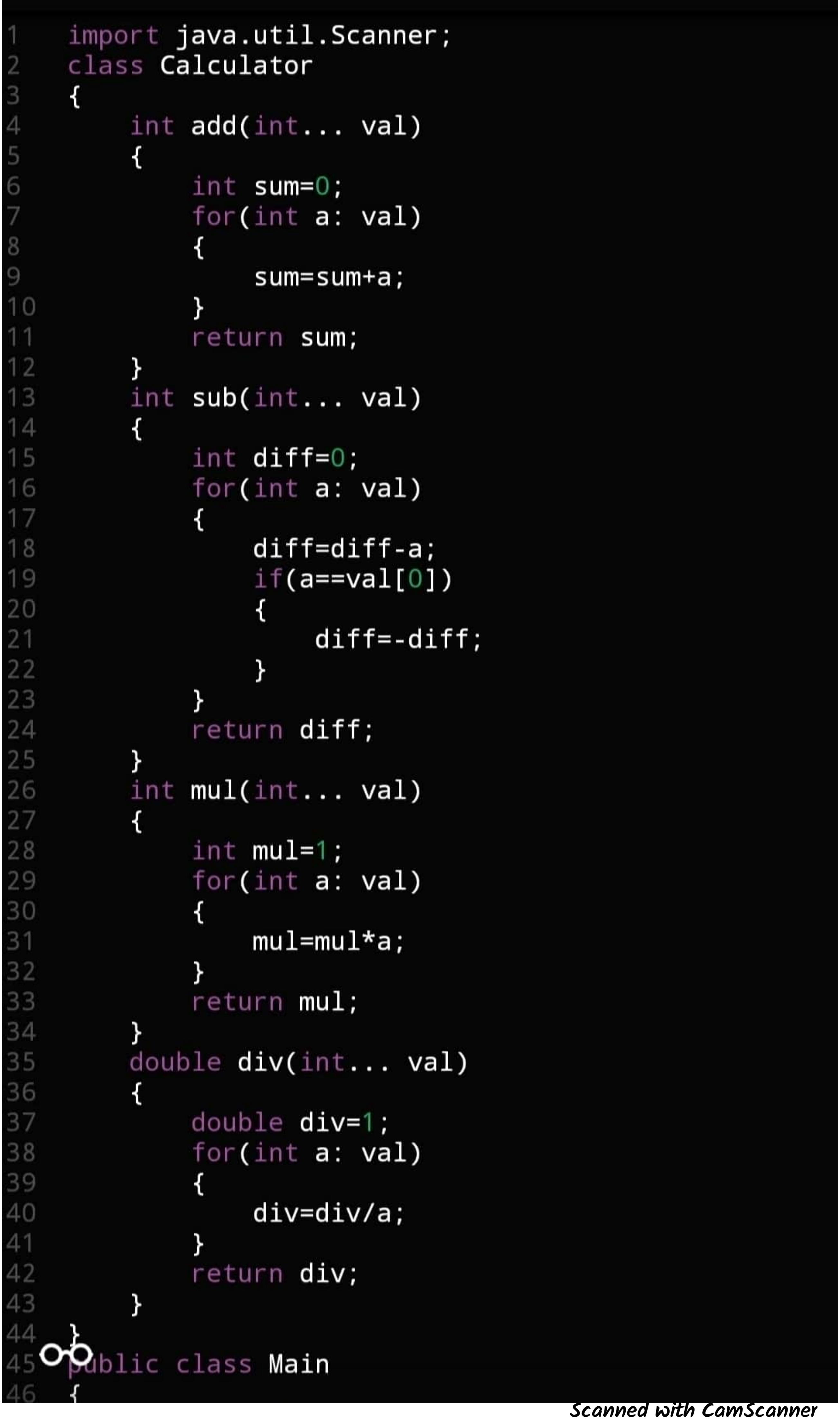
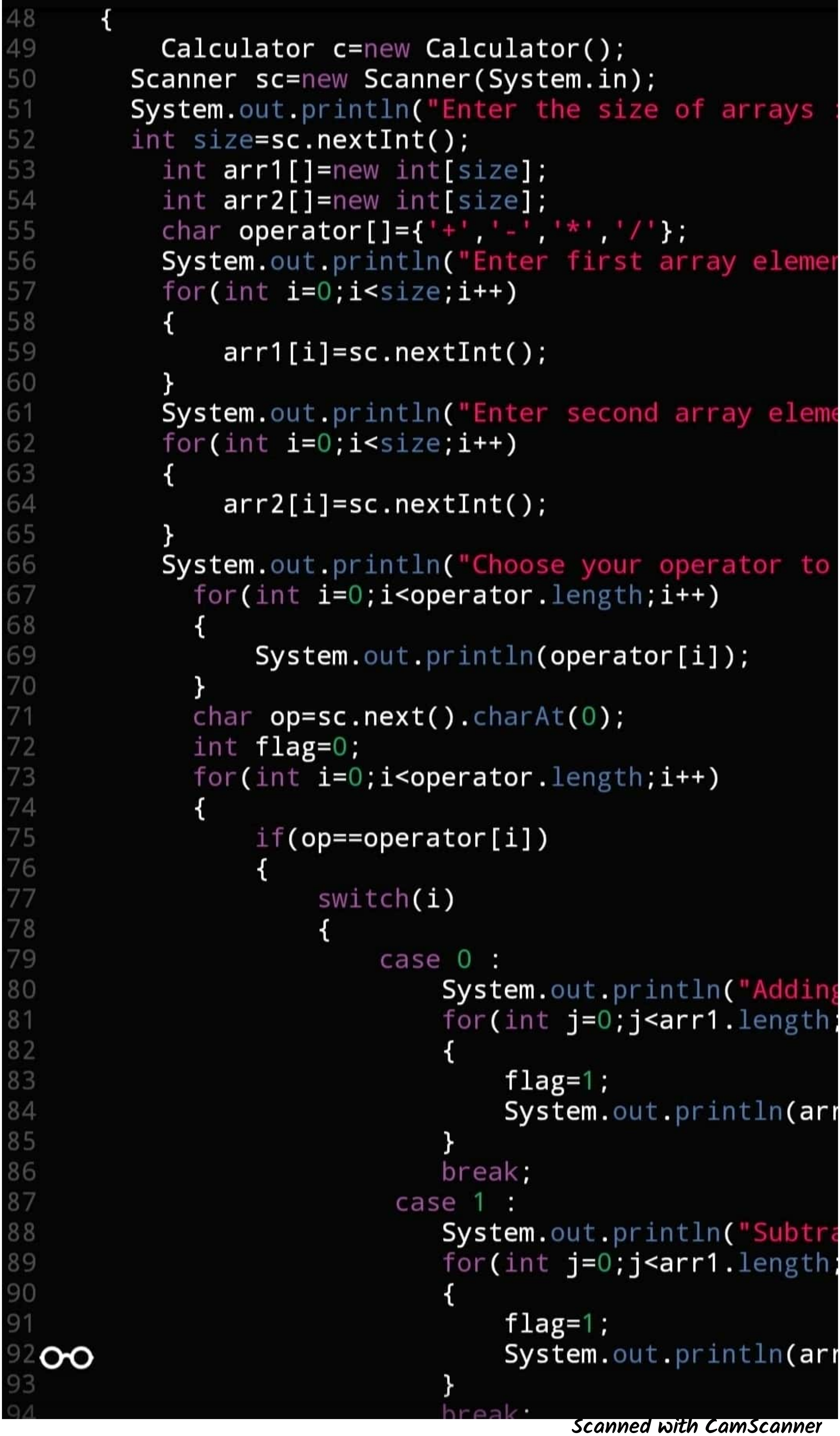
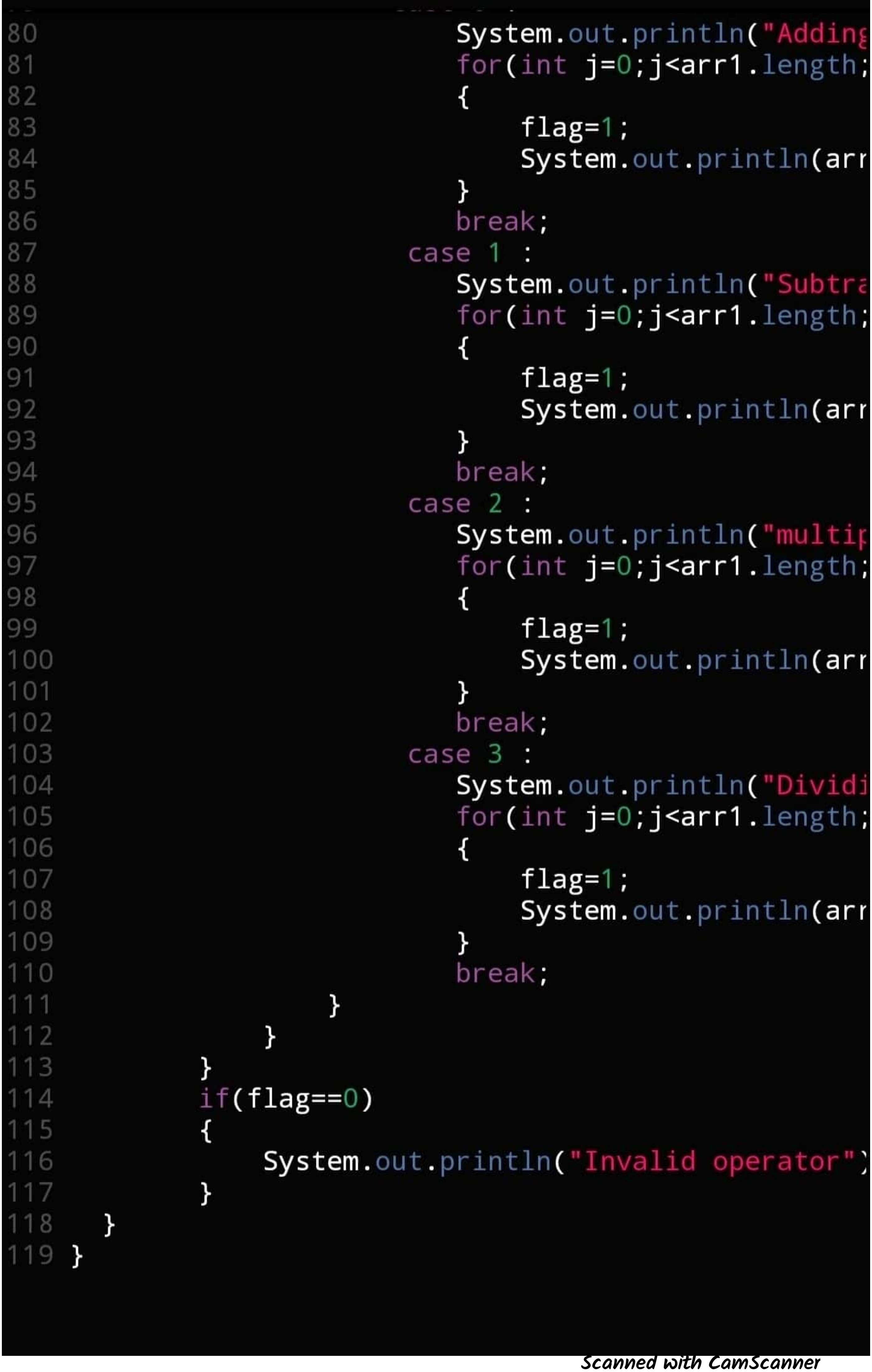


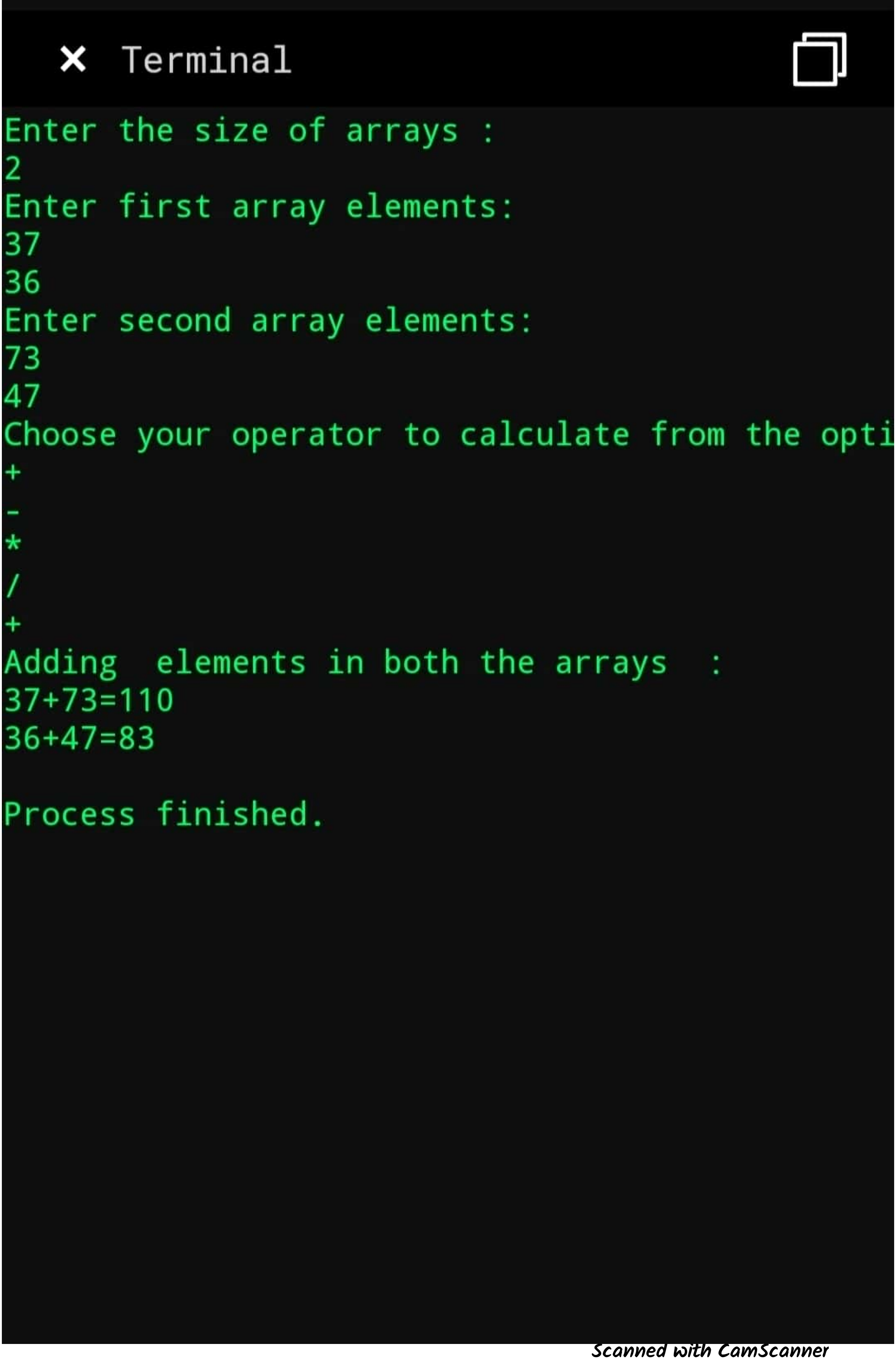
```java
import java.util.Scanner;
class Calculator
{
    int add(int... val)
    {
        int sum=0;
        for(int a: val)
        {
            sum=sum+a;
        }
        return sum;
    }
    int sub(int... val)
    {
        int diff=0;
        for(int a: val)
        {
            diff=diff-a;
            if(a==val[0])
            {
                diff=-diff;
            }
        }
        return diff;
    }
    int mul(int... val)
    {
        int mul=1;
        for(int a: val)
        {
            mul=mul*a;
        }
        return mul;
    }
    double div(int... val)
    {
        double div=1;
        for(int a: val)
        {
            div=div/a;
        }
        return div;
    }
}
public class Main
{
```



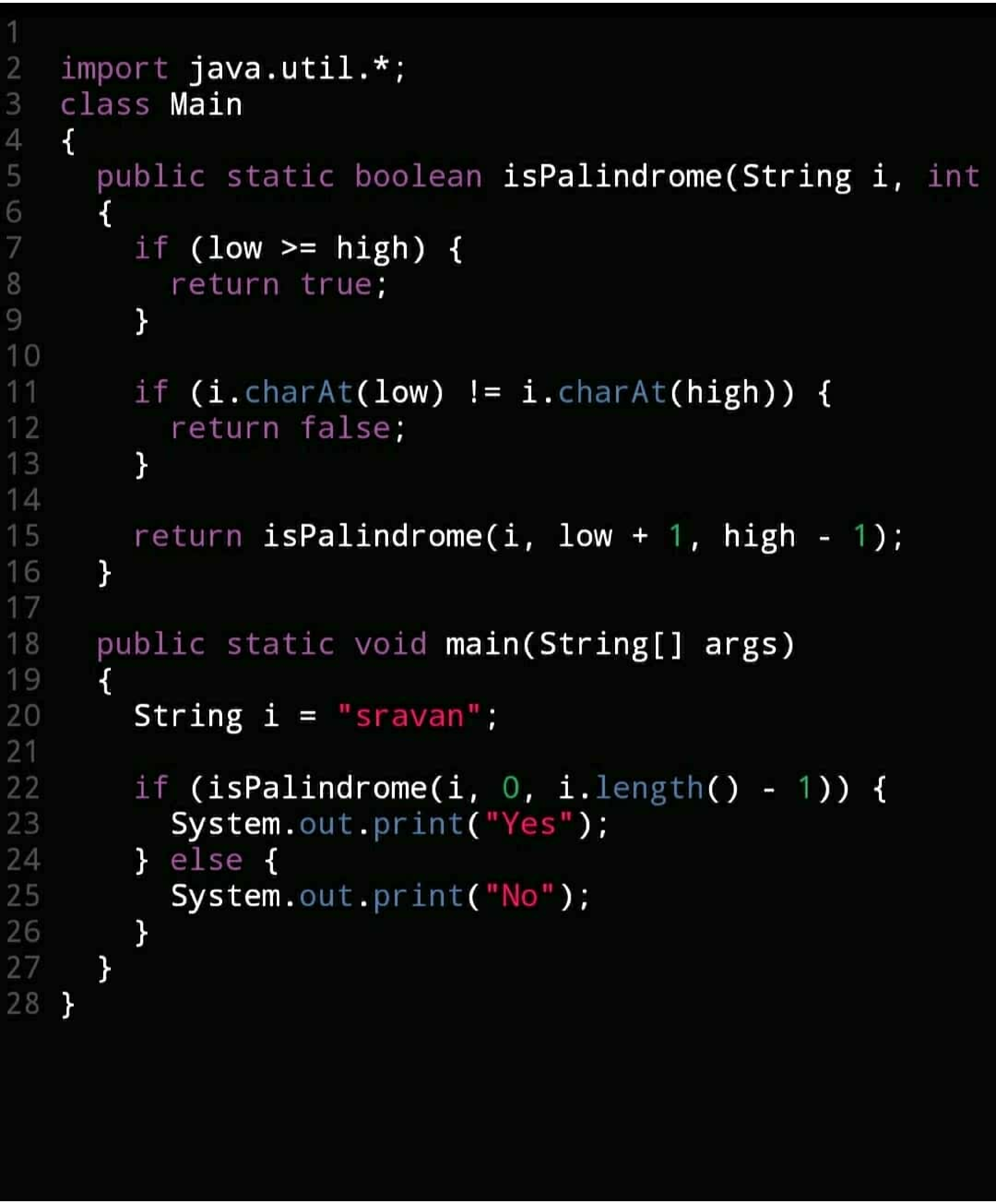
```
{
    Calculator c=new Calculator();
  Scanner sc=new Scanner(System.in);
  System.out.println("Enter the size of arrays
  int size=sc.nextInt();
    int arr1[]=new int[size];
    int arr2[]=new int[size];
    char operator[]={'+','-','*','/'};
    System.out.println("Enter first array elemen
    for(int i=0;i<size;i++)
    {
        arr1[i]=sc.nextInt();
    }
    System.out.println("Enter second array eleme
    for(int i=0;i<size;i++)
    {
        arr2[i]=sc.nextInt();
    }
    System.out.println("Choose your operator to
      for(int i=0;i<operator.length;i++)
      {
          System.out.println(operator[i]);
      }
      char op=sc.next().charAt(0);
      int flag=0;
      for(int i=0;i<operator.length;i++)
      {
          if(op==operator[i])
          {
              switch(i)
              {
                  case 0 :
                      System.out.println("Adding
                      for(int j=0;j<arr1.length
                      {
                          flag=1;
                          System.out.println(arr
                      }
                      break;
                   case 1 :
                      System.out.println("Subtra
                      for(int j=0;j<arr1.length
                      {
                          flag=1;
                          System.out.println(arr
                      }
                      break;
```



```
                        System.out.println("Adding
                        for(int j=0;j<arr1.length;
                        {
                            flag=1;
                            System.out.println(arr
                        }
                        break;
                     case 1 :
                        System.out.println("Subtra
                        for(int j=0;j<arr1.length;
                        {
                            flag=1;
                            System.out.println(arr
                        }
                        break;
                     case 2 :
                        System.out.println("multip
                        for(int j=0;j<arr1.length;
                        {
                            flag=1;
                            System.out.println(arr
                        }
                        break;
                     case 3 :
                        System.out.println("Dividi
                        for(int j=0;j<arr1.length;
                        {
                            flag=1;
                            System.out.println(arr
                        }
                        break;
                }
            }
        }
        if(flag==0)
        {
            System.out.println("Invalid operator")
        }
  }
}
```

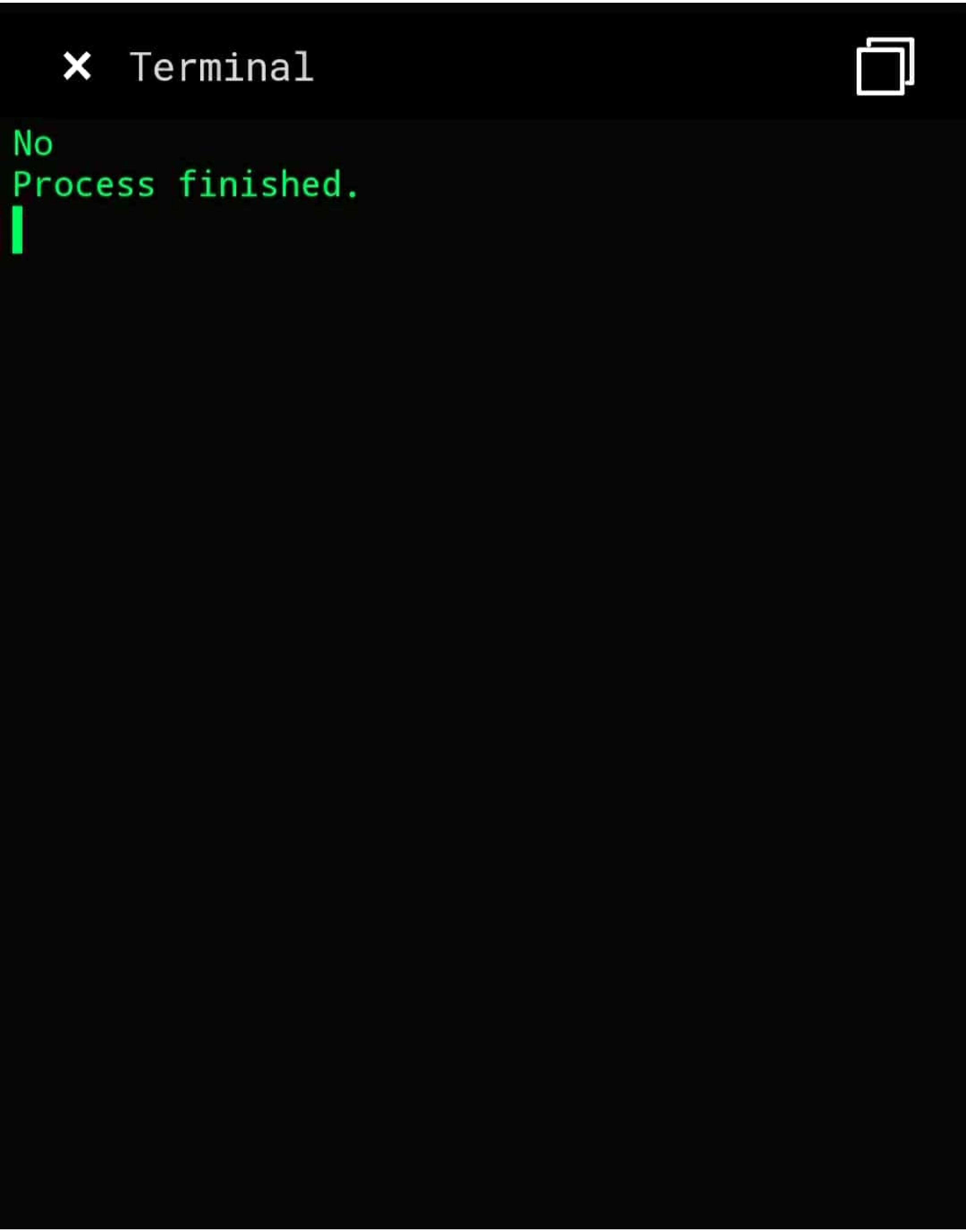


```
Enter the size of arrays :
2
Enter first array elements:
37
36
Enter second array elements:
73
47
Choose your operator to calculate from the opti
+
-
*
/
+
Adding  elements in both the arrays  :
37+73=110
36+47=83

Process finished.
```

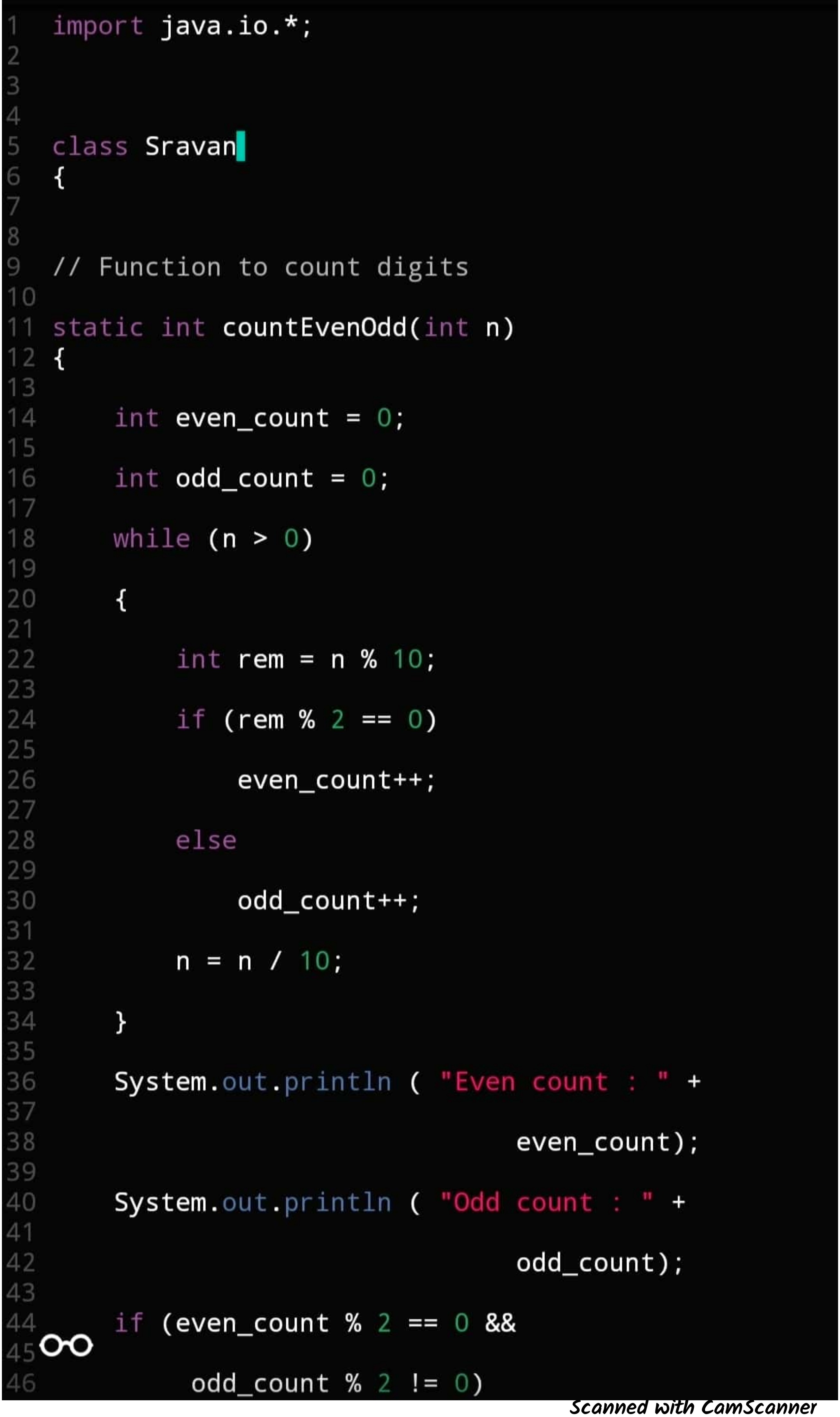


```java
import java.util.*;
class Main
{
  public static boolean isPalindrome(String i, int
  {
    if (low >= high) {
      return true;
    }

    if (i.charAt(low) != i.charAt(high)) {
      return false;
    }

    return isPalindrome(i, low + 1, high - 1);
  }

  public static void main(String[] args)
  {
    String i = "sravan";

    if (isPalindrome(i, 0, i.length() - 1)) {
      System.out.print("Yes");
    } else {
      System.out.print("No");
    }
  }
}
```

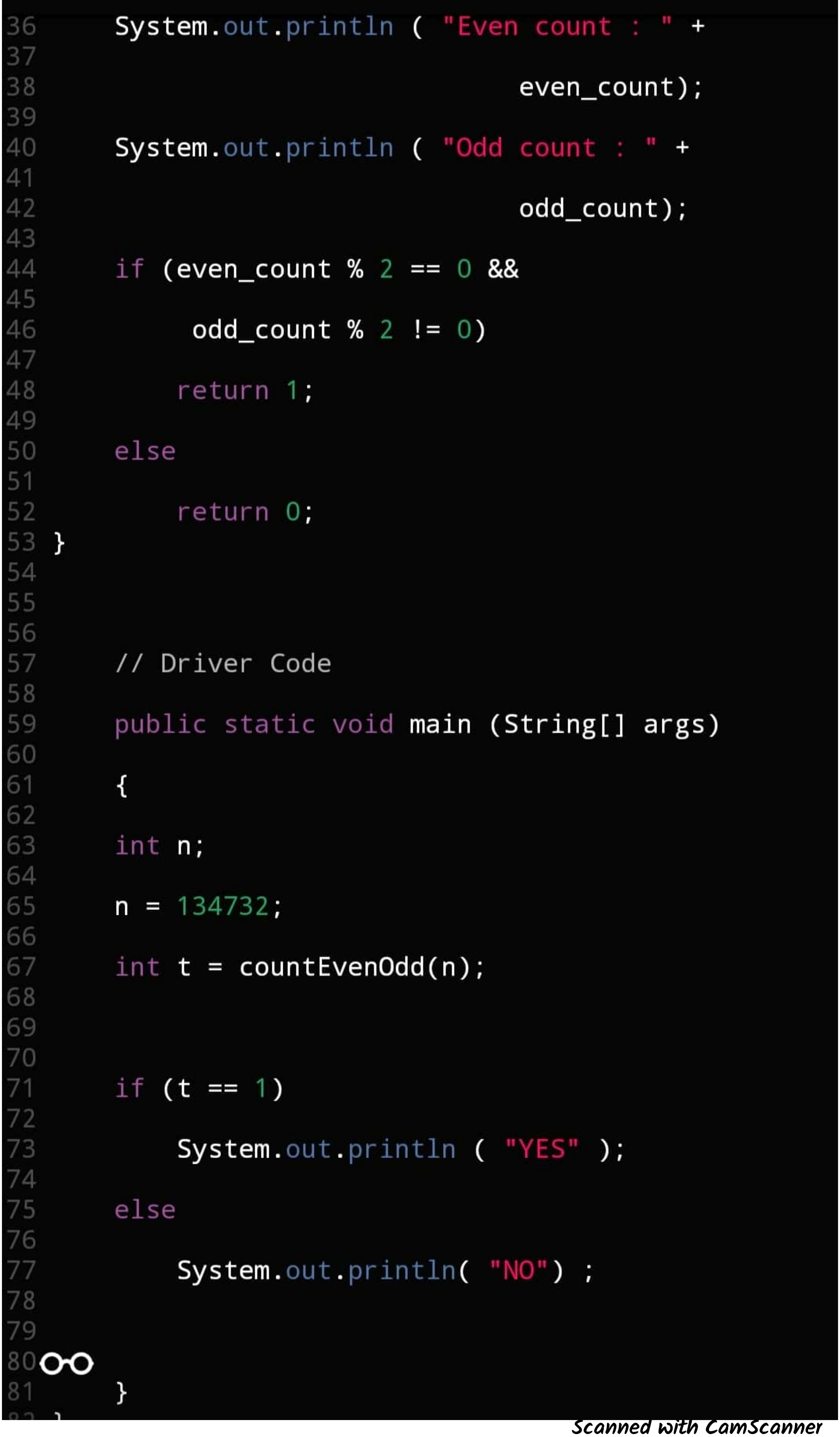
Scanned with CamScanner



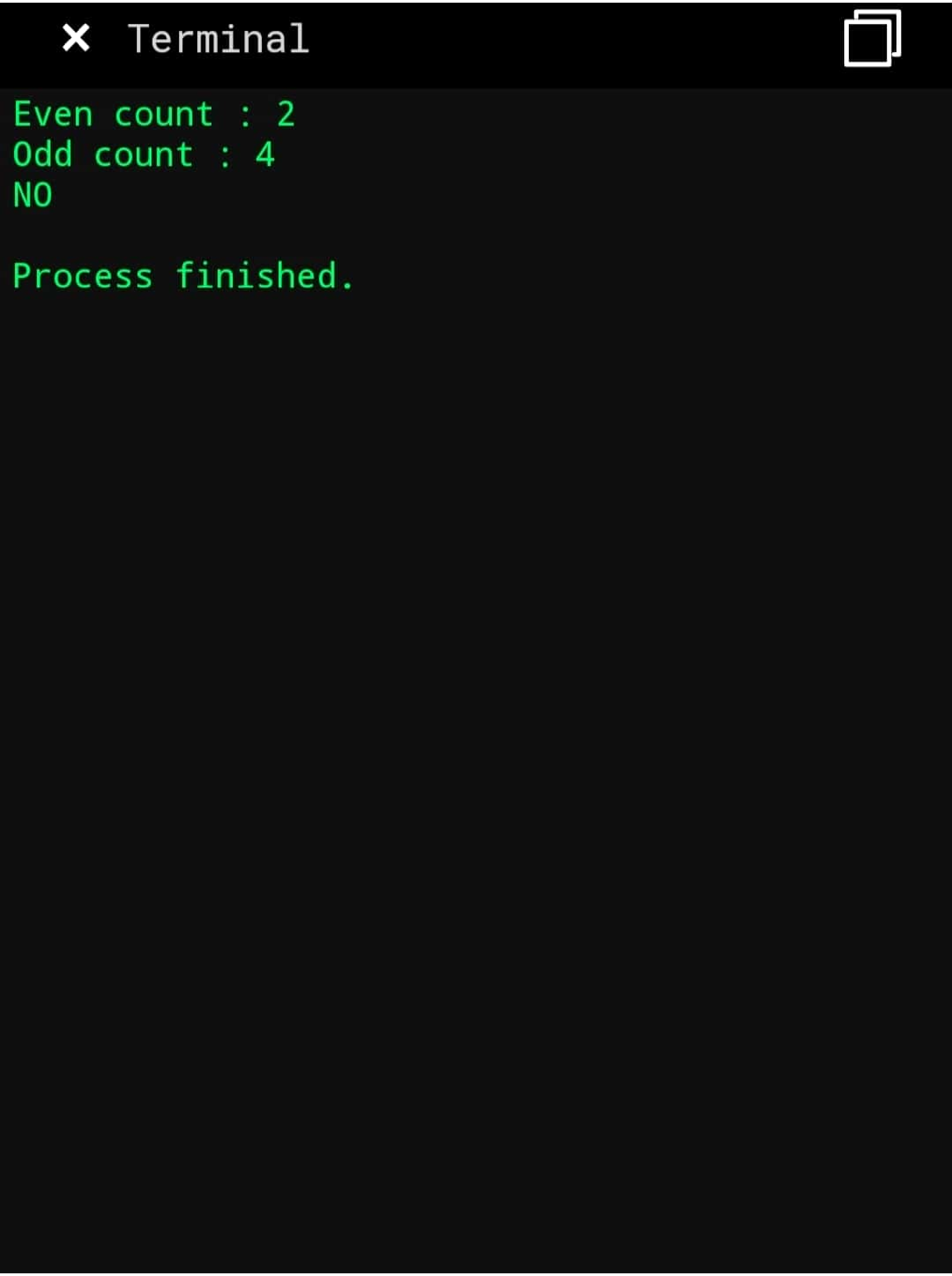
```
No
Process finished.
```



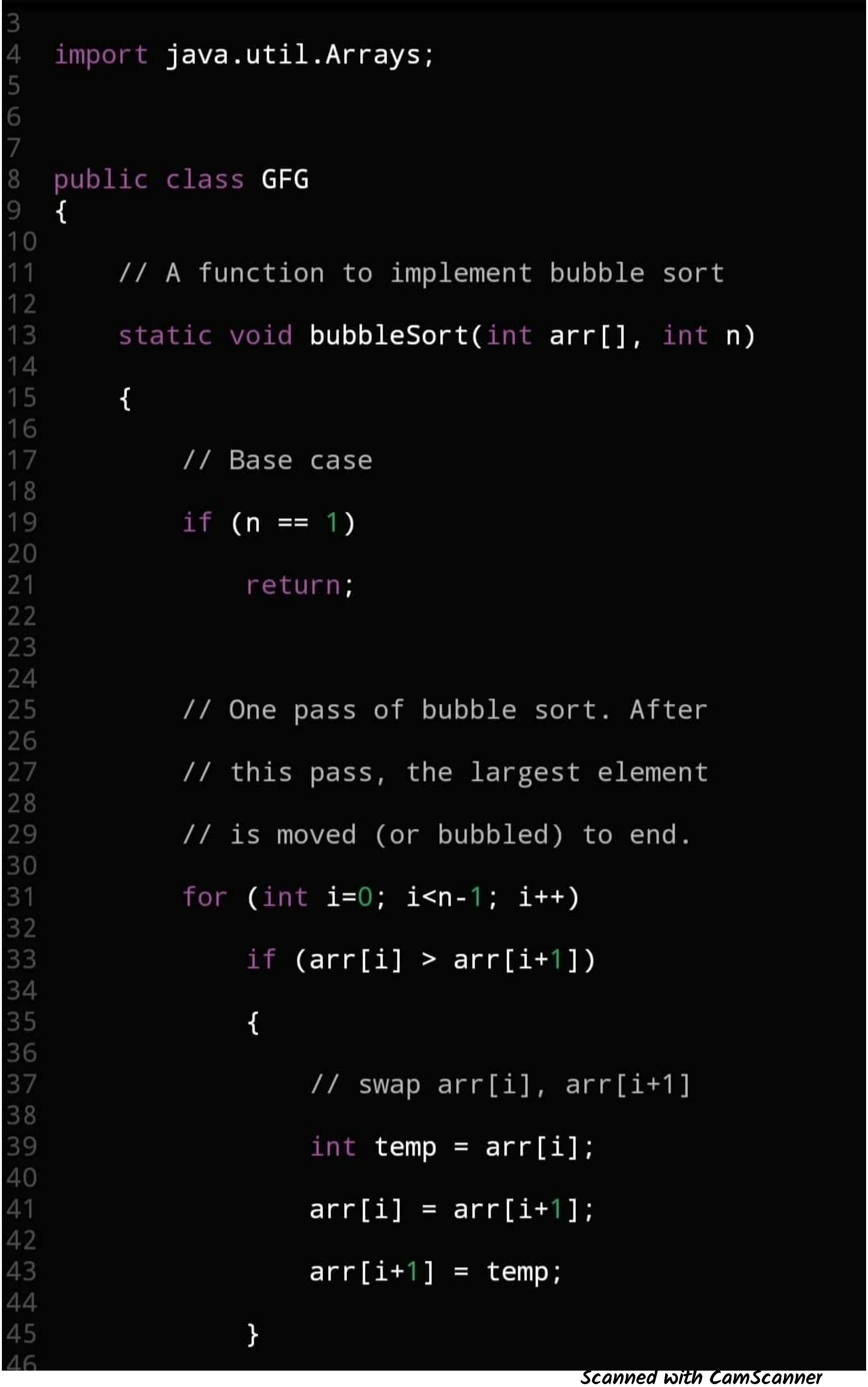
```java
import java.io.*;



class Sravan
{


// Function to count digits

static int countEvenOdd(int n)
{

    int even_count = 0;

    int odd_count = 0;

    while (n > 0)

    {

        int rem = n % 10;

        if (rem % 2 == 0)

            even_count++;

        else

            odd_count++;

        n = n / 10;

    }

    System.out.println ( "Even count : " +

                              even_count);

    System.out.println ( "Odd count : " +

                              odd_count);

    if (even_count % 2 == 0 &&

         odd_count % 2 != 0)
```



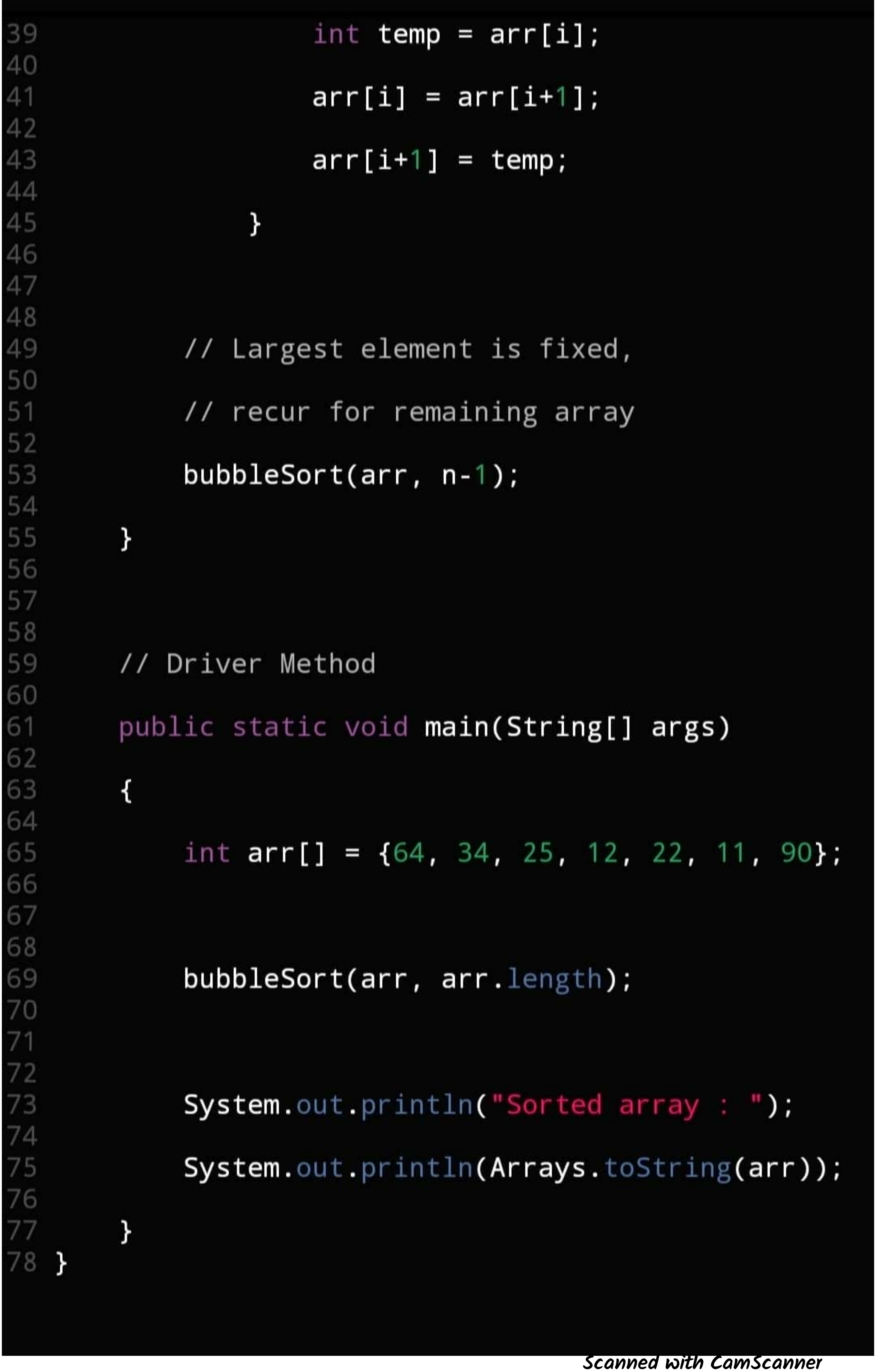
```
    System.out.println ( "Even count : " +

                              even_count);

    System.out.println ( "Odd count : " +

                              odd_count);

    if (even_count % 2 == 0 &&

         odd_count % 2 != 0)

        return 1;

    else

        return 0;
}



    // Driver Code

    public static void main (String[] args)

    {

    int n;

    n = 134732;

    int t = countEvenOdd(n);



    if (t == 1)

        System.out.println ( "YES" );

    else

        System.out.println( "NO") ;



    }
}
```



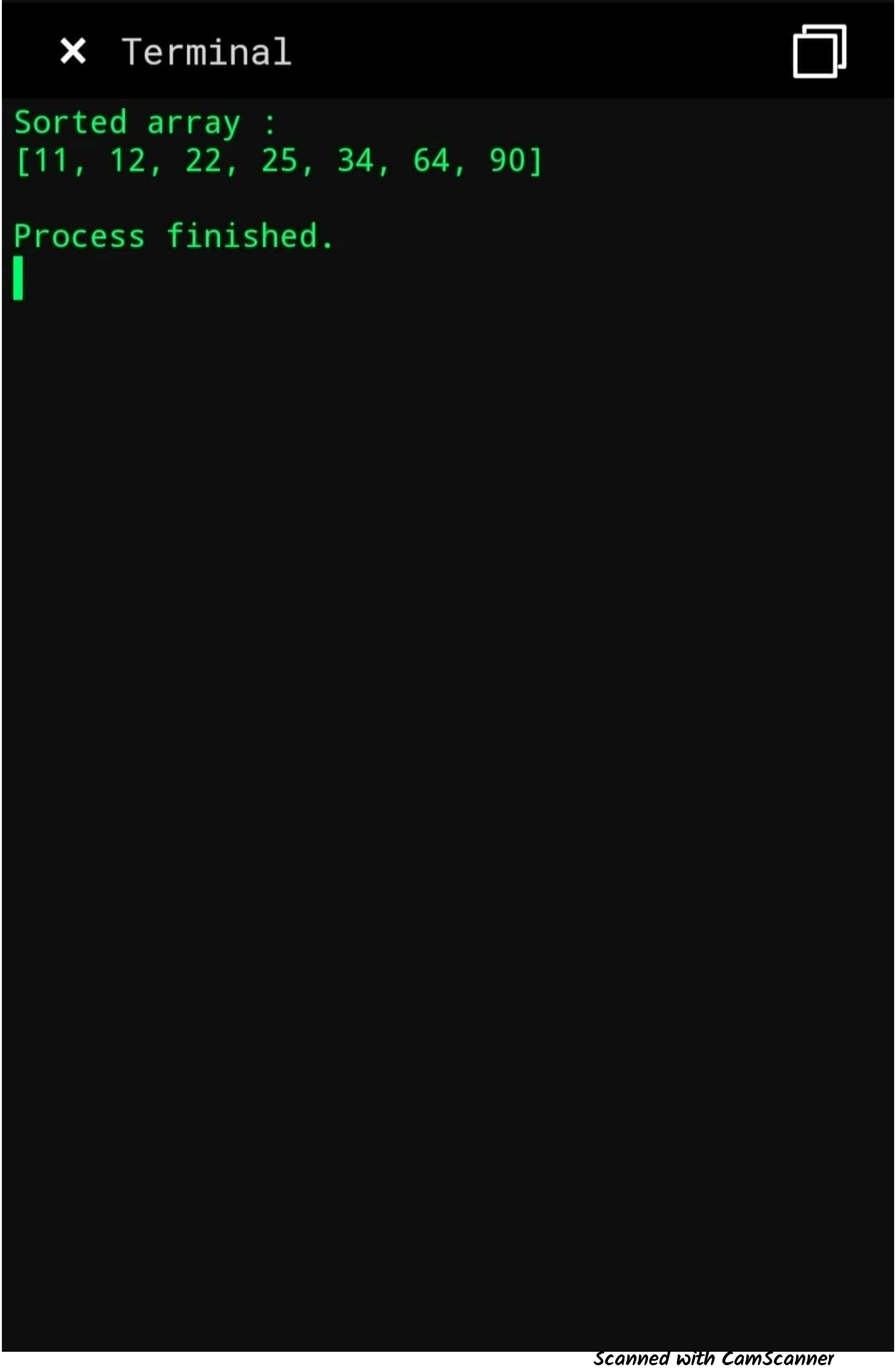
```
Even count : 2
Odd count : 4
NO

Process finished.
```



```

import java.util.Arrays;



public class GFG
{

    // A function to implement bubble sort

    static void bubbleSort(int arr[], int n)

    {

        // Base case

        if (n == 1)

            return;



        // One pass of bubble sort. After

        // this pass, the largest element

        // is moved (or bubbled) to end.

        for (int i=0; i<n-1; i++)

            if (arr[i] > arr[i+1])

            {

                // swap arr[i], arr[i+1]

                int temp = arr[i];

                arr[i] = arr[i+1];

                arr[i+1] = temp;

            }

```



```
                int temp = arr[i];

                arr[i] = arr[i+1];

                arr[i+1] = temp;

            }



        // Largest element is fixed,

        // recur for remaining array

        bubbleSort(arr, n-1);

    }



    // Driver Method

    public static void main(String[] args)

    {

        int arr[] = {64, 34, 25, 12, 22, 11, 90};



        bubbleSort(arr, arr.length);



        System.out.println("Sorted array : ");

        System.out.println(Arrays.toString(arr));

    }
}
```



```
Sorted array :
[11, 12, 22, 25, 34, 64, 90]

Process finished.
```