HARINI.D.S
231901009

Week-4: Sample Digital Program

1. To print first repeated number

```java
import java.util.Scanner; class Repeat{
public static void main(String[] args){
    Scanner s=new Scanner(System.in);
    System.out.println("Enter number of elements in array:");
    int size=s.nextInt();
int[] array = new int[size];
    System.out.println("Enter " + size + " elements:");
    for (int i = 0; i < size; i++) {
array[i] = s.nextInt();
    }
    boolean foundRepeat = false;
    for (int i = 0; i < size; i++) {
for (int j = i + 1; j < size; j++) {
if (array[i] == array[j]) {
            System.out.println("First repeated number: " + array[i]);
foundRepeat = true;
            break;
        }
    }
    if (foundRepeat) {
break;
    }
    }
    if (!foundRepeat) {
    System.out.println("No repeated numbers found.");
    }
    s.close();
  }
}
```

Output:

HARINI.D.S
231901009

```
Enter the size of the array: 6

Enter 6 elements:

1

2

4

1

2

8

First repeated number: 1
```

2. Program to print sum of the prime numbers except the largest prime number if there is no prime numbers the largest number is neglected and the sum of the other number to be printed.

Program:
```java
import java.util.Scanner;

public class PrimeSumExcludingLargest {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter the size of the array: ");
      int size = scanner.nextInt();
      int[] array = new int[size];
      int largestPrime = 0, primeSum = 0, largestElement = Integer.MIN_VALUE;

      System.out.println("Enter " + size + " elements:");
      for (int i = 0; i < size; i++) {
        array[i] = scanner.nextInt();
        if (isPrime(array[i])) {
           primeSum += array[i];
           if (array[i] > largestPrime) {
              largestPrime = array[i];
           }
        }
        if (array[i] > largestElement) {
           largestElement = array[i];
        }
      }

      if (primeSum == 0) {
        primeSum = sumArray(array) - largestElement;
        System.out.println("No prime numbers found. Sum excluding the largest element: " +
primeSum);
      } else {
        primeSum -= largestPrime;
        System.out.println("Sum of primes excluding the largest prime: " + primeSum);
      }
```
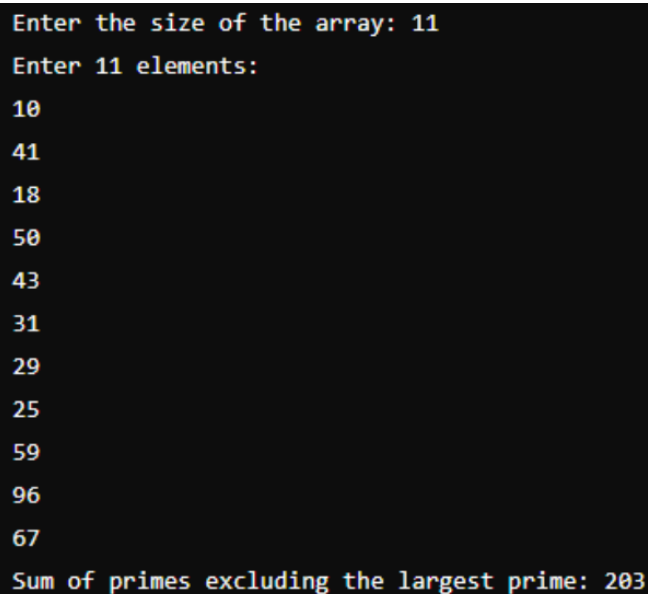
```
      scanner.close();
  }

  public static boolean isPrime(int num) {
     if (num <= 1) return false;
     for (int i = 2; i * i <= num; i++) {
        if (num % i == 0) return false;
     }
     return true;
  }

  public static int sumArray(int[] array) {
     int sum = 0;
     for (int num : array) {
        sum += num;
     }
     return sum;
  }
}
```

Output:

```
Enter the size of the array: 11
Enter 11 elements:
10
41
18
50
43
31
29
25
59
96
67
Sum of primes excluding the largest prime: 203
```

3. Given an array of n elements , the program have to find the sum of the values that are present in non prime indexes of the array.

Program:

import java.util.Scanner;

HARINI.D.S
231901009

```java
public class SumNonPrimeIndexes {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the size of the array: ");
        int size = scanner.nextInt();
        int[] array = new int[size];

        System.out.println("Enter " + size + " elements:");
        for (int i = 0; i < size; i++) {
            array[i] = scanner.nextInt();
        }

        int sum = 0;
        for (int i = 0; i < size; i++) {
            if (!isPrime(i)) {
                sum += array[i];
            }
        }

        System.out.println("Sum of values at non-prime indexes: " + sum);
        scanner.close();
    }

    public static boolean isPrime(int num) {
        if (num <= 1) return false;
        for (int i = 2; i * i <= num; i++) {
            if (num % i == 0) return false;
        }
        return true;
    }
}
```

Output:

```
Enter the size of the array: 10
Enter 10 elements:
10
20
30
40
50
60
70
80
90
100
Sum of values at non-prime indexes: 340
```

CSE(CYBER SECURITY)

HARINI.D.S

231901009

```
Enter the size of the array: 2
Enter 2 elements:
-2
-4
Sum of values at non-prime indexes: -6
```