

Aim: Practicing Java programming language basics.

(1) Structured program development using selection structures (if, else, switch-case) and repetition structures (for, while, do-while).

(2) Program development using arrays and methods.

TO DO @ LAB:

1. Write a program to process daily temperature in degrees Celsius. Your program should read the temperature value of the day from the user and then should classify the value as one of the following results: “hot day” (with a temperature value as 30 degrees Celsius or greater), “pleasant day” (with a temperature value between 29-17 degrees Celsius) or “cold day” (with a temperature value as 16 degrees Celsius or lower).

Sample output (bold parts are entered by user):

```
Please enter a degree value (in Celsius): 15
15 is a cold day.
```

2. Write a program to process a collection of daily temperature values in degrees Celsius. Your program should count and print the number of hot days (with a temperature value as 30 degrees Celsius or greater), the number of pleasant days (with a temperature value between 29-17 degrees Celsius), the number of cold days (with a temperature value as 16 degrees Celsius or lower), and the hottest and coldest days' temperature values. The program should display the classification result for each temperature value entered and should also display the average temperature at the end. In your program, use a loop structure that helps you read input values from the user continuously. There is no predefined number of input values specified, so read the temperature values from the user until a -1000 (negative thousand) value is entered and print the related classification result for each input value. When the value -1000 is entered by the user the loop will be terminated and the other results will be printed as shown below.

Sample output (bold parts are entered by user):

```
Please enter degree values in Celsius (enter -1000 to quit):
15
15 is a cold day
18
18 is a pleasant day
11
11 is a cold day
34
34 is a hot day
29
29 is a pleasant day
10
10 is a cold day
-1000
```

```
Number of cold days: 3
Number of pleasant days: 2
Number of hot days: 1
```

```
The lowest temperature is 10 degrees Celsius.
The highest temperature is 34 degrees Celsius.
The average temperature is 19.5 degrees Celsius.
```

3. Write a program to compute and print the **factorial** of a given nonnegative integer.

Sample output:

```
Enter a nonnegative integer: 6
6! = 720
```

4. Write a program to compute and print the **factorial** values of the integers 1 through 7.

5. Suppose that a “hot day” is the one with a temperature value greater than 33 degrees Celsius. Write a program to process an **array** of 30 integers where each item refers to a temperature value for a day. In the program, read 30 integer temperature values into the array, print out the total number of hot days in the array, and also print out the highest temperature value in the array.

TO DO @ HOME:

6. Write a program to compute and print the result value of 5^3 . (To compute the result, either call the `static method pow` of `class Math` or use a **loop** to find the repeated multiplication of **5** for **3** times.)

7. Write a program to compute and print the result of a^b where **a** and **b** refer to nonnegative integer values to be entered by the user. (In your program, do not call the method `pow`; use a **loop** to find the repeated multiplication of **a** for **b** times.)

8. Write a program to do the following tasks in sequence:

- Declare and define an **array** of 8 integers.
- Fill the array using the values to be given by the user.
- Print out the contents of the array.
- Search the array for a key value to be given by the user. If the key value exists in the array, the program will print out "Found", else "Not found" will be printed out.
- Find and print out the minimum value in the array.

9. Write a program that contains the following 3 methods: *iterativeFactorial*, *recursiveFactorial*, and *main*. The method *iterativeFactorial* takes a nonnegative integer parameter and returns the value of its factorial using **iteration**. The method *recursiveFactorial* takes a nonnegative integer parameter and returns the value of its factorial using **recursion**. The method *main* prints out the factorial of 5 using one of the methods above.

The factorial of a nonnegative integer n can be defined using the following formulas:

iterative algorithm: $n! = 1 * 2 * \dots * (n-1) * n$

recursive algorithm: *base condition, $n=0$:* $0! = 1$ *recursive condition, $n>0$:* $n! = n * (n-1)!$

10. A **two-dimensional array** may represent a matrix. Write a program that asks user to fill two 3x2 matrices of integers and adds them. After printing the addition result as a matrix on the screen, the program finds and prints the maximum number in the result matrix.

An example addition of two 3x2 matrices:

$$\begin{bmatrix} 1 & 3 \\ 1 & 0 \\ 1 & 2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 7 & 5 \\ 2 & 1 \end{bmatrix} = \begin{bmatrix} 1+0 & 3+0 \\ 1+7 & 0+5 \\ 1+2 & 2+1 \end{bmatrix} = \begin{bmatrix} 1 & 3 \\ 8 & 5 \\ 3 & 3 \end{bmatrix}$$