



UTM
UNIVERSITI TEKNOLOGI MALAYSIA

SCHOOL OF COMPUTING
Faculty of Engineering

UNIVERSITI TEKNOLOGI MALAYSIA

TEST 2 (PROGRAMMING)

SEMESTER I 2020/2021

SUBJECT CODE : SECJ/SCSJ1013
SUBJECT NAME : PROGRAMMING TECHNIQUE I
YEAR/COURSE : 1 (SECJ/ SECV/ SECB/ SECR/ SECP)
TIME : 21:00 – 23:00 MYT (1 hour 50 minutes)
DATE : 4th JANUARY 2021 (Monday)

INSTRUCTIONS TO THE STUDENTS:

- Please read the *General Guidelines for the Programming Technique I Test 2* that is shared in Telegram's Group and/ or provided in UTM e-learning
- Read the problem and instructions carefully.
- You are given **ONE HOUR FIFTY MINUTES** to complete the test inclusive of the submission of your program (**1 hour 20 minutes to answer** the question, **15 minutes to submit** the partial answer, and **15 minutes to submit** the final answer).
- Your program must follow the input and output as required in the text and shown in the examples. You must test the programs with (but not limited to) all the input given in the examples.

IMPORTANT NOTES:

- All the **COMMENT STATEMENTS** in the submitted program **WILL NOT BE EVALUATED**.

SUBMISSION PROCEDURE:

- Only the source code is required for the submission and the source code's file shall be named as follows: *Name_matricesNo_section.cpp* (i.e. *AinaAli_A20EC018_01.cpp*).
- You do not need to compress the file.
- Submit the source code file via the **UTM's e-learning system**.

SAMPLE SET 1

Question

[65 Marks]

Write a complete C++ program that helps the cashier to calculate the price of the rent movies. The program should perform the following tasks:

Task 1: Write a function named **calcAverage**. (4.5 marks)

- a) It takes the total number of each rented movie type as input parameters.
- b) The function should calculate the average number of movies rented per movie type.
- c) It should return the average value calculated in (b).

Task 2: Write a function named **calcPrice**. (9.5 marks)

- a) It takes the total rate as an input parameter.
- b) The function should calculate the final price based on the discount based on the conditions in Table 1.

Table 1

Rate Range	Discount(%)
Below RM 35.00	10
Between RM 35.00 to RM 60.00	15
Above RM 60.00	25

- c) It should return the final price calculated in (b).

Task 3: Write a function named **getInput**. (5.5 marks)

- a) This is a non-returning function.
- b) It takes the number of movies to rent for each movie type (movie type 1, movie type 2, and movie type 3) as input parameters.
- c) The function should ask the user to enter the number of movies to rent for each movie type.
- d) It sends all the values entered by the user in (c) back to the calling module through the use of reference parameters.

Task 4: Write a function named **dispRentalSummary**. (13 marks)

- a) This is a non-returning function.
- b) It takes the overall total of total rate and the total number of each rented movie type as input parameters.
- c) The function should display the overall total of total rate and the average number of movies rented per movie type by calling the **calcAverage** function as been shown in the Rental Summary section as in sample execution given below.
- d) The function also needs to determine which type of movie that have the highest total number of rental and how much the total number of the highest rented movie type.

Note: You are **NOT ALLOWED** to use any **predefined function(s)** to determine the highest number of rented movie types.

Task 5: Write a **main** function to perform the following tasks: **(21.5 marks)**

- You need to use an appropriate **LOOP** to perform the process in this function.
The loop will be terminated when the user press 'N'.
- You are **NOT ALLOWED** to use **arrays** except an array of characters.
- The function should ask the user to enter a renter name.
- The function should calculate the total rate using the following formula:
$$\text{Total Rate} = (\text{Movie Type 1} \times 2) + (\text{Movie Type 2} \times 5) + (\text{Movie Type 3} \times 3.5)$$
- The function will sum up the overall total of total rate and calculate the total number of each rented movie type.
- The function may need to call the functions that are defined in Task 2, Task 3, and Task 4.
- The program should produce the output as in the sample execution given below.
Note: The values in **bold** are input by the user.

Task 6: List all function prototypes. **(4 marks)**

Task 7: You must ensure your program fulfill the following criteria: **(7 marks)**

- The program is able to run.
- The program uses an appropriate structure for the program (e.g. all required header files are included, the program is properly written, proper indentation, etc.)

SAMPLE PROGRAM EXECUTION

[illegible]

Renter Name : **Nick Fury**

Movie Type 1: **1**

Movie Type 2: **2**

Movie Type 3: **1**

[illegible]

Total Rate (RM) : 15.5

Final Price (RM) : 13.95

Press [N] to stop...**y**

[illegible]

Renter Name : **Tony Stark**

Movie Type 1: **4**

Movie Type 2: **4**

Movie Type 3: 0

[illegible]

Total Rate (RM): 30

Final Price(RM): 27

Press [N] to stop...**n**

<<<<<<< RENTAL SUMMARY >>>>>>>

Overall Total Rate (RM) : 160.5

Average for 3 types of rented movies: 16

Highest type of movie rented : Movie Type 1 (19)

SAMPLE SET 2

Question

[65 Marks]

Write a complete C++ program that helps the teacher to calculate the result of students in the test of Programming Technique. The program should perform the following tasks:

Task 1: Write a function named **getInput**. (5.5 marks)

- a) This is a non-returning function.
- b) It takes the score of question 1, score of question 2, and score of question 3 as input parameters.
- c) The function should ask the user to enter the score (per 100) for each question.
- d) It sends all the values entered by the user in (c) back to the calling module through the use of reference parameters.

Task 2: Write a function named **dispTier**. (7.5 marks)

- a) This is a non-returning function.
- b) It takes the total marks as an input parameter.
- c) The function should display the tier based on the conditions in Table 1.

Table 1

Tier	Total Marks
Tier 3	Below 40%
Tier 2	Below 75%
Tier 1	75% and above

Task 3: Write a function named **calcAverage**. (4 marks)

- a) It takes the number of students and total marks as input parameters.
- b) The function should calculate the average marks per student.
- c) It should return the average marks calculated in (b).

Task 4: Write a function named **dispSummary**. (4 marks)

- a) This is a non-returning function.
- b) It takes the total marks as an input parameter.
- c) The function should display the total marks and tier by calling the **dispTier** function.

Task 5: Write a **main** function to perform the following tasks: (33 marks)

- a) You need to use an appropriate **LOOP** to perform the process in this function. The loop will be terminated when the user press ENTER or empty value for student's name.
- b) You are **NOT ALLOWED** to use **arrays** except an array of characters.
- c) The function should ask the user to enter a student's name.
- d) The function may need to call the functions that are defined in Task 1, Task 3, and Task 4.

- e) The function should calculate the total marks using the following formula:

$$\text{Total Marks} = \left(\frac{\text{Score for Question 1}}{100} \times 35 \right) + \left(\frac{\text{Score for Question 2}}{100} \times 25 \right) + \left(\frac{\text{Score for Question 3}}{100} \times 40 \right)$$

- f) The function should determine the student with the highest and lowest total marks, and calculate the number of students, and the average marks per student.
Note: You are **NOT ALLOWED** to use any **predefined function(s)** to determine the highest and lowest total marks.
- g) The program should produce the output as in the sample execution given below.
Note: The values in **bold** are input by the user.

Task 6: List all function prototypes. **(4 marks)**

Task 7: You must ensure your program fulfill the following criteria: **(7 marks)**

- The program is able to run.
- The program uses an appropriate structure for the program (e.g. all required header files are included, the program is properly written, proper indentation, etc.)

SAMPLE PROGRAM EXECUTION

```
<<<<<<<<<<< DATA >>>>>>>>>>>
Name      : Rashid Hamid
Q1 mark:   41
Q2 mark:   42
Q3 mark:   35
```

```
<<<<<<<<<< SUMMARY >>>>>>>>>>
Total marks: 38
Tier      : Tier 3
```

```
<<<<<<<<<<< DATA >>>>>>>>>>
```

Name	:	Hamidah Ahmad
Q1 mark:		75
Q2 mark:		56
Q3 mark:		68

```
<<<<<<<<<< SUMMARY >>>>>>>>>>
Total marks: 67
Tier      : Tier 2
```

```
<<<<<<<<<<< DATA >>>>>>>>>>>
```

Name	:	Lim Chong Kit
Q1 mark:		75
Q2 mark:		72
Q3 mark:		80

```
<<<<<<<<<< SUMMARY >>>>>>>>>>
Total marks: 76
Tier      : Tier 1
```

Q3 mark: **55**

Tier : Tier 2

Name :

Average for 4 students: 60

SAMPLE SET 3

Question

[65 Marks]

Write a complete C++ program that helps the Ministry of Health (MOH) to determine the status of a zone by calculating the number of active cases for COVID 19. The program should perform the following tasks:

Task 1: Write a function named **dispStatus**. (10 marks)

- This is a non-returning function.
- It takes the number of active cases as an input parameter.
- The function should display the status of a zone based on the conditions in Table 1.

Table 1

Number of active cases	Status of zone
Above 40	Red
21 until 40	Orange
1 until 20	Yellow
No case	Green

Task 2: Write a function named **getInput**. (7 marks)

- This is a non-returning function.
- It takes the number of total cases, new cases, total death, and total recovered as input parameters.
- The function should ask the user to enter the number of total cases, new cases, total death, and total recovered.
- It sends all the values entered by the user in (c) back to the calling module through the use of reference parameters.

Task 3: Write a function named **dispOutput**. (4 marks)

- This is a non-returning function.
- It takes the number of active cases as an input parameter.
- The function should display the number of active cases and zone status by calling the **dispStatus** function.

Task 4: Write a function named **calcAverage**. (4 marks)

- It takes the number of states and total active cases as input parameters.
- The function should calculate the average number of active cases per state.
- It should return the average value calculated in (b).

Task 5: Write a **main** function to perform the following tasks: (29 marks)

- You need to use an appropriate **LOOP** to perform the process in this function. The loop will be repeated when the user press ENTER.
- You are **NOT ALLOWED** to use **arrays** except an array of characters.
- The function should ask the user to enter a state name.

- $$\text{Number of active cases} = \text{Total cases} + \text{New cases} - \text{Total Death} - \text{Total Recovered}$$

(4 marks)

(7 marks)

- header files are included, the program is properly written, proper indentation, etc.)

```
<<<<<<<<<< DATA >>>>>>>>>>
```

State name	: Negeri Sembilan
Total cases	: 7103
New cases	: 57
Total death	: 11
Total recovered:	6274

Press <ENTER> to continue...

Press <ENTER> to continue...

3

