МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»

Факультет радіоелектроніки, комп'ютерних систем та інфокомунікацій

Кафедра комп'ютерних систем, мереж і кібербезпеки

Лабораторна робота

з Кросплатформенні технології (назва дисципліни)

на тему: «Інструментальні засоби розроблення для платформонезалежного програмування»

Виконав: студент <u>4</u> курсу групи № <u>545б</u>
напряму підготовки (спеціальності)
123 – комп'ютерна інженерія
• •
(шифр і назва напряму підготовки (спеціальності))
Поліщук А.О.
(прізвище й ініціали студента)
Прийняв: асистент каф.503
Годованюк П.А.
(посада, науковий ступінь, прізвище й ініціали)
Національна шкала:
Кількість балів:
Оцінка: ECTS

- 1. Постановка задачи
- 1.1 Ознакомиться с интегрированной средой разработки IntelliJ IDEA
- 1.2 Ознакомиться с интегрированной средой разработки Eclipse.
- 2. Создание простого приложения с помощью IntelliJ IDEA

Сначала был создан проект по шаблону «Console application», дальше добавил в метод таіп вывод простого текста на терминал и построил проект.

Скриншот дерева файлов проекту и коду приведен на рисунке 1.

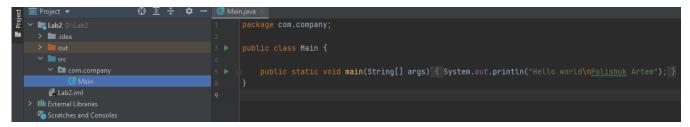


Рисунок 1 – Дерево файлов и код проекта

Скриншот работы проекта по выводу результата в терминал изображено на рисунке 2 .

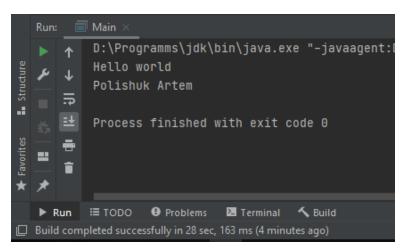


Рисунок 2 – Вывод результата в терминал

3. Основы разработки приложения на Java в IDE Eclipse

Изначально создали 2 проекта, добавили библиотеку в проект основной программы, а далее написали в главной программе простой код.

Код приведен на рисунке 3

```
LibClass.java

package acrostic;

import org.me.mylib.LibClass;

public class Main {

public static void main(String[] args) {

String result = LibClass.acrostic(args);

System.out.println("Result = " + result);

}

}
```

Рисунок 3 – Код метода таіп

Теперь добавим входным параметром в программу текст, чтобы в результате выполнения программы была получено строчка — Hello(изображено на рисунке 4).

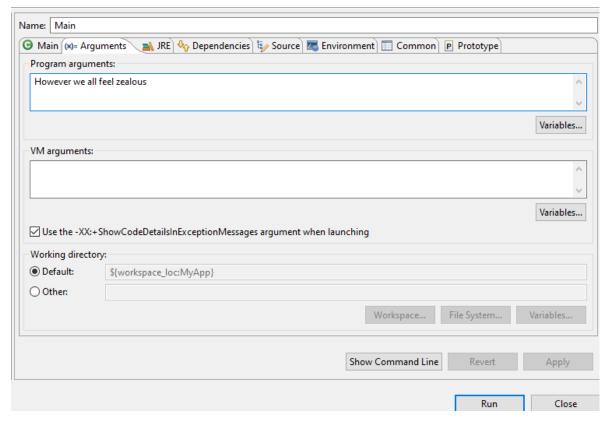


Рисунок 4 – Установка входных аргументов для приложения

Далее добавим комментарии к коду и сгенирируем документацияю, пример генерации приведен на рисунке 5.

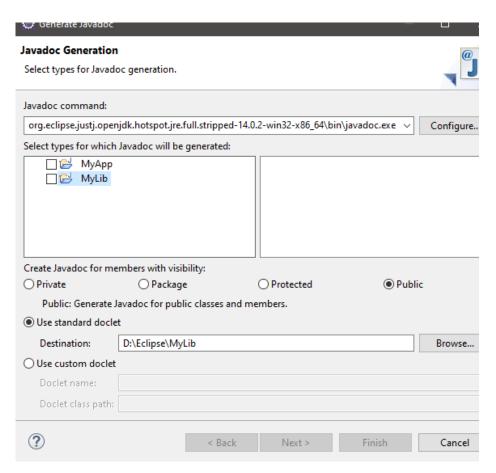


Рисунок 5 – Окно генерации javadoc

Теперь создадим тест для метода библиотеки, и выполним его (Результат на рисунке 6).



Рисунок 6 – Скриншот теста и результат его выполнения

Теперь попробуем добавить breakpoint в библиотеку, и на этой точке остановки посмотреть значения вссез переменных, результат этих действий приведен на рисунке 7.

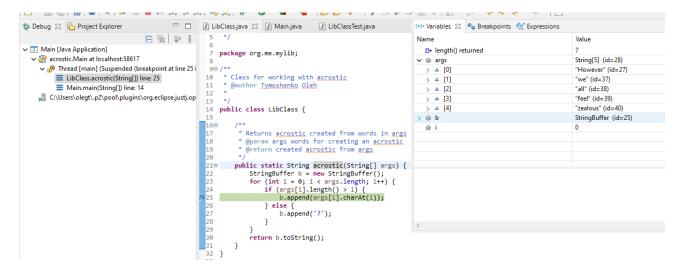


Рисунок 7 – Отладка приложения с изображением значений всех переменных

Структура пректа после всех проведенных действий на рисунке 8.

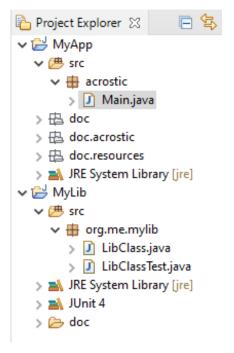


Рисунок 8 – Структура проекта после всех действий

4. Основи розроблення Java застосунків в Eclipse IDE

Package Explorer с добавленным архивом Junit и развернутым файлом TestCase.java представленный на рисунке 9.

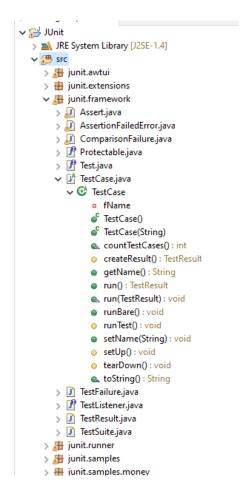


Рисунок 9 – Package Explorer с добавленным архивом Junit и развернутым файлом TestCase.java

Согласно заданию, далее был добавлен новый метод VectorTest.java. Скриншот метода в коде и новой структуры проекта на рисунке 10

```
⊫ Package Explorer ८১
                                                         📼 😂 | ⊕″ 8 — Li 🔐 vectoriest.java ८८
                                                                                                   assertTrue(fFull.contains(new Integer(1)));
                                                                                   239
🗸 📂 JUnit
                                                                                   0₄40
41
   > A JRE System Library [J2SE-1.4]
                                                                                                   assertTrue(!fEmpty.contains(new Integer(1)));
   public void testElementAt() {| |
    Integer i= (Integer)fFull.elementAt(0);
    assertTrue(i.intValue() == 1);
     > 🔠 junit.awtui
     > 🌐 junit.extensions
      > 🔠 junit.framework
                                                                                                   > Æ junit.runner
      } catch (ArrayIndexOutOfBoundsException e) {
        > 🚺 AllTests.java
         > 🔝 SimpleTest.java
                                                                                                   fail("Should raise an ArrayIndexOutOfBoundsException");
         ✓ 🕖 VectorTest.java

√ 

R

VectorTest

                                                                                              public void testRemoveAll() {
                 S main(String[]): void
                                                                                                   fFull.removeAllElements();
fEmpty.removeAllElements();
assertTrue(fFull.isEmpty());
                 S suite() : Test

    fEmpty

                o fFull
                                                                                                   assertTrue(fEmpty.isEmpty());
                🙉 setUp() : void
                                                                                              public void testRemoveElement() {
    ffull.removeElement(new Integer(3));
    assertTrue(!ffull.contains(new Integer(3)) );
                testCapacity(): void
                testClone(): void
                testContains(): void
                                                                                              public void testSizeIsThree () {
                 testElementAt() : void
                 testRemoveAll(): void
                                                                                                   //TODO: Check size
                testRemoveElement(): void
                 testSizeIsThree(): void
```

Рисунок 9 – Скриншот метода в коде и новой структуры проекта

Теперь внесем в код ошибку, удалив символ ';' в файле TestCase.java и сохраним файл. В результате IDE подсветит ошибки в нескольких местах. Скриншот ошибок на рисунке 10.

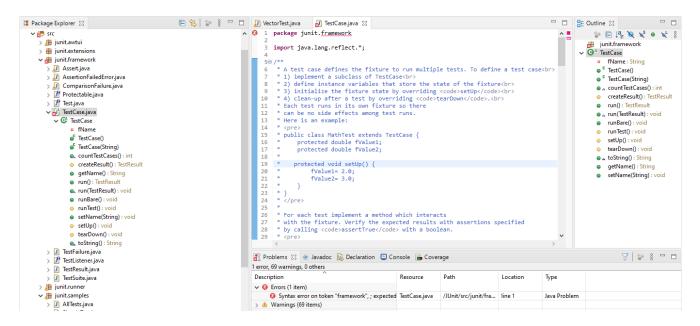


Рисунок 10 – Скриншот ошибок в нескольких местах

Далее был создан новый метод testValues, и в нем был написан код с помощью шаблонов, результат на рисунке 11.

```
🚦 Package Explorer 🛭 🕒 💈 🦻 🖁 🗀 📋 *TestCase.java 🔑 VectorTest.java 🛭

    JONIT
    → M JRE System Library [J2SE-1.4]
    ✓ 炒 src
                                                                                                                                                                                                                                                                                           public void testRemoveAll() {
                                                                                                                                                                                                                                                                                   iunit.samples
                                                                                             fFull.removeAllElements();
fEmpty.removeAllElements();
assertTrue(fFull.isEmpty());
assertTrue(fEmpty.isEmpty());
                                                                                                                                                                                                                                                                         > 🔠 junit.awtui
> 🖶 junit.extensions
                                                                                                                                                                                                                                                                                       fFull: Vector
                                                                                      }
public void testRemoveElement() {
    fFull.removeElement(new Integen(3));
    assertTrue(!fFull.contains(new Integen(3)) );
        > 🔠 junit.framework
                                                                                                                                                                                                                                                                                   S main(String[]): void
        > # junit.runner
> # junit.samples
                                                                                                                                                                                                                                                                                  setUp() : void
S suite() : Test
           > AllTests.java
> SimpleTest.java
> VectorTest.java
                                                                                     }
public void testSizeIsThree () {
    assertTrue (ffull.size () == 3);
    Vector v = new Vector ();
    for (int i = 0; i <3; i ++)
    vaddElement (new Object ());
    Assert.assertTrue(v.size() == fFull.size());
}</pre>
                                                                                                                                                                                                                                                                                  testClone(): void
                                                                                                                                                                                                                                                                                  testContains(): void
        > # junit.samples.money
                                                                                                                                                                                                                                                                                   testElementAt(): void
         testRemoveAll(): void
                                                                                                                                                                                                                                                                                  testRemoveElement(): void
        > # junit.tests
                                                                                                                                                                                                                                                                                        testSizeIsThree(): void
                                                                                     testValues(): void
           iunit.tests.extensions
        > 🔠 junit.tests.runner
                                                                                            expected [e] = new arrange. tv...
}
Integer [] actual = (Integer[]) ffull.toArray(new Integer[fFull.size()]);
assertEquals (expected.length, actual.length);
for (int i = 0; i (actual.length; i ++)
    assertEquals (expected [i], actual [i]);
          # junit.textui
```

Рисунок 11 — Скриншот созданного с помошью шаблонов метода и структуры проекта

Далее создали новый пакет, класс, добавили методы, свойства, геттер и сеттер. Скриншот кода и структуры проекта в результате приведен на рисунке 12.

```
🖹 😤 🐉 🖁 🗀 🖟 VectorTest.java 🖟 MyTestCase.java 🎗
                                                                                                                                                                       E Outline ⋈ 💝 🖹 🞝 💘 🤘
                                                                      package test;
🗸 📂 JUnit
                                                                                                                                                                        # test

• • MyTestCase
  > M JRE System Library [J2SE-1.4]
                                                                    3⊖ import java.util.Vector;
                                                                                                                                                                              container : Vec
     > 🔠 junit.awtui
                                                                   5 import junit.framework.TestCase;
     > # junit.extensions
                                                                                                                                                                              MyTestCase(String)
                                                                   7 public class MyTestCase extends TestCase {
     > # junit.framework
                                                                                                                                                                             o setUp(): void
     > 🔠 junit.runner
                                                                                                                                                                              o _ tearDown() : void
                                                                           private Vector container;
                                                                                                                                                                             getContainer() : Vector
     public MyTestCase() {
    // TODO Auto-generated constructor stub
       > 🚺 AllTests.java
                                                                                                                                                                              setContainer(Vector) : void
       > 🔝 SimpleTest.java
        VectorTest.java
     > 🌐 junit.samples.money
                                                                          public MyTestCase(String name) {
     > # junit.swingui
                                                                16
217
                                                                             super(name);
// TODO Auto-generated constructor stub
     > 🖶 junit.swingui.icons
     > i junit.tests
       iunit.tests.extensions
                                                                          protected void setUp() throws Exception {
   container = new Vectpr();
     > # junit.tests.framework
                                                                21
22
23
240
25
26
27
28
290
30
31
32
     > 🔠 junit.tests.runner
      iunit.textui
                                                                          protected void tearDown() throws Exception {
       > MyTestCase.java
                                                                              super.tearDown();
                                                                          public Vector getContainer() {
    return container;
```

Рисунок 12 – Скриншот кода и структуры проекта

5. Посилання на репозитарій із проєктом у системі керування версіями. https://github.com/PolishukArtem/Cross/tree/main/LAB2

Висновки:

В результате выполнения лабораторной работы я ознакомился с 2-мя основными IDE для разработки на Java, это IntelliJ IDEA и Eclipse. Я создал простое приложение, состоящий из основного приложения, библиотеки и юнит-теста метода библиотеки. Также я поработал с архивом Junit в Eclipse IDE, научился использовать основные возможности этой среды разработки для более быстрого и удобного написания кода. После недолгого ознакомления с этими 2-мя среде разработки, для себя я решил, что IntelliJ IDEA больше понравилась мне через большую простоту и вид дизайна ide. В плане функционала, как минимум для начального уровня, обе среды разработки дают хорошие возможности для удобного написания кода.