

Task 4: Parameterized testing for Armstrong Number using Junit.

AIM: To perform parameterized testing for Armstrong number using Junit.

Procedure:

Steps:

1. Creating Java project.
 - Click on File and select New project
 - Enter project name as com.vogella.JUnit.Armstrong
 - Click on Next and then on Finish
2. Creating java class
 - Right click on com.vogella.JUnit.Armstrong and click on New .
 - Click on class and give the class name as Armstrong
 - Click on Finish
 - Type the following code
 -

```
package com.vogella.JUnit.Armstrong;
import java.lang.Math;
```

```
public class Armstrong {

    public int isArmstrong(int n)
    {
        int sum=0,k=n;
        int l=0;
        while(n>0)
        {
            n=n/10;
            l++;
        }
        n=k;
        int r;
        while(n>0)
        {
            r=n%10;
            sum+=(int)Math.pow(r, l);
            n=n/10;
        }
        if(sum==k)
            return 1;
        else
            return 0;
    }

}
```

3. Creating java Test Project
 - Right click on com.vogella.JUnit.Armstrong
 - Click on properties and select tab java build path

198W1A0537 Section: A 17CS4752B SOFTWARE TESTING METHODOLOGIES LAB Sheet:

- Click on source and click on Create New Folder.
- Give the folder name as Test and click on next.
- Click on Finish and then on OK

4. Create java test class

- Right click on com.vogella.JUnit.Armstrong and click on new
- Click on Junit test case.
- Change the name of folder “src” to test in source folder “Test”.
- Click on browse and select “Armstrong” class and click on Next.
- Click on Finish and then on OK.
- Add the following code

```
package com.vogella.JUnit.Armstrong;

import static org.junit.Assert.*;

import org.junit.Test;
import java.util.Arrays;
import java.util.Collection;
import org.junit.runner.RunWith;
import org.junit.runners.Parameterized;
import org.junit.runners.Parameterized.Parameters;
@RunWith(Parameterized.class)

public class ArmstrongTest {

    private int a;
    private int t;

    public ArmstrongTest(int number,int sample)
    {
        this.t=number;
        this.a=sample;
    }
    @Parameters
    public static Collection<Integer[]> Armstrongs()
    {
        return Arrays.asList(new Integer[][] { {1,153},{1, 370},{1, 371},{1,
407},{1, 1634} });
    }
    //198W1A0517
    @Test
    public void testArmstrong()
    {
        Armstrong ar=new Armstrong();
        assertEquals(t,ar.isArmstrong(a));
    }
    public void testIsArmstrong()
    {
        fail("Not yet Implemented");
    }
}
```

OUTPUT:

CASE 1: Pass

198W1A0537 Section: A 17CS4752B SOFTWARE TESTING METHODOLOGIES LAB Sheet:

The screenshot shows an IDE with the following components:

- Project Explorer:** Shows the package structure `com.vogella.junit.Armstrong.ArmstrongTest` with 5 test runs, all successful.
- JUnit Console:** Displays "Runs: 5/5", "Errors: 0", and "Failures: 0".
- Source Editor:** Contains the `ArmstrongTest` class with the following code:

```
1 package com.vogella.junit.Armstrong;
2
3 import static org.junit.Assert.*;
4
5 import org.junit.Test;
6 import java.util.Arrays;
7 import java.util.Collection;
8 import org.junit.runner.RunWith;
9 import org.junit.runners.Parameterized;
10 import org.junit.runners.Parameterized.Parameters;
11 @RunWith(Parameterized.class)
12
13 public class ArmstrongTest {
14
15     private int a;
16     private int t;
17
18     public ArmstrongTest(int number, int sample)
19     {
20         this.t=number;
21         this.a=sample;
22     }
23     @Parameters
24     public static Collection<Integer[]> Armstrongs()
25     {
26         return Arrays.asList(new Integer[][] { {1,153},{1, 370},{1, 371},{1, 407},{1, 1634} });
27     }
28     //198W1A0517
29     @Test
30     public void testArmstrong()
31     {
32         Armstrong ar=new Armstrong();
33         assertEquals(t,ar.isArmstrong(a));
34     }
35     public void testIsArmstrong()
36     {
37         fail("Not yet Implemented");
38     }
39 }
```

CASE 2:Fail

The screenshot shows the same IDE as before, but with a failed test run:

- JUnit Console:** Displays "Runs: 5/5", "Errors: 0", and "Failures: 1".
- Source Editor:** The code is identical to the previous screenshot.
- Failure Trace:** Shows the following error:

```
java.lang.AssertionError: expected:<0> but was:<1>
    at com.vogella.junit.Armstrong.ArmstrongTest.testArmstrong(ArmstrongTest.java:32)
```

RESULT: To perform parameterized testing for Armstrong number using Junit is successfully completed.

Marks: _____

Staff Signature : _____