Date: 6th April 2021

Online Lab Tasks

- Submitted to: Azfar Shakeel Khan
- Submitted by: FA20- BSE-094
- Name: M.Ruslan Babar

Question 6.1

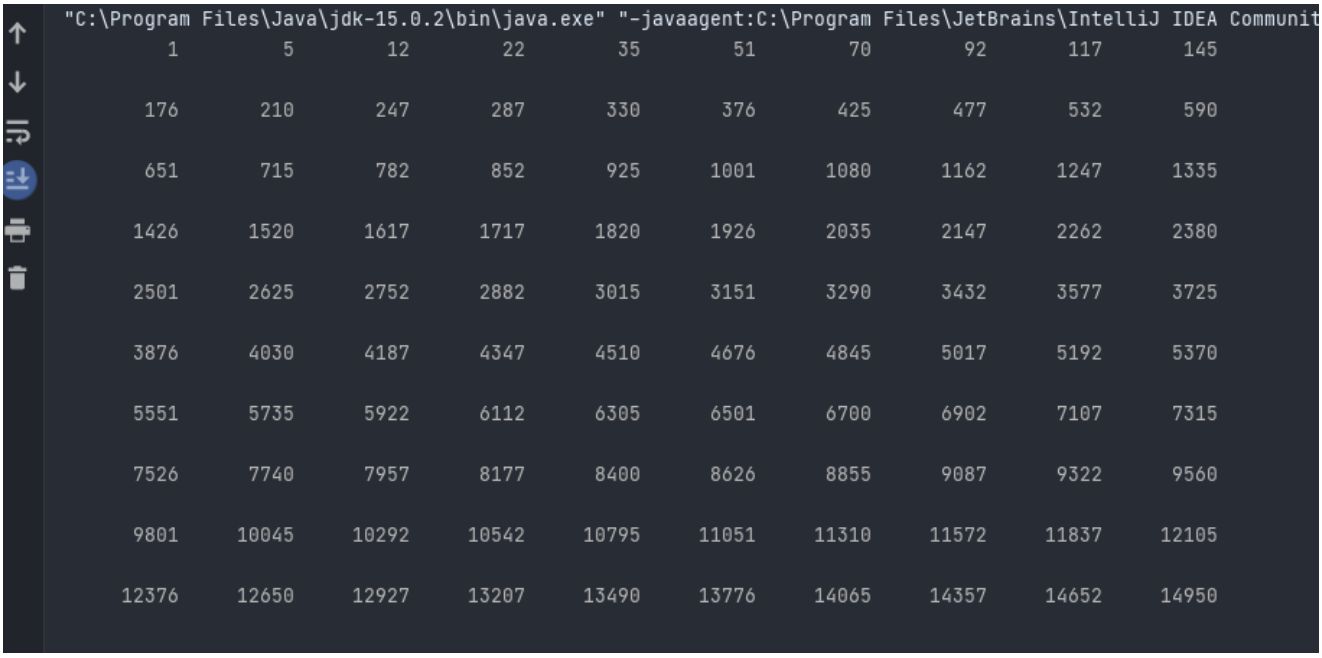
6.1 (*Math: pentagonal numbers*) A pentagonal number is defined as $n(3n-1)/2$ for $n = 1, 2, \ldots$, and so on. Therefore, the first few numbers are 1, 5, 12, 22, ... Write a method with the following header that returns a pentagonal number:

```
public static int getPentagonalNumber(int n)
```

Write a test program that uses this method to display the first 100 pentagonal numbers with 10 numbers on each line.

```java
package russi;
import java.util.*;
public class PentagonalNumbers {
    public static void main(String[] args) {
        //Per line Constant
        final int PER_LINE = 10;
        //Loop
        int i = 1;
        while(i <= 100){
            //Caller function
            int pentagonalNumber = getPentagonalNumber(i);
            //Displaying returned Value
            System.out.printf("%10d",pentagonalNumber);
            //Per line Changer
            if (i % 10 == 0)
                System.out.println("\n");
            //Counter Controller
            i++;
        }
    }
    //Method for Pentagon Value Return
    public static int getPentagonalNumber(int n){
        //Results variable for return value to main method
        int results = 0;
        //Formulae
        results = n * ( 3 * n - 1 ) / 2;
        //return
        return results;
    }
}
```

Output: 6.1



```
"C:\Program Files\Java\jdk-15.0.2\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Communit
        1         5        12        22        35        51        70        92       117       145

      176       210       247       287       330       376       425       477       532       590

      651       715       782       852       925      1001      1080      1162      1247      1335

     1426      1520      1617      1717      1820      1926      2035      2147      2262      2380

     2501      2625      2752      2882      3015      3151      3290      3432      3577      3725

     3876      4030      4187      4347      4510      4676      4845      5017      5192      5370

     5551      5735      5922      6112      6305      6501      6700      6902      7107      7315

     7526      7740      7957      8177      8400      8626      8855      9087      9322      9560

     9801     10045     10292     10542     10795     11051     11310     11572     11837     12105

    12376     12650     12927     13207     13490     13776     14065     14357     14652     14950
```

Question 6.2

***6.2** (*Sum the digits in an integer*) Write a method that computes the sum of the digits in an integer. Use the following method header:

```
public static int sumDigits(long n)
```

For example, `sumDigits(234)` returns `9` (2 + 3 + 4). (*Hint*: Use the `%` operator to extract digits, and the `/` operator to remove the extracted digit. For instance, to extract 4 from 234, use `234 % 10` (= 4). To remove 4 from 234, use `234 / 10` (= 23). Use a loop to repeatedly extract and remove the digit until all the digits are extracted. Write a test program that prompts the user to enter an integer and displays the sum of all its digits.

```java
package russi;
import java.util.*;
public class SumOf_Digits_In_A_Number {
    public static void main(String[] args) {
        //Scanner Object
        Scanner input = new Scanner (System.in);
        //Fetching User Input
        System.out.print("Enter a number: ");
        long number = input.nextInt();
        //Method Calling
        System.out.println("The sum of Digits is: "+sumDigits(number));
    }
    // SumDigits Method
    public static int sumDigits(long n){
        //Results
        int results = 0;
        //String for number length finding
        String string = ""+n;
        //Summing the digits in for loop
        for (int i = 0; i < string.length(); i++) {
            //Formula for digits separation
            results += n % 10;
            n /= 10;
        }
        //Return
        return results;

    }
}
```

Output: 6.2

```
"C:\Program Files\Java\jdk-15.0.2\bin\java.
Enter a number: 21341
The sum of Digits is: 11

Process finished with exit code 0
```

Question 6.3

**6.3 (*Palindrome integer*) Write the methods with the following headers

```
// Return the reversal of an integer, i.e., reverse(456) returns 654
public static int reverse(int number)

// Return true if number is a palindrome
public static boolean isPalindrome(int number)
```

Use the `reverse` method to implement `isPalindrome`. A number is a palindrome if its reversal is the same as itself. Write a test program that prompts the user to enter an integer and reports whether the integer is a palindrome.

```
1  package russi;
2  import java.util.*;
3  public class Palendrome_Integer {
4      public static void main(String[] args) {
5          //Scanner Object
6          Scanner input = new Scanner (System.in);
7          //User Input
8          System.out.print("Enter a number to check Palindrome: ");
9          int number = input.nextInt();
10         boolean isPal = isPalindrome(number); //Caller Function for palindrome
           validity
11         if(isPal) //Verifying results obtained from method returns
12             System.out.println("The number: "+number+" is Palindrome");
13         else
14             System.out.println("The number: "+number+" is not Palindrome");
15     }
16     public static boolean isPalindrome(int number){// Palindrome validator
17         if (number == reverse(number)) {
18             return true;
19         }
20         return  false;
21     }
22     // Reverse method that reverses an integer
23     public static int reverse(int number){
24         String stringNum = "" + number;
25         String reverseNumber = "";
26         for (int i = 0; i < stringNum.length(); i++)
27             reverseNumber = stringNum.charAt(i) + reverseNumber;
28         return (Integer.parseInt(reverseNumber));
29     }
30 }
```

Output: 6.3

```
"C:\Program Files\Java\jdk-15.0.2\bin\java
Enter a number to check Palindrome: 1234
The number: 1234 is not Palindrome

Process finished with exit code 0
```

```
"C:\Program Files\Java\jdk-15.0.2\bin\java.
Enter a number to check Palindrome: 2332
The number: 2332 is Palindrome

Process finished with exit code 0
```

Question 6.4

***6.4** (*Display an integer reversed*) Write a method with the following header to display an integer in reverse order:

```
public static void reverse(int number)
```

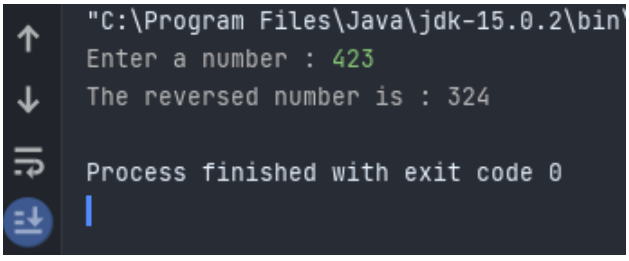
For example, `reverse(3456)` displays `6543`. Write a test program that prompts the user to enter an integer and displays its reversal.

```java
// package russi;
import java.util.Scanner;
public class IntegerReversed {
    public static void main(String[] args) {
        //Scanner Object
        Scanner input = new Scanner (System.in);
        //User Input
        System.out.print("Enter a number : ");
        int number = input.nextInt();
        //Caller Function for reversing
        int reverseD = reverse(number);
        System.out.println("The reversed number is : "+reverseD);

    }
    public static int reverse(int number){
        String stringNum = "" + number;
        String reverseNumber = "";
        for (int i = 0; i < stringNum.length(); i++)
            reverseNumber = stringNum.charAt(i) + reverseNumber;

        int results = 0;
        results = Integer.parseInt(reverseNumber);
        return results;
    }
}
```

Output: 6.4



Question 6.5

***6.5** (*Sort three numbers*) Write a method with the following header to display three numbers in increasing order:

```
public static void displaySortedNumbers(
  double num1, double num2, double num3)
```

```
1  import java.util.*;
2  public class SortThreeNumbers {
3      public static void main(String[] args) {
4          Scanner input = new Scanner(System.in);
5          System.out.print("Enter numbers with spaces: ");
6          double a = input.nextDouble(),b = input.nextDouble(),c = input.nextDouble();
7          displaySortedNumbers(a, b, c);
8      }
9      public static void displaySortedNumbers(double num1, double num2, double num3){
10         if( num1 < num2 && num1 < num3){
11             System.out.print(num1 + " ");
12             if (num2 < num3)
13                 System.out.print(num2 + " " + num3);
14             else
15                 System.out.print(num3 + " " + num2);
16         }
17         else if( num2 < num1 && num2< num3){
18             System.out.print(num2+" ");
19             if (num1 < num3)
20                 System.out.print(num1 + " " + num3);
21             else
22                 System.out.print(num3 + " " + num1);
23         }
24         else if ( num3 < num2 && num3 < num1){
25             System.out.print(num3 + " ");
26             if (num2 < num1)
27                 System.out.print(num2 + " " + num1);
28             else
29                 System.out.print(num1 + " " + num2);
30         }
31     }}
```

Output: 6.5