

PROJECT SCSP1103 SEM 1 2017/18

QUESTION 1

Write a complete C program that reads in a list of data (Name of student, height in centimeter and weight in Kilogram) from a text file as shown below (Figure 1.1).

Salina binti Abu	154	65
Tan Tin Tun	180	70
Muhammad Aiman	170	55
Marina Chin	160	60
:	:	:
:	:	:

Figure 1.1

1. The program is able to
 - a. Calculate the BMI (body mass index) of students based on their weight and height. (use formula of BMI).
 - b. Determine and display the category of each student as follows.

underweight	BMI less than 18.5
Normalweight	BMI between 18.5 and 24.9
overweight	BMI between 25 and 29.9
obese	BMI 30 and above

- c. Calculate the number of student for each category
 - d. Find and display the average BMI of all students and average category.
2. The output of this program should be displayed on the screen and also into the output file as shown in Figure 1.2

Name	height (m)	weight (kg)	BMI	Category
=====	=====	=====	===	=====
Salina binti Abu	1.54	65	27.41	overweight
Tan Tin Tun	1.80	70	21.61	normalweight
Muhammad Aiman	1.70	52	17.99	underweight
Marina Chin	1.60	60	23.44	normalweight
:	:	:	:	:
:	:	:	:	:
Number of student underweight		: 3		
Number of student normal weight		: 12		
Number of student overweight		: 8		
Number of student obesity		: 0		
Average BMI of all students		: 22.61		
Average category		: normalweight		

Figure 1.2

3. The number of student is based on the input data from file data.
4. Use array (one-dimension or two-dimension) to store the input data from file and the output data.
5. The program should be written in several functions for example `calBMI()` function to calculate BMI, `category()` function to find category of each student, etc. Each function must be implemented with the concept of parameter passing. Use appropriate arguments for each function. Do not use global variables.

QUESTION 2

Write a program for a restaurant. Your program should display a main menu that allows the customer to select between the food menu option, drink menu option, and check out option (Figure 2.1). After selected an option from the main menu, your program should continue to display a sub-menu that will show the available foods/ drinks and prices (Figure 2.2 and Figure 2.3). Your program should allow the customer to select the food/ drink and enter the order quantity (Figure 2.4). Your program should continue to allow the customer to order their food/ drink until they select the check out option in the main menu. Then your program will display the order list with the quantity, prices and total amount need to pay by the customer as shown in Figure 2.5 on the screen. Then, ask the user to select whether exit or back to the main menu for other customer make an order. Your program should able to calculate the total sales and the number of order as shown in Figure 2.6 and display it into the output file.

```
#main menu#  
1. Food menu  
2. Drink menu  
3. Check out  
Select your option:
```

Figure 2.1

```
#Food menu#  
1. Chicken chop -RM10.00/unit  
2. Fish and chip -RM13.00/unit  
3. Beef stick -RM11.50/unit  
Select your option:
```

Figure 2.2

```
#drink menu#  
1. Fresh orange -RM3.00/unit  
2. Manggo Lassi -RM3.50/unit  
3. Strawberry -RM4.00/unit  
Select your option:
```

Figure 2.3

```
Select your option: 1  
Enter quantity: 3
```

```
#main menu#  
1. Food menu  
2. Drink menu  
3. Check out  
Select your option:
```

Figure 2.4

```
Select your option: 3  
Order list  
Item                Price/unit  quantity  amount  
====              =====  
Chicken chop        10.00      3         30.00  
Fresh orange         3.00      3          9.00  
Total                39.00  
-----  
Back to the main menu-click 'y' / exit-click 'n':
```

Figure 2.5

- Use array (one-dimension or two-dimension) to store the order and the output.
- The program should be written in several functions for example `orderfood()` function to select food menu, `displaylist()` function to display order list, etc. Each function must be implemented with the concept of parameter passing. Use appropriate arguments for each function. Do not use global variables.

REPORT		
Number of order	:	5
Item	quantity	amount
====	=====	=====
Chicken chop	3	30.00
Fish and chip	0	0.00
Beef stick	2	23.00
Fresh orange	3	9.00
Manggo lassi	4	14.00
Strawberi	2	8.00
Total sales		84.00

Figure 2.6

QUESTION 3

Write a complete C program that reads userID, name, Kilowatt Hours(KWH), previous bill from the input file as shown below to generate electric bill.

011	Amirul Aiman	1500	150.00
112	Attasha Ali	300	0.00
143	Phua Pheng Lee	2000	100.00
:			
:			

Figure 3.1

The program could

- Calculate the current electric bill based on the rates as stipulated in the conditions supplied by Table below. Total amount to be paid is calculated by adding the current electric bill to the previous bill and 6% of the current bill (GST).

Usage	Rate
0-999 KWH	\$0.60
1000-1499 KWH	\$0.45
1500-1999 KWH	\$0.40
2000 or more KWH	\$0.35

- Ask the user to enter userID and display the electric bill for the entered userID on the screen as shown below.

Please enter the userID>> <u>143</u>	
ELECTRIC BILL	

UserID	: 143
Name	: Phua pheng Lee
KWH	: 2000
Rate`	: 0.35
Previous bill	: 100.00
Current bill	: 700.00
GST	: 42.00
Total amount	: 842.00

Figure 3.2

- c. Calculate the average of the current bill for all users and display the current bill into the output file as shown in Figure 3.3

ELECTRIC BILL OF ALL USERS		
UserID	Name	Total amount
=====	=====	=====
011	Amirul Aiman	786.00
112	Attasha Ali	190.80
143	Phua Pheng Lee	842.00
:		
:		
.....		
Average of the current bill for all users: RM516.67		

- d. The number of user is based on the input data from file data.
- e. Use array (one-dimension or two-dimension) to store the input data from file and the output data.
- f. The program should be written in several functions for example `calcBill()` function to calculate electric bill, `displayBill()` function to display electric bill on the screen, etc. Each function must be implemented with the concept of parameter passing. Use appropriate arguments for each function. Do not use global variables.

QUESTION 4

Write a complete C program that reads in a list of data (Matric No, Name of student, Point value for subject1, number of credit for subject1, Point value for subjek2, number of credit for subject2) from the input file as shown below (Figure 4.1).

170013	Salina binti Abu	3.33	4	4.00	3
170026	Tan Tin Tun	2.33	3	3.33	2
170127	Muhammad Aiman	2.00	3	4.00	3
170113	Marina Chin	3.67	3	4.00	4
:	:	:	:	:	:

Figure 4.1

The program is able to

- a. Calculate total point value
 $\text{Total pointvalue} = \text{pointvalue of subject1} * \text{credit of subject1} + \text{pointvalueof subject2} * \text{credit of subject2}$
- b. Calculate Grade point average (GPA)
 $\text{GPA} = \text{totalpointvalue} / \text{total credit}$
- c. Determine the transcript message based on the GPA as shown in the following table.

GPA	Transcript message
0.0-0.99	Failed
1.0-1.99	On probation
2.0-2.99	Average
3.0-3.49	Dean's list
3.5-4	Highest honour

- d. Ask the user to enter matric number and display the transcript for the entered matric number on the screen as shown below (Figure 4.2)

```

Please enter matric no >> 170026

*****
TRANSCRIPT
Name      :      Tan Tin Tun
Matric No  :      170026

No      Pointvalue  kredit
==      =====  =====
1.      2.33        3
2.      3.33        2

Total pointvalue      : 13.65
Total credit          : 5
Grade Point Average (GPA) : 2.73
"Average"

```

Figure 4.2

- e. Calculate number of students fail and pass in this semester and display the transcript for all students into the file output as shown in Figure 4.3

Matric No	Name	GPA	Message
=====	=====	=====	=====
170013	Salina binti Abu	3.62	Highest honour
170026	Tan Tin Tun	2.73	Average
170127	Muhammad Aiman	3.00	Dean list
170113	Marina Chin	3.86	Highest hounour
:	:	:	:
.....			
Number of student fail		:	2
Number of student pass		:	7

Figure 4.3

- The number of students is based on the input data from file data.
- Use array (one-dimension or two-dimension) to store the input data from file and the output data.
- The program should be written in several functions for example `calcGPA()` function to calculate GPA, `Message()` function to determine transcript message, etc. Each function must be implemented with the concept of parameter passing. Use appropriate arguments for each function. Do not use global variables

QUESTION 5

Write a complete C program that reads in a list of rainfall data in **inch** from the file input as shown below

	Jan	Feb	Mar	Apr	Mei	Jun	July	Aug	Sept	Oct	Nov	Dec
North	14	13	11	9	5	3	1	1	4	8	9	12
South	17	18	15	13	11	9	7	8	9	10	13	15
East	9	8	6	4	2	1	0	1	3	7	9	10
West	12	11	9	6	4	2	1	3	5	8	10	13

Figure 5.1

The program should

- Convert the rainfall in **inch** to **millimeter** (mm).
- Calculate the average of the rainfall for each month and each region.
- Determine whether “Dry”, “Moderate” or “Wet” for each month and each region based on the table below

Rainfall in millimeter	Type
≤ 127 mm	Dry
> 127 mm and ≤ 254 mm	Moderate
> 254 mm	Wet

- Calculate the number of Dry month, the number of Moderate month, and the number of Wet month.
- Find the driest month and the wettest month of the year.
- Display the output on the screen and also into the output file as shown in Figure 5.2.

RAINFALL												
	Jan	Feb	Mar	Apr	Mei	Jun	July	Aug	Sept	Oct	Nov	Dec
	====	====	====	===	===	===	=====	===	=====	===	===	=====
Avg(mm)	330.2	304.8	279.4	56.2	76.2	203.2
Type	wet	wet	wet	Dry	Dry	Moderate
Region	Avg (mm)		Type									
=====	=====		=====									
North	177.8		Moderate									
South	304.8		Wet									
East	127.0		Dry									
West	177.8		Moderate									
number of Dry month : 3												
number of moderate month: 6												
number of wet month : 3												
the driest month : Aug												
the wettest month : Jan												

Figure 5.2

- Use array (one-dimension or two-dimension) to store the input data from file and the output data.
- The program should be written in several functions for example `calAvg()` function to calculate average, `detType()` function to determine dry, moderate and wet, etc. Each function must be implemented with the concept of parameter passing. Use appropriate arguments for each function. Do not use global variables.