

# LAB 4

## Flow of Control (Repetition)

Name	:	Lee Pei Hui
Matric Number	:	22004781/1
Occurrence	:	7
Lecturer	:	Dr. Soraya

1. Write a program that accepts an integer from user. Then, display its entire factors in increasing order.

```
Enter an Integer: 24
The factors are: 1, 2, 3, 4, 6, 8, 12, 24
```

```
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);
        System.out.println("Enter an Integer: ");
        int integer = input.nextInt();

        System.out.print("The factors are: ");
        for (int j =1; j <= integer; j++) {
            if (integer % j == 0) {
                System.out.print(j + ", ");
            }
        }
    }
}
```

2. Write a program that accepts an integer n from user and then sum the following series.

$$1 + (1+2) + (1+2+3) + \dots + (1+2+3+\dots+n)$$

```
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner keyboard = new Scanner(System.in);

        System.out.print("Enter a number: ");
        int num = keyboard.nextInt();

        System.out.println((int)(Math.pow(num,3) + 3 * Math.pow(num,2) + 2 * num)/6);

        keyboard.close();
    }
}
```

3. Write a program that calculates the minimum, maximum, average and standard deviation (s) of the exam score in a subject. The program will accept the score and quit if negative score is entered. A sample output is given below.

$$s = \sqrt{\frac{\sum X^2 - \frac{(\sum X)^2}{N}}{N - 1}}$$

```
Enter a score [negative score to quit]:75
Enter a score [negative score to quit]:34
Enter a score [negative score to quit]:58
Enter a score [negative score to quit]:12
Enter a score [negative score to quit]:96
Enter a score [negative score to quit]:-1
1095.0
Minimum Score: 12
Maximum Score: 96
Average Score: 55.00
Standard Deviation: 33.09
```

```
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        //Declares variable
        int input=0, min=999, max=0, X=0, Xsqrd=0, N=0;

        //Get input
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a score [negative score to quit]: ");
        input = scanner.nextInt();

        while(input>=0) {
            min = Math.min(input, min);
            max = Math.max(input, max);
            X += input;
            Xsqrd += input * input;
            N++;

            System.out.print("Enter a score [negative score to quit]: ");
            input = scanner.nextInt();
        }

        System.out.print("Minimum Score: "+min);
        System.out.print("\nMaximum Score: "+max);
        System.out.print("\nAverage Score: "+ (double)X / N);

        double pt2 = Math.pow(X,2)/N;
        double X2 = (double)Xsqrd;
        System.out.printf("\nStandard Deviation: %.2f", Math.sqrt((X2 - pt2)/(N-1)));
    }
}
```

4. Write a program that ask user to enter the year and the first day of the year (0 for Sunday, 1 for Monday, ... , 6 for Saturday). Display the calendar for May and August.

```
import java.util.Scanner;

public class L4Q4 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        int year = 0, d1 = 0;

        System.out.print("Enter year and first day of the year (separated by 1 white space): ");
        year = sc.nextInt();
        d1 = sc.nextInt();

        //Add the number of days to May & August
        int daystoMay = 31 + (isLeap(year) ? 29 : 28) + 31 + 30;
        int daystoAug = daystoMay + 31 + 30 + 31;

        displayCalendar((d1 + daystoMay) % 7, year, "May", 31);
        displayCalendar((d1 + daystoAug) % 7, year, "August", 31);

        sc.close();
    }

    static boolean isLeap(int year) {
        return (year % 400 == 0) || (year % 100 != 0) && (year % 4 == 0);
    }

    if (m1>m2){
        System.out.print("Player 1 get the highest score! Player 1 won the game!");
    }else{
        System.out.print("Player 2 get the highest score! Player 1 won the game!");
    }
}
}
```

```

static void displayCalendar(int dayinWeek, int year, String month, int daysinMonth) {

    System.out.printf("\nCalendar for %s %d\n", month, year);
    System.out.println("┌───┬───┬───┬───┬───┬───┬───┐");
    System.out.println("│ Sun │ Mon │ Tue │ Wed │ Thu │ Fri │ Sat │");
    System.out.println("└───┴───┴───┴───┴───┴───┴───┘");
    System.out.print(("│      ").repeat(dayinWeek));

    for (int i = 1; i <= daysinMonth; i++) {

        System.out.printf("│ %s  ", (i < 10) ? "0" + i : i);
        dayinWeek++;

        if (dayinWeek % 7 == 0) {
            System.out.print("│\n┌───┬───┬───┬───┬───┬───┬───┐\n");
            dayinWeek = 0;
        }
    }

    System.out.print(("│      ").repeat(7 - dayinWeek) + "│\n┌───┬───┬───┬───┬───┬───┬───┐\n");
}
}

```

5. Write a simple two players dice game. Each player will take turns to roll a dice. The first players that reach more than 100 points win the game. If the player rolls a 6, the player will score 6 points and has the chance to roll again.

```
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        int p1Score = 0, p2Score = 0, roll = 0, bonus = 0;
        boolean isP1Turn = true;

        while(Math.max(p1Score,p2Score)<=100){
            roll = (int)(Math.random()*(6)+1);
            if(roll == 6){
                bonus = (int)(Math.random()*(6)+1);
            }

            if(isP1Turn){ p1Score += roll + bonus; } else {p2Score += roll + bonus;}

            System.out.printf("Player 1 Score: %d, Player 2 Score: %d\n", p1Score, p2Score);

            isP1Turn = !isP1Turn; //To take turns
        }
        System.out.printf("Player %d won!", p1Score > p2Score ? 1 : 2);
    }
}
```

6. Write a program that generates a non-negative random integer. Display the number of digits in the integer.

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

```
public class Main {
```

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

```
        int randomNum = (int) (Math.random() * 100); //Generate number
```

```
        int counter = (int) Math.log10(randomNum); //To calc number of digits, result+1
```

```
        System.out.printf("Number of digits in %d is %d.", randomNum, counter+1);
```

```
    }
```

```
}
```



7. Write a program that used to calculate mortgage loan. The program will create the amortization table with equal total payment plan. The following are the formula and the sample output.

$$M = (P * \frac{i}{12 * 100}) / (1 - (1 + \frac{i}{12 * 100})^{-N})$$

M = Monthly payment  
P = Principal  
i = yearly interest rate in %  
N = total number of months

$$C_n = M * (1 + \frac{i}{12 * 100})^{-(1 + N - n)}$$

$$L_n = M - C_n$$

$$R_n = L_n / \frac{i}{12 * 100} - C_n$$

C = Principal portion due  
n = month under consideration  
L = interest due  
R = remaining principal balance due

Enter principal amount: 10000

Enter interest in %: 4

Enter total number of month(s): 12

Month	Monthly Payment	Principal	Interest	Unpaid Balance	Total Interest
1	851.50	818.17	33.33	9181.83	33.33
2	851.50	820.89	30.61	8360.94	63.94
3	851.50	823.63	27.87	7537.31	91.81
4	851.50	826.37	25.12	6710.94	116.93
5	851.50	829.13	22.37	5881.81	139.30
6	851.50	831.89	19.61	5049.92	158.91
7	851.50	834.67	16.83	4215.25	175.74
8	851.50	837.45	14.05	3377.80	189.79
9	851.50	840.24	11.26	2537.56	201.05
10	851.50	843.04	8.46	1694.52	209.51
11	851.50	845.85	5.65	848.67	215.16
12	851.50	848.67	2.83	0.00	217.99

```
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner keyboard = new Scanner(System.in);

        double M = 0.0, P = 0.0, i = 0.0, C = 0.0, L = 0.0, R = 0.0, totalInterest = 0.0;
        int N = 0;

        System.out.print("Enter principal amount: ");
        P = keyboard.nextDouble();

        System.out.print("Enter interest in %: ");
        i = keyboard.nextDouble();

        System.out.print("Enter total number of month(s): ");
        N = keyboard.nextInt();

        System.out.printf("Month%23s%17s%16s%22s%20s\n", "Monthly Payment", "Principal", "Interest", "Unpaid Balance", "Total Interest");

        for (int n = 1; n <= N; n++) {

            M = (P * (i / (12 * 100))) / (1 - Math.pow((1 + (i / (12 * 100))), -N));
            C = M * Math.pow((1 + (i / (12 * 100))), -(1 + N - n));
            L = M - C;
            R = (L / (i / (12 * 100))) - C;
            totalInterest += L;

            System.out.printf("%s%23.2f%17.2f%16.2f%22.2f%20.2f\n", (n < 10) ? n + " " : n, M, C, L, R, totalInterest);
        }

        keyboard.close();
    }
}
```

8. Write a program that generates the first n prime number. n is an random integer within 0 to 100.

```
public class Main {
    public static void main(String[] args)
    {
        int n = 0; //The no. of numbers
        int ni = 0; //The counter
        int primeNum = 2; //The first prime number
        n = (int) (Math.random()*100); //Print prime number within 0 to 100

        System.out.printf("First %d prime number%s:\n", n,(n > 1) ? "s" : "");
        /*
        %d: n: Random no. of numbers generated
        %s: if n>1 is numbers, if n=1 is number
        */

        while (ni < n){ //while counter is less than the no. of numbers
            boolean isPrime = true; //if the number is prime

            for(int i = 2; i < primeNum; i++){
                //initialize i = 2
                // will skip the first loop as i=2=primeNum
                // the if loop is not run in the 1st time, i remains 2
                // the while loop runs, primeNum++; primeNum = 2+1 =3
                // the if loop starts to run if fulfill condition

                if(primeNum % i == 0){ //if the number is divisible by 2, the num is not prime
                    isPrime = false; //this will stop the if(isPrime) loop
                    break; //break the if loop and start with another num
                    // i++ won't function as each time the while loop runs a new if loop
                }
            }

            if(isPrime){ //Print out if the number is prime number
                System.out.printf("%3d%s%s",primeNum,
                    (ni == n-1)? "":", ", //If not the last digit, print "," after number
                    ((ni+1)%10 ==0)? "\n":"" ); //Print at new line for every 10 numbers
                ni++; //Counter+1 if prime number is printer
            }
            primeNum++; //Increase recent prime number by 1 before end while loop
        }
    }
}
```

Link:

<https://github.com/PeiHui369/Fundamentals-Of-Programming/tree/main/Lab%204>