

CSA0979 Java programming

Assignment-1

T.Deekshitha

(192011256)

1) Write a program to find the number of composite numbers in an array of elements

Sample Input::

Array of elements = {16, 18, 27, 16, 23, 21, 19}

Sample Output:

Number of Composite Numbers = 5

Test cases:

1. Array of elements = {26, 28, 37, 26, 33, 31, 29}
2. Array of elements = {1.6, 1.8, 2.7, 1.6, 2.3, 2.1, .19}
3. Array of elements = {0, 160, 180, 270, 160, 230, 210, 190, 0}
4. Array of elements = {200, 180, 180, 270, 270, 270, 190, 200}
5. Array of elements = {100, 100, 100, 100, 100, 100, 100, 100}

OUTPUT:

```
import java.util.*;
```

```
public class composite_in_array {
```

```
public static void main(String[] args) {  
    try {  
        int n, comp = 0;  
        Scanner sc = new Scanner(System.in);  
        System.out.println("enter value of n:");  
        n = sc.nextInt();  
        int a[] = new int[100];  
        for (int i = 0; i < n; i++) {  
            a[i] = sc.nextInt();  
        }  
        for (int i = 0; i <= n; i++) {  
            int count = 0;  
            for (int j = 2; j < a[i]; j++) {  
                if (a[i] % j == 0) {  
                    count = count + 1;  
                }  
            }  
            if (count > 0) {  
                comp = comp + 1;  
            }  
        }  
        System.out.println("no o composite numbers= " + comp);  
    }  
    catch(Exception e)
```

```

    {
        System.out.println("invalid input.");
    }
}
}
}

```

OUTPUT:

The screenshot displays the Programiz Online Java Compiler interface. The browser tabs include WhatsApp, Word | Microsoft 365, Document 88.docx - Microsoft V..., Document 90.docx - Microsoft V..., and Online Java Compiler. The address bar shows the URL: programiz.com/java-programming/online-compiler/.

The interface features a header with the Programiz logo, a banner for "LOOKING TO LEARN PROGRAMMING?" with the text "Start your programming journey with Programiz AT NO COST.", and a button for "Interactive Java Course".

The main area is divided into two panels. The left panel, titled "Main.java", contains the following Java code:

```

1- public class CountComposites {
2-     public static boolean isComposite(int num) {
3-         if (num < 2) {
4-             return false;
5-         }
6-         for (int i = 2; i <= Math.sqrt(num); i++) {
7-             if (num % i == 0) {
8-                 return true;
9-             }
10-        }
11-        return false;
12-    }
13-
14-    public static int countComposites(int[] arr) {
15-        int count = 0;
16-        for (int num : arr) {
17-            if (isComposite(num)) {
18-                count++;
19-            }
20-        }
21-        return count;
22-    }
23-
24-    public static void main(String[] args) {
25-        int[] arr = {16, 18, 27, 16, 23, 21, 19};
26-        System.out.println("Number of Composite Numbers = " + countComposites(arr));
27-    }
28-}

```

The right panel, titled "Output", shows the result of running the code:

```

Java - cp /tmp/pamLx900r/ CountComposites
Number of Composite Numbers = 5

```

The bottom of the image shows a Windows taskbar with a search bar, several application icons, and system status information indicating 29°C, Mostly clear, and the date 04-04-2023.

```
Main.java
1  int n, comp = 0;
2  Scanner sc = new Scanner(System.in);
3  System.out.println("enter value of n:");
4  n = sc.nextInt();
5  int a[] = new int[100];
6  for (int i = 0; i < n; i++) {
7      a[i] = sc.nextInt();
8  }
9  for (int i = 0; i <= n; i++) {
10     int count = 0;
11     for (int j = 2; j < a[i]; j++) {
12         if (a[i] % j == 0) {
13             count = count + 1;
14         }
15     }
16     if (count > 0) {
17         comp = comp + 1;
18     }
19 }
20 System.out.println("no o composite numbers= " + comp);
21 catch(Exception e)
22 {
23     System.out.println("invalid input.");
24 }
25 }
```

```
Output
java -cp /tmp/tC3v85p4Jsb composite_in_array
enter value of n:
9
0
160
180
270
160
230
210
190
0
no o composite numbers= 7
```

```
Main.java
1  import java.util.*;
2  public class composite_in_array {
3      public static void main(String[] args) {
4          try {
5              int n, comp = 0;
6              Scanner sc = new Scanner(System.in);
7              System.out.println("enter value of n:");
8              n = sc.nextInt();
9              int a[] = new int[100];
10             for (int i = 0; i < n; i++) {
11                 a[i] = sc.nextInt();
12             }
13             for (int i = 0; i <= n; i++) {
14                 int count = 0;
15                 for (int j = 2; j < a[i]; j++) {
16                     if (a[i] % j == 0) {
17                         count = count + 1;
18                     }
19                 }
20                 if (count > 0) {
21                     comp = comp + 1;
22                 }
23             }
24             System.out.println("no o composite numbers= " + comp);
25         }
26         catch(Exception e)
27         {}
28     }
```

```
Output
java -cp /tmp/69HCSwMMTi composite_in_array
enter value of n:
7
1.6
invalid input.
```

```
Main.java
2  public static boolean isComposite(int num) {
3      if (num < 2) {
4          return false;
5      }
6      for (int i = 2; i <= Math.sqrt(num); i++) {
7          if (num % i == 0) {
8              return true;
9          }
10     }
11     return false;
12 }
13
14 public static int countComposites(int[] arr) {
15     int count = 0;
16     for (int num : arr) {
17         if (isComposite(num)) {
18             count++;
19         }
20     }
21     return count;
22 }
23
24 public static void main(String[] args) {
25     int[] arr = {26, 28, 37, 26, 33, 31, 29};
26     System.out.println("Number of Composite Numbers = " + countComposites(arr));
27 }
```

```
Output
java -cp /tmp/tBn2KEL575 CountComposites
Number of Composite Numbers = 4
```

Programiz Online Java Compiler

Calculate a Reverse Mortgage
SPONSORED BY LIBERTY REVERSE MORTGAGE

LEARN MORE

Interactive Java Course

Main.java

```

1  int n, comp = 0;
2  Scanner sc = new Scanner(System.in);
3  System.out.println("enter value of n:");
4  n = sc.nextInt();
5  int a[] = new int[100];
6  for (int i = 0; i < n; i++) {
7      a[i] = sc.nextInt();
8  }
9  for (int i = 0; i <= n; i++) {
10     int count = 0;
11     for (int j = 2; j < a[i]; j++) {
12         if (a[i] % j == 0) {
13             count = count + 1;
14         }
15     }
16     if (count > 0) {
17         comp = comp + 1;
18     }
19 }
20 System.out.println("no o composite numbers= " + comp);
21 catch(Exception e)
22 {
23     System.out.println("invalid input.");
24 }
25 }
26 }
27 }
28 }
29 }
30 }
31 }

```

Output

```

Java -cp /tmp/aCw85n4Jsb composite_in_array
enter value of n:
7
200
180
180
270
270
190
200
200
200
no o composite numbers= 7

```

Programiz Online Java Compiler

Interactive Java Course

Main.java

```

1  int n, comp = 0;
2  Scanner sc = new Scanner(System.in);
3  System.out.println("enter value of n:");
4  n = sc.nextInt();
5  int a[] = new int[100];
6  for (int i = 0; i < n; i++) {
7      a[i] = sc.nextInt();
8  }
9  for (int i = 0; i <= n; i++) {
10     int count = 0;
11     for (int j = 2; j < a[i]; j++) {
12         if (a[i] % j == 0) {
13             count = count + 1;
14         }
15     }
16     if (count > 0) {
17         comp = comp + 1;
18     }
19 }
20 System.out.println("no o composite numbers= " + comp);
21 catch(Exception e)
22 {
23     System.out.println("invalid input.");
24 }
25 }
26 }
27 }
28 }
29 }
30 }
31 }

```

Output

```

Java -cp /tmp/aCw85n4Jsb composite_in_array
enter value of n:
8
100
100
100
100
100
100
100
100
100
no o composite numbers= 6

```

2) Write a program for matrix addition?

Sample Input:

Mat1 = 1 2

5 3

Mat2 = 2 3

4 1

Sample Output:

Mat Sum = 3 5

9 4

```
import java.util.Scanner;
```

```
public class MatrixAddition {
```

```
    public static void main(String args[]) {
```

```
        Scanner scanner = new Scanner(System.in);
```

```
        System.out.print("Enter number of rows and columns of matrix: ");
```

```
        int rows = scanner.nextInt();
```

```
        int columns = scanner.nextInt();
```

```
        System.out.println("Enter first matrix:");
```

```
        int[][] matrix1 = new int[rows][columns];
```

```
        for (int i = 0; i < rows; i++) {
```

```
            for (int j = 0; j < columns; j++) {
```

```
                matrix1[i][j] = scanner.nextInt();
```

```
            }
```

```
        }
```

```
        System.out.println("Enter second matrix:");
```

```
        int[][] matrix2 = new int[rows][columns];
```

```
        for (int i = 0; i < rows; i++) {
```

```
            for (int j = 0; j < columns; j++) {
```

```
                matrix2[i][j] = scanner.nextInt();
```

```
            }
```

```
        }
```

```
        int[][] sum = new int[rows][columns];
```

```

for (int i = 0; i < rows; i++) {

    for (int j = 0; j < columns; j++) {

        sum[i][j] = matrix1[i][j] + matrix2[i][j];

    }

}

```

```

System.out.println("Matrix Sum:");

```

```

for (int i = 0; i < rows; i++) {

    for (int j = 0; j < columns; j++) {

        System.out.print(sum[i][j] + " ");

    }

    System.out.println();

}

}

}

```

OUTPUT:

The screenshot shows a web browser window with the URL `programiz.com/java-programming/online-compiler/`. The page title is "Programiz Online Java Compiler". There is a button labeled "Interactive Java Course".

The code editor shows the following Java code in `Main.java`:

```

1- import java.util.Scanner;
2-
3- public class MatrixAddition {
4-     public static void main(String args[]) {
5-         Scanner scanner = new Scanner(System.in);
6-
7-
8-         System.out.print("Enter number of rows and columns of matrix: ");
9-         int rows = scanner.nextInt();
10-        int columns = scanner.nextInt();
11-
12-
13-        System.out.println("Enter first matrix:");
14-        int[][] matrix1 = new int[rows][columns];
15-        for (int i = 0; i < rows; i++) {
16-            for (int j = 0; j < columns; j++) {
17-                matrix1[i][j] = scanner.nextInt();
18-            }
19-        }
20-
21-
22-        System.out.println("Enter second matrix:");
23-        int[][] matrix2 = new int[rows][columns];
24-        for (int i = 0; i < rows; i++) {
25-            for (int j = 0; j < columns; j++) {
26-                matrix2[i][j] = scanner.nextInt();
27-            }
28-        }
29-
30-        // Matrix Addition Logic
31-        int sum[][] = new int[rows][columns];
32-        for (int i = 0; i < rows; i++) {
33-            for (int j = 0; j < columns; j++) {
34-                sum[i][j] = matrix1[i][j] + matrix2[i][j];
35-            }
36-        }
37-
38-        System.out.println("Matrix Sum:");
39-        for (int i = 0; i < rows; i++) {
40-            for (int j = 0; j < columns; j++) {
41-                System.out.print(sum[i][j] + " ");
42-            }
43-            System.out.println();
44-        }
45-    }
46- }

```

The output window shows the following text:

```

java -cp /tmp/bZ38nNG6Gq MatrixAddition
Enter number of rows and columns of matrix: 2
2
Enter first matrix:
1 2
5 3
Enter second matrix:
3 5
9 4
Matrix Sum:4 7
14 7

```

The bottom of the image shows a Windows taskbar with the date and time as 23:22 on 04-04-2023.

3) OUTPUT:

The screenshot shows the Programiz Online Java Compiler interface. The code in `Main.java` is as follows:

```
1- import java.util.Scanner;
2-
3- public class SquareRoot {
4-     public static void main(String[] args) {
5-         Scanner scanner = new Scanner(System.in);
6-         System.out.print("Enter a positive integer: ");
7-         int n = scanner.nextInt();
8-         scanner.close();
9-
10-        int sqrt = (int) Math.sqrt(n);
11-
12-        if (sqrt * sqrt == n) {
13-            System.out.println("The square root of " + n + " is " + sqrt + ".");
14-            System.out.println("The square root of " + n + " is " + (-sqrt) + ".");
15-        } else {
16-            System.out.println(n + " is not a perfect square.");
17-        }
18-    }
19- }
```

The output window shows the following text:

```
Java -cp /tmp/bZ38nG6Gq SquareRoot
Enter a positive integer: 4
The square root of 4 is 2.
The square root of 4 is -2.
```

The screenshot shows the Programiz Online Java Compiler interface with a banner for "Bone-On-Bone Knees? See This!" at the top. The code in `Main.java` is identical to the previous screenshot:

```
1- import java.util.Scanner;
2-
3- public class SquareRoot {
4-     public static void main(String[] args) {
5-         Scanner scanner = new Scanner(System.in);
6-         System.out.print("Enter a positive integer: ");
7-         int n = scanner.nextInt();
8-         scanner.close();
9-
10-        int sqrt = (int) Math.sqrt(n);
11-
12-        if (sqrt * sqrt == n) {
13-            System.out.println("The square root of " + n + " is " + sqrt + ".");
14-            System.out.println("The square root of " + n + " is " + (-sqrt) + ".");
15-        } else {
16-            System.out.println(n + " is not a perfect square.");
17-        }
18-    }
19- }
```

The output window shows the following text:

```
Java -cp /tmp/gYT3wDAQ9o SquareRoot
Enter a positive integer: 8
8 is not a perfect square.
```


4)

The screenshot shows the Programiz Online Java Compiler interface. The code editor contains a Java program named `Main.java` that checks if a string is a palindrome. The program prompts the user to enter a string/number, reads the input, reverses it, and compares the two. The output window shows the execution results.

```
1 import java.util.*;
2 class Main{
3     public static void main(String args[])
4     {
5         String original, reverse = "";
6         Scanner in = new Scanner(System.in);
7         System.out.println("Enter a string/number to check if it is a palindrome");
8         original = in.nextLine();
9         int length = original.length();
10        for ( int i = length -1; i >= 0; i-- )
11            reverse = reverse + original.charAt(i);
12        if (original.equals(reverse))
13            System.out.println("True");
14        else
15            System.out.println("False");
16    }
17 }
```

Output:

```
java -cp /tmp/5InV4Bo0Q0 Main
Enter a string/number to check if it is a palindrome:
121
True
```

5)

The screenshot shows the Programiz Online Java Compiler interface. The code editor contains a Java program named `Age.java` that checks if a person is eligible to vote based on their age. The program prompts the user to enter their age, reads the input, and prints the eligibility status. The output window shows the execution results.

```
1 import java.util.*;
2
3 class Age {
4     public static void main(String[] args) {
5         Scanner scan = new Scanner(System.in);
6         System.out.println("Enter the age of person:");
7         int user_age = scan.nextInt();
8         System.out.println("The age of person is " + user_age);
9         if (user_age >= 18) {
10             System.out.println("You are eligible to Vote");
11         } else {
12             System.out.println("You are not eligible to vote and for you " + (18
13                 - user_age) + " years are left to be eligible");
14         }
15     }
16 }
```

Output:

```
java -cp /tmp/5InV4Bo0Q0 Age
Enter the age of person
20
The age of person is 20
You are eligible to Vote
```