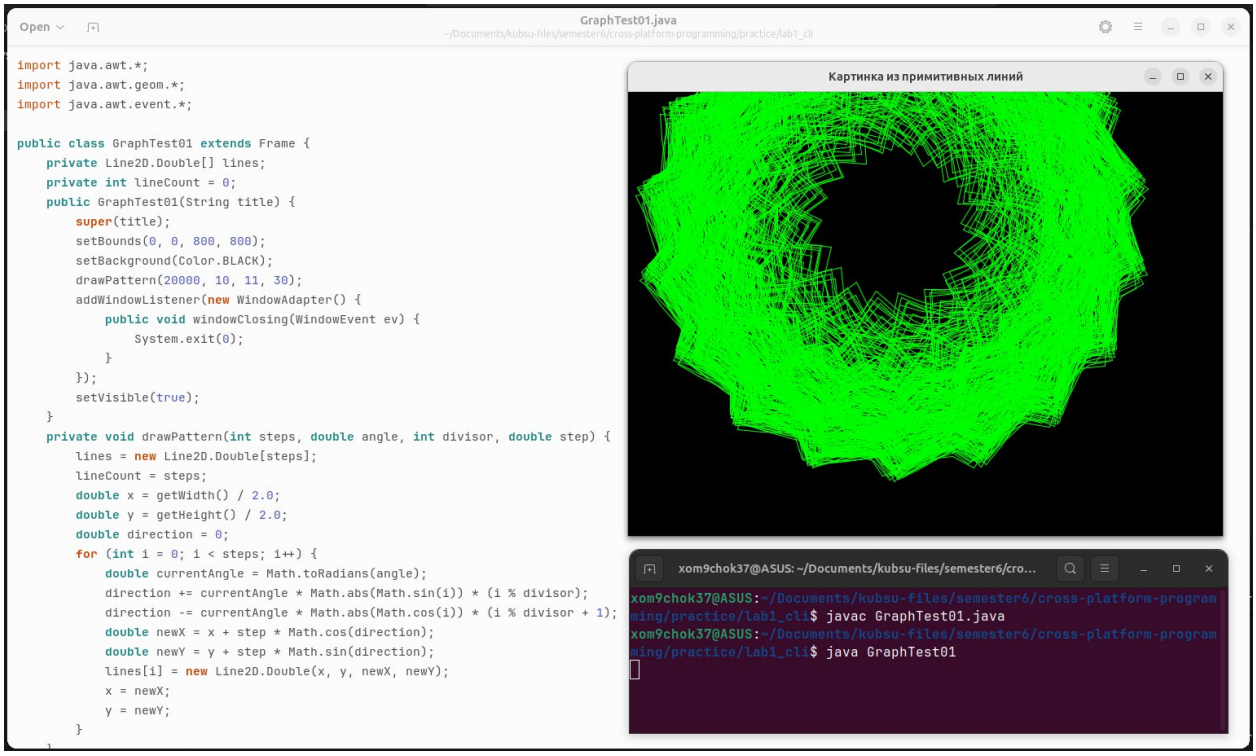

Рисунок 15 — Скомпилируем и запустим программу

3. Создаём структуру приложения и компилируем (рисунки 16-17).

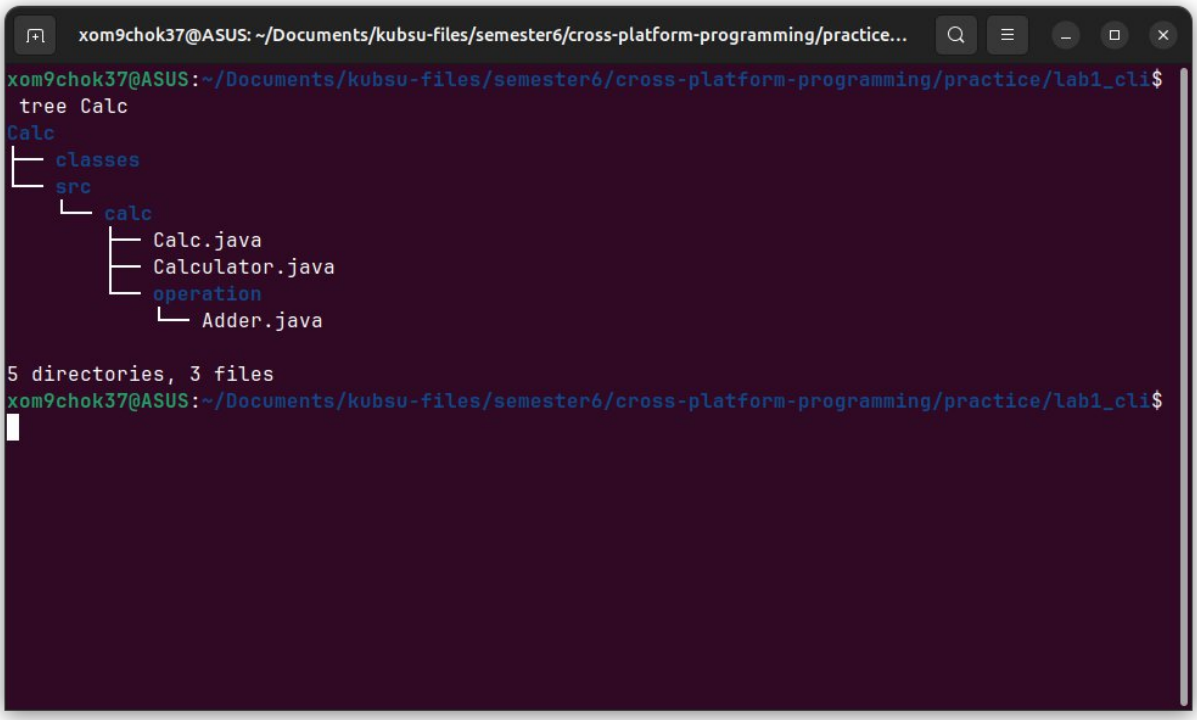

Рисунок 16 — Строим структуру

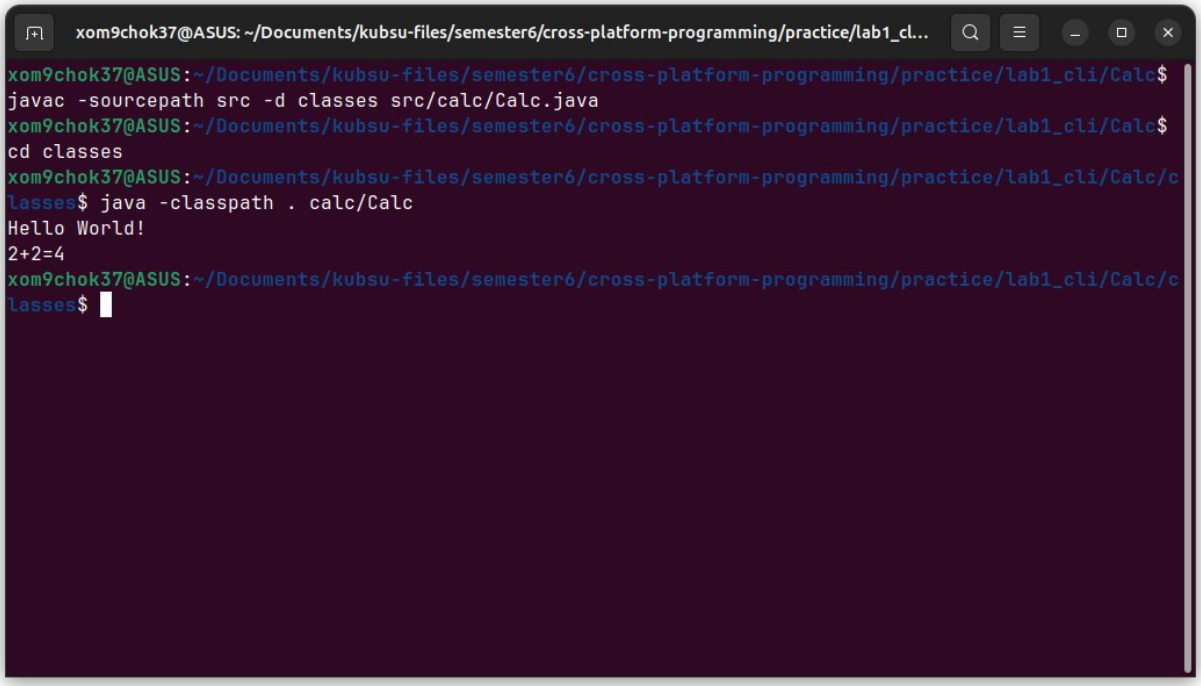

Рисунок 17 — Скомпилируем и запустим программу

4. Добавляем дополнительные классы, скомпилируем и запустим программу (рисунки 18-20).

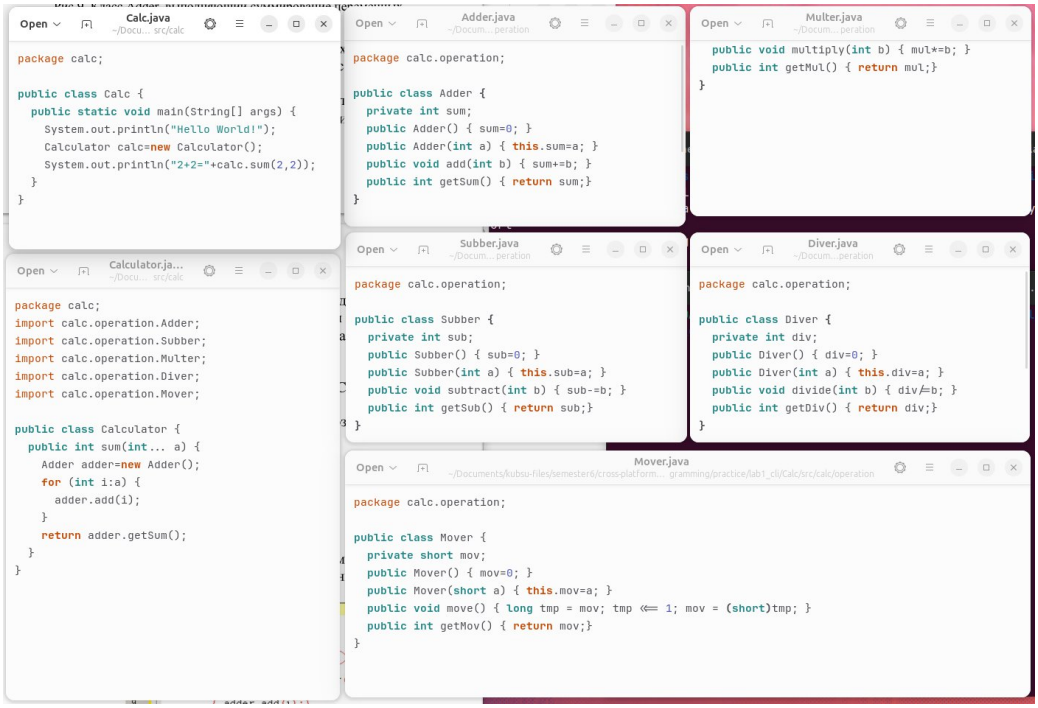

Рисунок 18 — Создаём новые классы

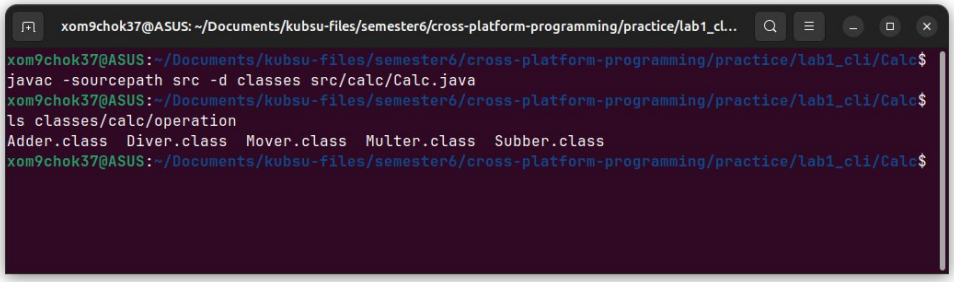

Рисунок 19 — Скомпилируем программу

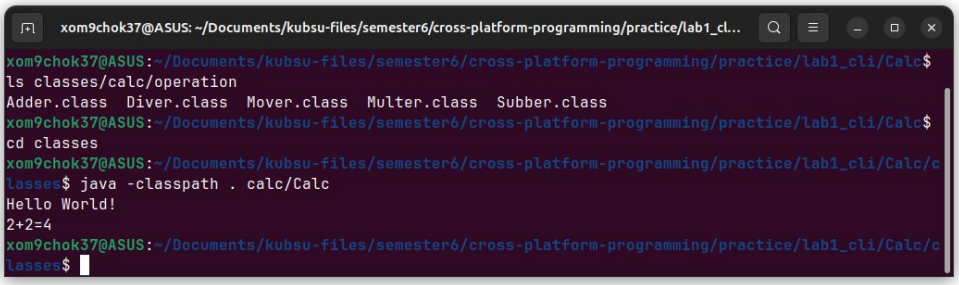

Рисунок 20 — Запустим программу

5. Добавим манифест в корневой каталог нашей программы, затем соберём программу в *JAR* и запустим её (рисунок 21).

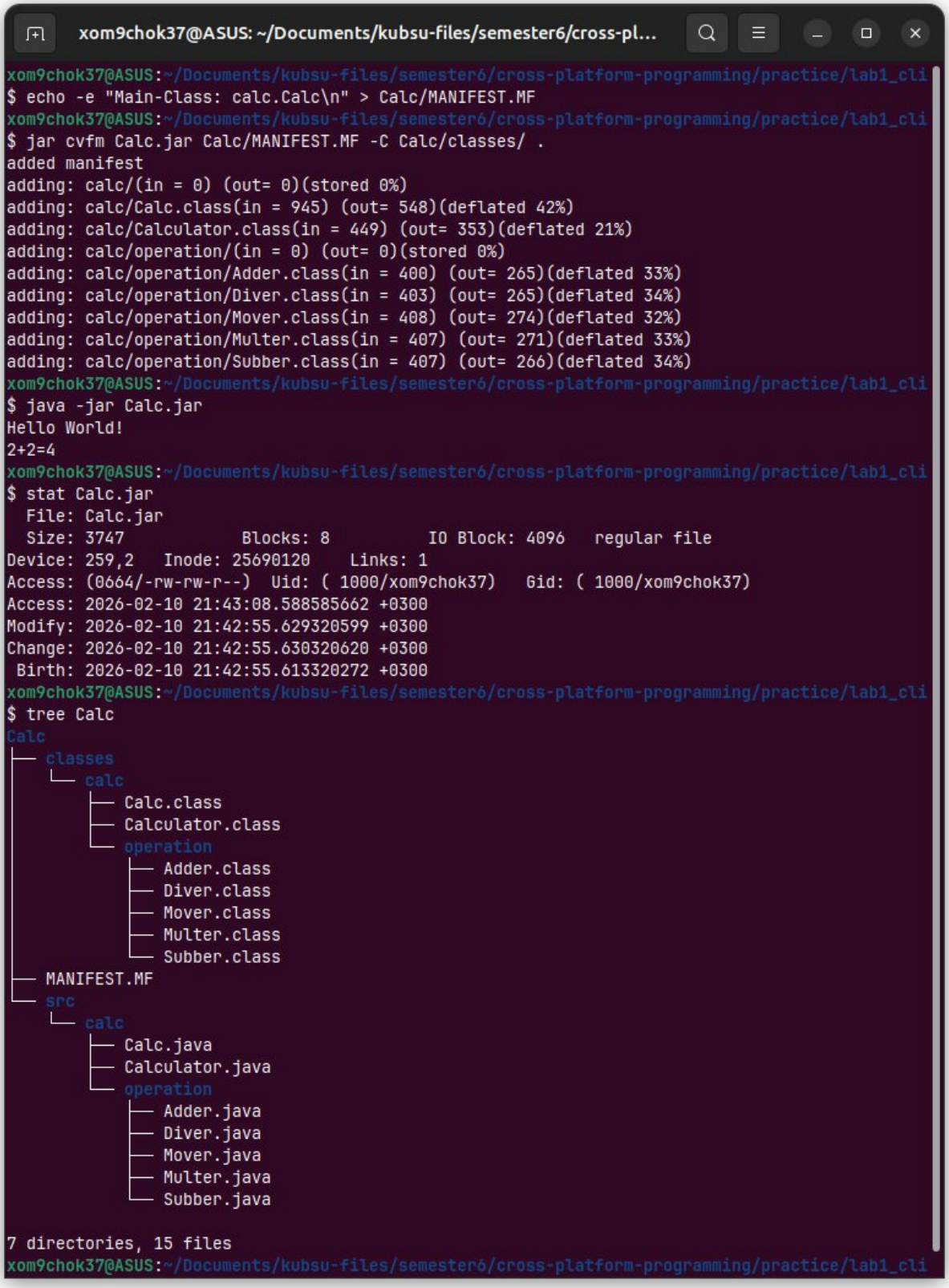

Рисунок 21 — Создаём манифест и собираем программу