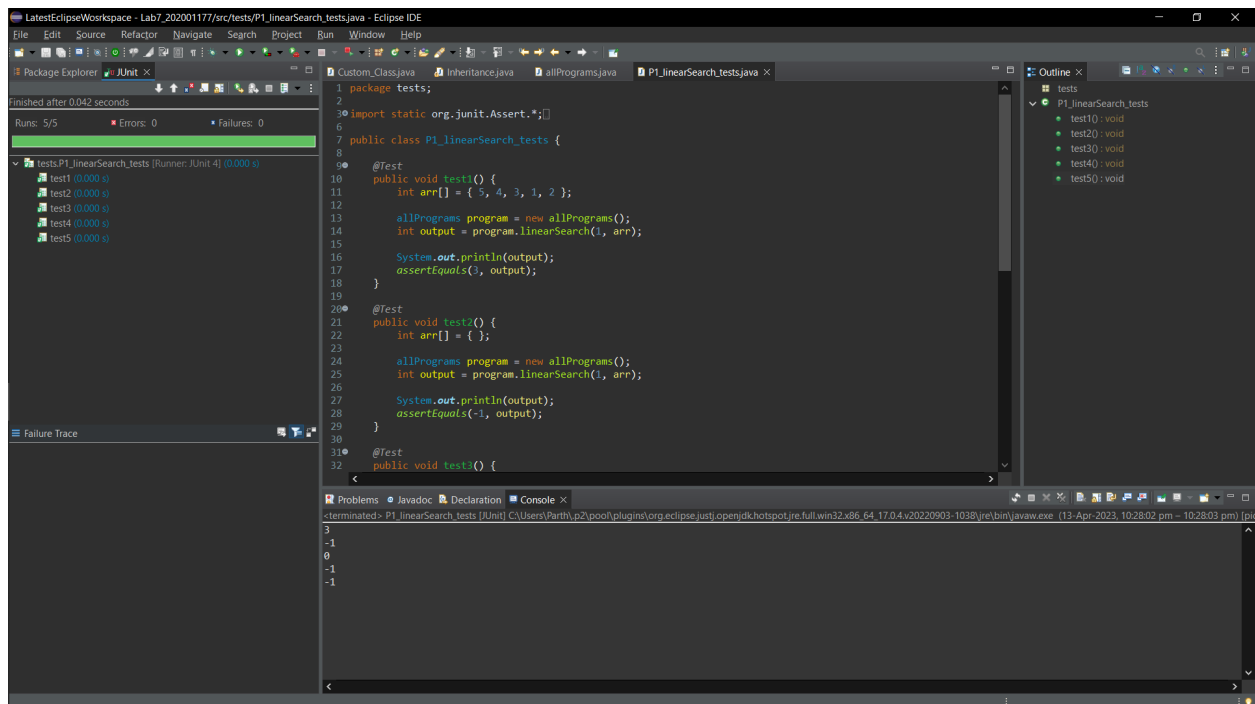


Name : Vansh Parikh

ID : 202001011

Lab : 07 (Software Engineering)

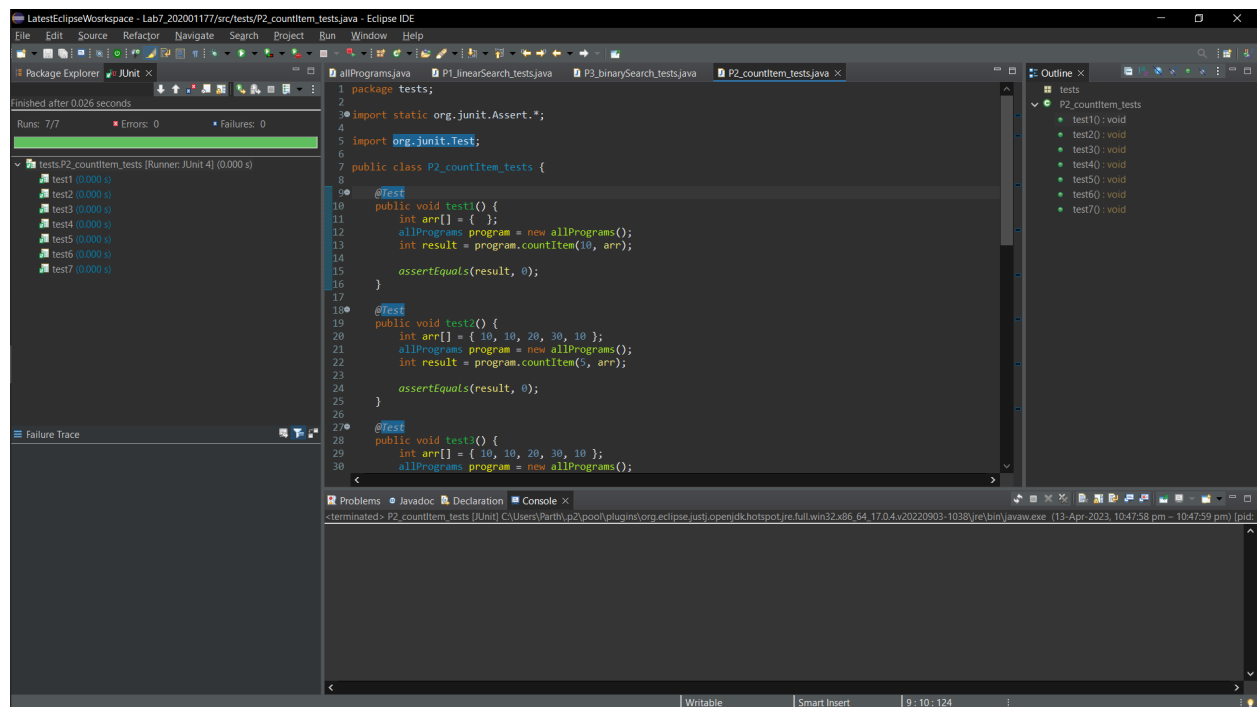
Program 1: LinearSearch



The screenshot shows the Eclipse IDE interface with a Java test class named `P1_linearSearch_tests.java` open. The class is located in the `tests` package and contains three test methods: `test1`, `test2`, and `test3`. Each test method calls `allPrograms.linearSearch` with a specific array and an expected output, and then uses `assertEquals` to verify the result. The `test1` method uses the array `{ 5, 4, 3, 1, 2 }` and expects `1`. The `test2` method uses an empty array `{ }` and expects `-1`. The `test3` method uses the array `{ }` and expects `-1`. The IDE also shows a Package Explorer on the left with the `tests` package expanded, and a Console window at the bottom showing the output of the tests.

```
1 package tests;
2
3 import static org.junit.Assert.*;
4
5 public class P1_linearSearch_tests {
6
7     @Test
8     public void test1() {
9         int arr[] = { 5, 4, 3, 1, 2 };
10
11         allPrograms program = new allPrograms();
12         int output = program.linearSearch(1, arr);
13
14         System.out.println(output);
15         assertEquals(1, output);
16     }
17
18     @Test
19     public void test2() {
20         int arr[] = { };
21
22         allPrograms program = new allPrograms();
23         int output = program.linearSearch(1, arr);
24
25         System.out.println(output);
26         assertEquals(-1, output);
27     }
28
29     @Test
30     public void test3() {
31         int arr[] = { };
32
33         allPrograms program = new allPrograms();
34         int output = program.linearSearch(1, arr);
35
36         System.out.println(output);
37         assertEquals(-1, output);
38     }
39 }
```

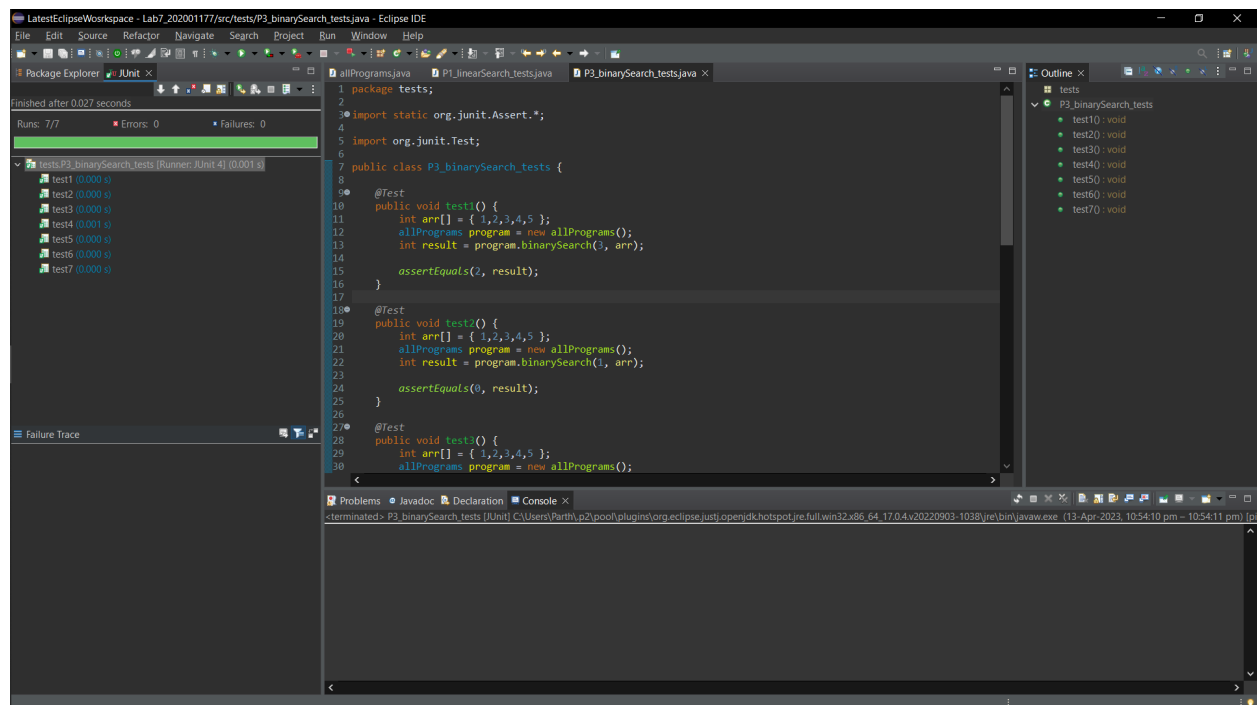
Program 2 : countItem



The screenshot shows the Eclipse IDE interface. The Package Explorer on the left shows the project structure. The Console window at the bottom left displays the test results for P2_countItem_tests, showing 7 tests passed. The Editor window shows the source code for P2_countItem_tests.java, which includes three test methods: test1, test2, and test3. The Outline view on the right shows the class structure.

```
1 package tests;
2
3 import static org.junit.Assert.*;
4
5 import org.junit.Test;
6
7 public class P2_countItem_tests {
8
9     @Test
10    public void test1() {
11        int arr[] = { };
12        allPrograms program = new allPrograms();
13        int result = program.countItem(10, arr);
14
15        assertEquals(result, 0);
16    }
17
18    @Test
19    public void test2() {
20        int arr[] = { 10, 10, 20, 30, 10 };
21        allPrograms program = new allPrograms();
22        int result = program.countItem(5, arr);
23
24        assertEquals(result, 0);
25    }
26
27    @Test
28    public void test3() {
29        int arr[] = { 10, 10, 20, 30, 10 };
30        allPrograms program = new allPrograms();
```

Program 3 : BinarySearch



The screenshot shows the Eclipse IDE interface. The Package Explorer on the left shows the project structure. The Console window at the bottom left displays the test results for P3_binarySearch_tests, showing 7 tests passed. The Editor window shows the source code for P3_binarySearch_tests.java, which includes three test methods: test1, test2, and test3. The Outline view on the right shows the class structure.

```
1 package tests;
2
3 import static org.junit.Assert.*;
4
5 import org.junit.Test;
6
7 public class P3_binarySearch_tests {
8
9     @Test
10    public void test1() {
11        int arr[] = { 1, 2, 3, 4, 5 };
12        allPrograms program = new allPrograms();
13        int result = program.binarySearch(3, arr);
14
15        assertEquals(2, result);
16    }
17
18    @Test
19    public void test2() {
20        int arr[] = { 1, 2, 3, 4, 5 };
21        allPrograms program = new allPrograms();
22        int result = program.binarySearch(1, arr);
23
24        assertEquals(0, result);
25    }
26
27    @Test
28    public void test3() {
29        int arr[] = { 1, 2, 3, 4, 5 };
30        allPrograms program = new allPrograms();
```

Program 4 : Triangle

```
1 package tests;
2
3 import static org.junit.Assert.*;
4
5
6 public class P4_triangle_tests {
7
8     @Test
9     public void test1() {
10         int a = 2, b = 2, c = 2;
11         allPrograms program = new allPrograms();
12         int result = program.triangle(a, b, c);
13
14         assertEquals(result, 0);
15     }
16
17     @Test
18     public void test2() {
19         int a = 3, b = 3, c = 4;
20         allPrograms program = new allPrograms();
21         int result = program.triangle(a, b, c);
22
23         assertEquals(result, 1);
24     }
25
26     @Test
27     public void test3() {
28         int a = 6, b = 5, c = 4;
29         allPrograms program = new allPrograms();
30         int result = program.triangle(a, b, c);
31
32         assertEquals(result, 2);
33     }
34 }
```

tests.P4_triangle_tests (Runner:JUnit4) (0.000 s)

Finished after 0.078 seconds

Runs: 13/13 Errors: 0 Failures: 0

Program 5 : Prefix

```
1 package tests;
2
3 import static org.junit.Assert.*;
4
5
6 public class P5_prefix_tests {
7
8     @Test
9     public void test1() {
10         String str1 = "good", str2 = "good morning";
11         allPrograms program = new allPrograms();
12         boolean result = program.prefix(str1, str2);
13
14         assertEquals(result, true);
15     }
16
17     @Test
18     public void test2() {
19         String str1 = "a", str2 = "abc";
20         allPrograms program = new allPrograms();
21         boolean result = program.prefix(str1, str2);
22
23         assertEquals(result, true);
24     }
25
26     @Test
27     public void test3() {
28         String str1 = "", str2 = "good morning";
29         allPrograms program = new allPrograms();
30         boolean result = program.prefix(str1, str2);
31
32         assertEquals(result, false);
33     }
34 }
```

tests.P5_prefix_tests (Runner:JUnit4) (0.000 s)

Finished after 0.017 seconds

Runs: 12/12 Errors: 0 Failures: 0