JAVA PROGRAMMING
ASSIGNMENT 2

Name:PUNEETH L
USN: 1BM24MC069

1. **Write a program called SumAverageRunningInt to produce the sum of 1, 2, 3, ..., to 100. Store 1 and 100 in variables lowerbound and upperbound, so that we can change their values easily. Also compute and display the average.**
   The output shall look like:
   The sum of 1 to 100 is 5050
   The average is 50.5

   Program:

   ```java
   import java.util.*;
   class SumAverageRunningInt {
       public static void main(String[] args) {
   Scanner sc = new Scanner(System.in);
   int start;
   int end;
   int sum;
   float avg;
   System.out.print("Enter the starting number:");
   start = sc.nextInt();
   System.out.print("Enter the ending number:");
   end = sc.nextInt();
     sum = (end - start + 1) * (start + end) / 2;
    System.out.println("The sum of " + start + " to " + end + "is: " + sum);
     avg = (float) sum / (end - start + 1);
     System.out.println("The average is: " + avg);
   }
   }
   ```

   OUTPUT:

```
D:\bmsce\2sem\Java programming\2ndweek>java SumAverageRunningInt
Enter the starting number:1
Enter the ending number:100
The sum of 1 to 100is: 5050
The average is: 50.5

D:\bmsce\2sem\Java programming\2ndweek>
```

2. **Write a program called HarmonicSum to compute the sum of a harmonic series, as shown below, where n=50000. The program shall compute the sum from left-to-right as well as from the right-to-left. Are the two sums the same? Obtain the absolute difference between these two sums and explain the difference. Which sum is more accurate?**

PROGRAM:

```java
public class harmonic {

public static void main(String[] args) {
 int n = 50000;
double leftToRight = 0.0;
double rightToLeft = 0.0;

for (int i = 1; i <= n; i++) {
       leftToRight += 1.0 / i;
}

for (int i = n; i >= 1; i--) {
       rightToLeft += 1.0 / i;
}

System.out.println("Left-to-right sum: " + leftToRight);
System.out.println("Right-to-left sum: " + rightToLeft);
System.out.println("Absolute difference: " + Math.abs(leftToRight - rightToLeft));
System.out.println("More accurate sum: Right-to-left (smaller terms added first)");
}
}
```

OUTPUT:

```
D:\bmsce\2sem\Java programming\2ndweek>java harmonic
Left-to-right sum: 11.397003949278504
Right-to-left sum: 11.397003949278519
Absolute difference: 1.4210854715202004E-14
More accurate sum: Right-to-left (smaller terms added first)

D:\bmsce\2sem\Java programming\2ndweek>
```

3. **Write a program called Fibonacci to print the first 20 Fibonacci numbers F(n), where F(n)=F(n–1) +F(n–2) and F (1) =F (2) =1. Also compute their average.**
   The output shall look like:
   The first 20 Fibonacci numbers are:
   1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987 1597 2584 4181 6765
   The average is 885.5

PROGRAM:

```
class fibonacci {

public static void main(String[] args) {
float avg;
int n = 20;
int i = 1;
int n1 = 1;
int n2 = 1;
int n3;
int sum = n1 + n2;
System.out.print("The first " + n + " Fibonacci numbers are: \n 1 1 ");
while (i <= n - 2) {
        n3 = n1 + n2;
        sum = sum + n3;
        n1 = n2;
        n2 = n3;
        i++;
        System.out.print(n3 + " ");
}
avg = (float) sum / n;

System.out.println("\nThe average is:" + avg);
        }
}
```

OUTPUT:


```
D:\bmsce\2sem\Java programming\2ndweek>java fibonacci
The first 20 Fibonacci numbers are:
 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987 1597 2584 4181 6765
The average is:885.5

D:\bmsce\2sem\Java programming\2ndweek>
```

4. **Write a program called ExtractDigits to extract each digit from an int, in the reverse order.**
   For example, if the int is 15423, the output shall be "3 2 4 5 1", with a space separating the digits.

   PROGRAM:

   ```java
   import java.util.*;

   class extractdigit {

       public static void main(String[] args) {
           int r;
           Scanner sc = new Scanner(System.in);
           System.out.println("Enter a number consisting of multiple digits:");
           int n = sc.nextInt();
           while (n != 0) {
               r = n % 10;
               System.out.print(r + " ");
               n = n / 10;
           }
       }
   }
   ```

   OUTPUT:

   ```
   D:\bmsce\2sem\Java programming\2ndweek>java extractdigit
   Enter a number consisting of multiple digits:
   784562
   2 6 5 4 8 7
   D:\bmsce\2sem\Java programming\2ndweek>
   ```