Lab 03 Introduction to Java programming language 2

Part One (String and Arrays): ★★★☆☆

Task 1- Write java method that perform the following calculations:

1. The power of 5 of an integer given as a parameter.
2. The product of two integers given as a parameter minus their sum.
3. The product of three integers given as an integer **Array** parameter.

Task 2- In following a table of some Real Madrid team players.
Write a Method that takes a T-shirt number and returns a player's name.

Player	Courtois	Dani	Alaba	Nacho	Fran	Mbappé	Bellingham	Rodrygo	Vinícius
Tshirt N°	1	2	4	6	20	9	5	11	7

Task 3- Write a java method that checks if an email ends with "**@esi-sba.dz**".

Task 4- What will be the output of **task4** program =>

```
1  class task4 {
2
3  public static void main(String args[]){
4      int a[]={5,1,15,20,25};
5      int i,j;
6      int m;
7      i=++a[1];
8      j=a[2]++;
9      m=a[i++];
10     System.out.print(i+" "+j+" "+m);
11     }
12 }
```

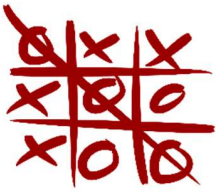
Task 5- Write a java method that takes a String as parameter and return the reverse of that String.

Task 6- Write a java method to reverse String using recursion.

Task 7- Code **Sales.java** (see end of last page) contains a Java program that prompts for and reads in the sales for each of 5 salespeople in a company. It then prints out the id and amount of sales for each salesperson and the total sales. Study the code, then modify the program as follows:

a) Compute and print the average sale. (no loop is necessary)
b) Find and print the maximum sale and the minimum sale.
c) After the list, sum, average, max and min have been printed, ask the user to enter a value. Then print the id of each salesperson who exceeded that amount, and the amount of their sales. Also print the total number of salespeople whose sales exceeded the value entered.
d) The salespeople are objecting to having an id of 0—no one wants that designation. Modify your program so that the ids run from 1–5 instead of 0–4. Do not modify the table
e) Instead of always reading in 5 sales amounts, at the beginning ask the user for the number of sales people and then create an array that is just the right size. The program can then proceed as before.

Lab 03 Introduction to Java programming language 2



Part Two (Coding a Game): ★★★★☆

We are interested to implement a **tic tac toe** game between two players using java programming language. The main method will contain the following code:

```
82  public static void main(String args[]){
83      show();
84      Scanner sc = new Scanner(System.in);
85      System.out.println("player 01 will play with: O ");
86      System.out.println("player 02 will play with: X ");
87      int row, col;
88      while(true){
89          System.out.println("player 01: select a cell number ");
90          char playr01 = sc.next().charAt(0);
91          System.out.println("player 01: select " + playr01);
92          row = get_cell_row(playr01);
93          col = get_cell_column(playr01);
94          playing(row,col, 'O' );
95          check_the_winner();
96          if(there_is_winner){
97              break;
98          }
99          System.out.println("player 02: select a cell number ");
100         char playr02 = sc.next().charAt(0);
101         System.out.println("player 02: select " + playr02);
102         row = get_cell_row(playr02);
103         col = get_cell_column(playr02);
104         playing(row,col, 'X' );
105         check_the_winner();
106         if(there_is_winner){
107             break;
108         }
109     }
110 }
```

The program output:

```
 7 | 8 | 9 |
 4 | 5 | 6 |
 1 | 2 | 3 |
player 01 will play with: O
player 02 will play with: X
player 01: select a cell number
1
player 01: select 1
 7 | 8 | 9 |
 4 | 5 | 6 |
 O | 2 | 3 |
Verification output:  789 456 O23 74O 852 963 753 359
player 02: select a cell number
2
player 02: select 2
 7 | 8 | 9 |
 4 | 5 | 6 |
 O | X | 3 |
Verification output:  789 456 OX3 74O 85X 963 753 359
player 01: select a cell number
4
player 01: select 4
 7 | 8 | 9 |
 O | 5 | 6 |
 O | X | 3 |
Verification output:  789 O56 OX3 7OO 85X 963 753 359
player 02: select a cell number
5
player 02: select 5
 7 | 8 | 9 |
 O | X | 6 |
 O | X | 3 |
Verification output:  789 OX6 OX3 7OO 8XX 963 7X3 3X9
player 01: select a cell number
7
player 01: select 7
 O | 8 | 9 |
 O | X | 6 |
 O | X | 3 |
Verification output:  O89 OX6 OX3 OOO 8XX 963 OX3 3X9
Player 01 is the winer !
```

Todo: In the first tasks, we will start implement the display of the game, then we will move to the game functionalities.

Task 01: Define a character 2D array that contains values from 1 to 9 (as shown in table 1) of type java **char [][].**

Task 02: Create a java method named ***show()*** that iterate for each value of the array and print it. The output should be as show in table 02:

Table 02

7	8	9
4	5	6
1	2	3

Table 01

7	8	9
4	5	6
1	2	3

Task 03: When a use choose a cell number, we should know which row and which column of his choice. For example if player 1 choose cellule 5 we need to implement two method that return column and row number from array

public static int get_cell_row(char input)

public static int get_cell_column(char input)

Task 04: Implement a method called ***playing(row, col, 'user character: O or X')*** that takes as parameters row, col, and player character to update the 2D array**.**

Task 05 :

Implement a method that check the winner called ***check_the_winner()***. That check how is the winner.

Task 06 << Optional **>>:**

Implement a method that makes one of the players play randomly (human vs Random player)

Task 07 << Optional **>>:**

Implement a function that treats all possible cases and makes it hard from a player to beat a computer.

```java
import java.util.Scanner;

public class Sales {
   public static void main(String[] args) {
        final int SALESPEOPLE = 5;
        int[] sales = new int[SALESPEOPLE];
        int sum = 0;
        Scanner scan = new Scanner(System.in);
        // Input sales for each salesperson
        for (int i = 0; i < sales.length; i++) {
           System.out.print("Enter sales for salesperson " + (i+1) + ": ");
           sales[i] = scan.nextInt();
        }
        System.out.println("\nSalesperson Sales");
        System.out.println("--------------------");

        // Display sales for each salesperson and calculate total sales
        for (int i = 0; i < sales.length; i++) {
            System.out.println(" " + (i + 1) + " " + sales[i]);
            sum += sales[i];
        }
        System.out.println("\nTotal sales: " + sum);
   }
}
```


Dr. Oussama SERHANE (o.serhane@esi-sba.dz)