

CS6474: Software Testing Laboratory (Spring 2023)

Bishwajit Prasad Gond 222CS3113

Master of Technology 222cs3113@nitrkl.ac.in

Department of Computer Science & Engineering NIT, Rourkela

April 9, 2023

Contents

Find	FindBugs		
1.1	Write a program to generate a Factorial of numbers (where stack length should be at 3 (max)). The numbers should be 5, 3, 8, and 15	2	
1.2	Write a program to generate Fibonacci numbers	2	
1.3	Write a program that performs sorting of a group of integer values using the quick sort technique	3	
1.4	Write a program that accepts elements of a matrix and displays its transpose	3	
1.5	Write a program to add two matrices and display the sum matrix	4	
1.6	Write a program to Print Prime Numbers from 1 to 100 using Scanner Class and For Loop	4	
1.7	Write a program to generate a palindrome of numbers	5	
1.8	Write a program to find out the sum of two arrays	5	
1.9	Write a program to check whether the number is even or odd	6	
1.10	Write a program for binary to hexadecimal conversion	6	

1 FindBugs

Findbugs scans for possible bugs in Java software. Each finding is reported as a warning, but not all of these warnings are necessarily defects, e.g. warnings referring to possible performance issues. The terms bug or bug pattern are used in a misleading way by Findbugs. A better way would be to talk just about warnings. In the following article, the term warning will be used. All warnings are classified in four ranks:

- (i) scariest,
- (ii) scary,
- (iii) troubling and
- (iv) of concern.

This is a hint to the developer about the possible impact/severity of the warnings. The current version reports 400 warnings in the nine categories:

Warnings List					
Category	Numbers	Samples			
Correctness	142	Illegal format string			
Bad practice	84	Confusing method names			
Dodgy code	71	Useless control flow			
Multithreaded Correctness	45	A thread was created using the default empty run			
		method			
Performance	27	Method concatenates strings using + in a loop			
Malicious Code Vulnerability	15	Finalizer should be protected, not public			
Security	11	Hardcoded constant database password			
Experimental	3	Method may fail to clean up stream or resource			
Internationalization	2	Consider using Locale parameterized version of			
		invoked method			

Installation of Eclipse Plug-In

The Eclipse plug-in work with Eclipse 3.x releases from 3.3. The plug-in runs under Java 1.5 or newer.

For Eclipse 4.2 (Juno) the next steps install the plug-in:

- 1. In Eclipse, click on Help Install New Software and press Add button.
 - Insert Name: Findbugs
 - Insert URL: http://Findbugs.cs.umd.edu/eclipse
 - press OK button
- 2. You should see Findbugs in the list. Select the entry and press Next button.
- 3. You should see the Install Details without errors and press Next button.
- 4. Select the "I accept the terms of the license agreement" option and click Finish button.
- 5. The plug-in is not digitally signed. Go ahead and install it anyway. (press OK button)
- 6. Click Yes to make Eclipse restart itself.

Test the below programs in FindBugs Tool

1.1 Write a program to generate a Factorial of numbers (where stack length should be at 3 (max)). The numbers should be 5, 3, 8, and 15.

```
eclipse-workspace - findBug/src/findBug/Factorial.java - Eclipse IDE
                                                                                                                                                                                        a
File Edit Source Refactor Navigate Search Project Run Window Help
 Q 🔡 🐉 🍁
                                                                                                                                                                      # Package Explorer 🗵 🗎 🥞 🖁 🔍 🗓 Factorial java 🗵 🖸 Oddor Even. java 🚨 Palindrome java 🚨 Prime java 🚨 Quicksort java 🚨 Sumofarray java 🚨 *Transpose java 🤻
                                 > > findBug > - src > - findBug > O Factorial > of fact(int): String

1 package findBug;
 > 📂 eclemma
 > MRE System Library [JavaSE-1.8]

> # src (7)
                                  5= public static String fact(int n) {
6    if(n==0)
7    {
8       return ""+1;

→ # findBua (7)

        > D BinarytoHex.java (1)
> D Factorial.java
                                               }
else if(n<0)
{
        > Pibonacci.java (1)
        > A MatrixAddition.iava (2)

    OddorEven.java
    Palindrome.java

                                                   return "For Negative number no factorial";
       > Prime.java (2)

    Quicksort.java
    Sumofarray.java

                                                     long fact[]=new long [n+1];
        > 🔑 Transpose.java (1)
                                                     fact[0]=1;
for(int i=1;i<=n;i++)
 > 醚 Jabuti
  ≓ jumble (1)

≓ Random
                                                     {
    fact[i]=fact[i-1]*i;
  return ""+fact[n];
  SpotBugs (2)
                                           }

    Problems 
    Javadoc 
    Declaration 
    Console 
    Julinit 
    Coverage 
    Bug Explorer 
    Bug Info

→ 

ifindBug (7)

                                                                      Writable Smart Insert 21:37:318
```

Figure 1: Findbug Screenshot

1.2 Write a program to generate Fibonacci numbers.

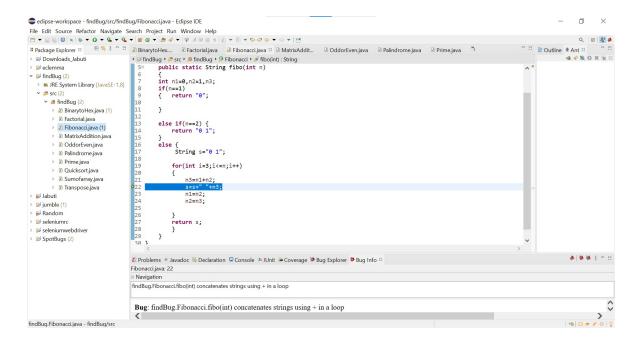


Figure 2: Findbug Screenshot

1.3 Write a program that performs sorting of a group of integer values using the quick sort technique.

```
eclipse-workspace - findBug/src/findBug/Quicksort.java - Eclipse IDE
                                                                                                                                                                                                    Ø
File Edit Source Refactor Navigate Search Project Run Window Help
 Q 🔡 🐉 🍁

    D BinarytoHex.java (1)
    Factorial.java
                                             public static int partition(int arr[], int start, int end) {
  int pivot = arr[end];
  int Index = start;
  for (int i = start; i < end; i++) {
     if (arr[i] < pivot) {
        swop(arr, i, Index);
        }
    }
}</pre>
         > 🔑 Fibonacci.java (1)
        MatrixAddition iava (2)

    OddorEven.java
    Palindrome.java

        > Prime.java (2)

    Quicksort.java
    Sumofarray.java

         > 🔎 Transpose.java (1)
                                                  swap(arr, Index, end);
return Index;
 > 醚 Jabuti
  ≓ jumble (1)

≓ Random
  public static int[] sort(int arr[], int start, int end) {
   if (start < end) {
     int Index = partition(arr, start, end);
     sort(arr, start, Index - 1);
     sort(arr, Index + 1, end);
}</pre>

    Problems 
    Javadoc 
    Declaration □ Console Ju JUnit □ Coverage 
    Bug Explorer 
    Bug Info

                                      Writable Smart Insert 1:12:11
```

Figure 3: Findbug Screenshot

1.4 Write a program that accepts elements of a matrix and displays its transpose.

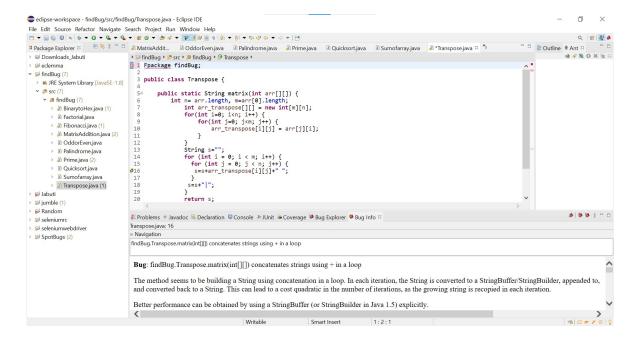


Figure 4: Findbug Screenshot

1.5 Write a program to add two matrices and display the sum matrix.

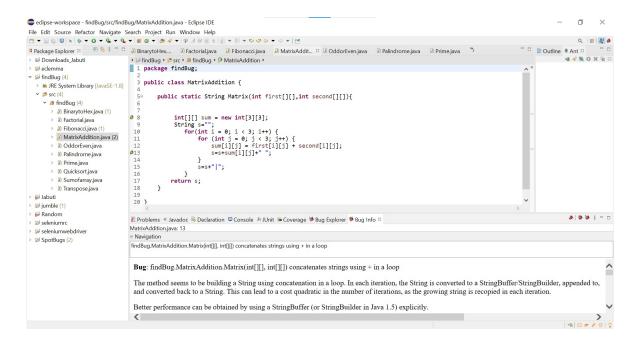


Figure 5: Findbug Screenshot

1.6 Write a program to Print Prime Numbers from 1 to 100 using Scanner Class and For Loop.

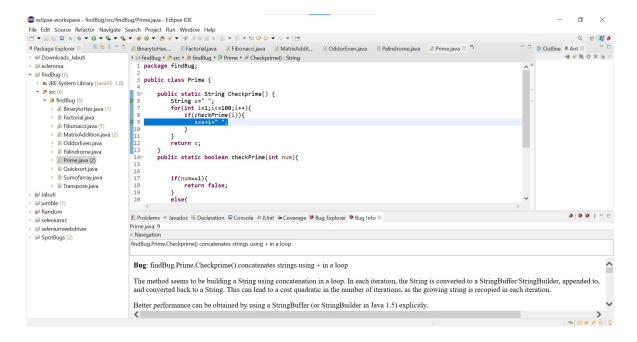


Figure 6: Findbug Screenshot

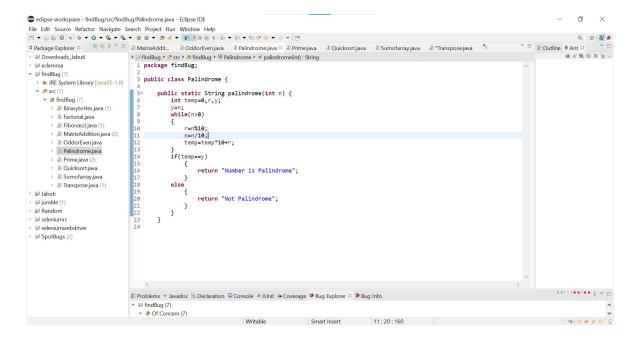


Figure 7: Findbug Screenshot

1.7 Write a program to generate a palindrome of numbers.

1.8 Write a program to find out the sum of two arrays.

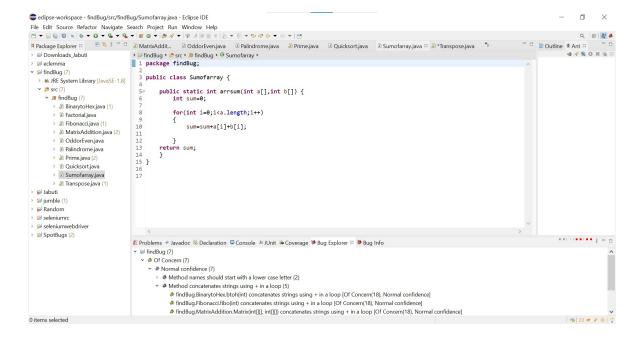


Figure 8: Findbug Screenshot

1.9 Write a program to check whether the number is even or odd.

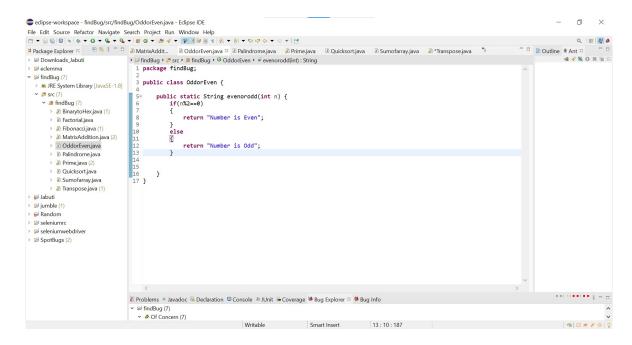


Figure 9: Findbug Screenshot

1.10 Write a program for binary to hexadecimal conversion.

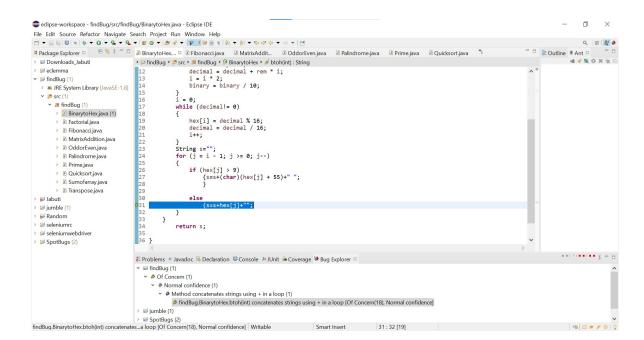


Figure 10: Findbug Screenshot