Institute of Software Engineering

Graduate Diploma in Software Engineering

Final Examination, Semester 1, 2021

ITS1010 Programming Fundamentals

Time Duration: 3 Hours
This paper has 4 pages including this page and Any appendices.

Authorized Materials

Materials you can use during this open-book exam are usually unrestricted. Materials and resources that you might use on can include:

- Your own notes from lectures, given lecture notes, online learning resources
- Readings, reference to materials or textbooks

Instruction to students

The main restriction for an open-book assessment is that the work must be Students' own work to avoid plagiarism and collusion. See following guidance on plagiarism for further information.

- During the assessment you must not be helped by any other person to answer any questions. Collusion will be treated as an academic offence.
- DON'T use another student's work, or submit someone else's work as your own.
- DON'T pay other students or someone outside to do your work.
- DON'T use copy and paste to take text from another source. Instead, you should do paraphrasing and summarizing.

If someone violates any of the terms above, all the parties who are involved in plagiarism will be given zero marks.

How can student write answers to the exam paper?

Students should write answers by using a black or blue ink pen in white A4 papers and scan (or simply take photos if they have a good camera phone) the handwritten work to a clean, small-sized, black-and-white PDF (single file) and submit to the Google Classroom.

File Name format: **<Your Full Name> - <Your Student ID>**

Submitting outside of the set time will constitute a failure.

Section A

The weather department has decided to computerize their manual data analysis process. As the first step they have decided to computerize the rain data. The department daily monitor and log the rain details, they log the daily rain amount in millimetres (mm). You have been assigned to write the system to store and analyse the rain data. Using arrays of arrays (two dimensional arrays) write a program to store the rain data for a year specific area in Sri Lanka. (E.g. Colombo city area) and achieve the following requirements.
Consider the following:
The days in a month can vary for each month (28, 30, 31… etc.)
The rain amount is logged with 0.00 format.

1. Create a two-dimensional array to store rain data (rows – months, columns – number of days per month and you can user the following single int array to create your array.

```
int [] days_per_months = { 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };
```

2. Write a java method to input all rain data from the keyboard (365 days) for a specific year.
('Scanner' can be used for keyboard input and nested for-loop must be used).

3. Write a java method to print all rain data in a tabular form as follows.

January	93.00	72.00	42.00	26.00	97.00	83.00	94.00	69.00	66.00	..	..	..	..	..
February	37.00	28.00	58.00	64.00	88.00	37.00	72.00	77.00	42.00	..	..	..	..	
March	79.00	15.00	52.00	21.00	15.00	50.00	81.00	..	..	..	..	..		
April	98.00	83.00	82.00	2.00	69.00	37.00	16.00	97.00						

(The following String array can be used to print the data as the above form)

```
String[] month_names = {"January", "February", "March", "April", "May", "June", "July", "August", "September", "October",
"November", "December"};
```

4. Write a method to return the day in a given month, which had a
   a. Minimum rain reading in a month.
   b. Maximum rain reading in a month.

5. Write a method to return the day (dd-mm format) in a year, which had a
   a. Minimum rain reading in the year.
   b. Maximum rain reading in the year.

6. Write a method to return the average rain in a given month.

7. Write a method to return the average rain in the year.

8. Write a method to return a list of months which had rain more than yearly average.

9. Write a method to return a list of months which had rain less than yearly average.

10. Write a method to return a list which contains display the rained amount for the given day of each month.

Section B – Answer All Questions

Question 01

Write statements to accomplish each of the following tasks:

a. Declare variables *physics*, *chemistry*, *mathematic* and *total* to be type *int* and "*average*" to be of type *double*.
b. Input marks for each subject from the keyboard, Assume Scanner variable input can be used to read a value from the keyboard.
c. Find the total of all subjects and assign into the variable total.
d. Find the average marks and assign into variable 'average'
e. Print the total marks, as "Total subjects marks is : 156 "Use method System.out.println().
f. If the average is greater than or equals to 65 print "The Student is eligible" otherwise "not eligible " .

Question 02

Perform the following tasks for an array called *fractions*:

a. Declare a constant variable array_size that's initialized to 10.
b. Declare an array with array_size elements of type double, and initialize the elements to 0.
c. Assign the value 1.667 to array element 9.
d. Assign the value 3.333 to array element 6.
e. Sum all the elements of the array, using a "for-each" loop.

Question 03

a. What are the loops that you learned in java?
b. Explain the difference between while loop and do while loop.
c. What is the usage of for-each loop? Explain with coding examples.
d. Write a program to enter the numbers till the user wants and at the end it should display the count of positive, negative and zeros entered.

Question 04

a. Explain post increment and pre increment using coding examples.
b. What is casting and conversation? Explain all using coding examples.
c. Write a program to simulate an air conditioner using a java switch case. Normally air conditioners work in power saving mode. If the temperature is 20, **the fan starts**. If the temperature is 50, switch **on the air cooler.**
d. Explain the data representation on RAM? (Should explain stack and heap memory)

Section C – Answer All Questions

Question 01

Write a java method to sort an integer array (Use the Selection Sort algorithm or bubble sort algorithm). The selection sort is a combination of searching and sorting. During each pass, the unsorted element with the smallest (or largest) value is moved to its proper position in the array. The number of times the sort passes through the array is one less than the number of items in the array.

Question 02

Write a method called **mode** that returns the most frequently occurring element of an array of integers. Assume that the array has at least one element and that every element in the array has a value between 0 and 100 inclusive. Break ties by choosing the lower value. For example, if the array passed contains the values {27, 15, 15, 11, 27},
Your method should return 15.

Question 03

Write a java method to find the sum of digits of a given number.