Programming Technique 1 Assignment 2

Question 1:

Write a program that implement the day of the week in a program. The program should be able to perform the following operations:

- Set the day.
- Print the day.
- Return the day.
- Return the next day.
- Return the previous day.
- Calculate and return the day by adding certain days to the current day. For example, if the current day is Monday and we add 4 days, the day to be returned is Friday. Similarly, if today is Tuesday and we add 13 days, the day to be returned is Monday.

Your program should allow the user to input as many day as he wants until he choose to terminate the program.

Question 2:

Write a program that receive user's inputs today's date as three integers using the day/month/year format, and determines and prints tomorrow's date correctly. Your program should be capable of handling cases such as shown below:

INPUT	OUTPUT
32 5 1993	Invalid date
31 5 1993	1 6 1993
28 2 1993	1 3 1993
29 2 1993	Invalid date
29 2 1992	1 3 1992
31 12 1993	1 1 1994
30 6 1993	1 7 1993

Your program should allow the user to input as many date as he wants until he choose to terminate the program.

Question 3:

Write a program that asks the user to enter today's sale rounded to the nearest RM100 for each of three stores. The program should then display a bar graph comparing each store's sales. Create each bar in the graph by displaying a row of asterisks. Each asterisk should represent RM100 of sales. Your program should also display the store which have the highest sales along with its sales.

Input Validation: Do no accept sales figures less tha RM0.00.

Here is an example of the program's output. User input is shown in bold.

```
Enter today's sales for store 1: 1000

Enter today's sales for store 2: 1200

Enter today's sales for store 3: 900
```

Sample Output:

```
DAILY SALES

(each * = RM100)

Store 1: **********

Store 2: ************

Store 3:*********


Store with highest sales: Store 2

Highest Sales: RM1200
```

Question 4:

Write a program that lets the user enter the loan amount and loan period in number of years and displays the monthly and total payments for each interest rate starting from 3.5% to 8%, with an increment of ¼. If you enter the loan amount RM5000 for five years, it will display a table as follows:

```
Loan Amount: 5000

Number of Years: 5

Interest Rate          Monthly Payment        Total Payment

3.5%                   146.51                 5274.37
```

```
3.75%                   147.06                  5294.32
……..
7.75%                   156.1                   5619.8
8.0%                    156.68                  5640.54
```

Question 5:

The monthly payment for a given loan pays the principal and the interest. The monthly interest is computed by multiplying the monthly interest rate and the balance (the remaining principal). The principal paid for the month is therefore the monthly payment minus the monthly interest. Write a program that lets the user enter the loan amount, number of years, and interest rate, then displays the amortization schedule for the loan. If you enter the loan amount RM10000 for one year with an interest rate of 7%, it will display a table as follows.

```
Loan Amount: 10000
Number of Years: 1
Annual Interest Rate: 7%
Monthly Payment: RM865.26
Total Payment: RM10383.21
Payment #     Interest      Principal     Balance
1             58.33         806.93        9193.07
2             53.62         811.64        8381.43
……..
11            10.0          855.26        860.27
12            5.01          860.25        0.01
```

Question 6:

Write a program that prompts the user to enter the year and first day of the year, and displays the first day of each month in the year on the screen. For example, if the user entered year 2005, and 6 for Saturday, January 1, 2005, your program should display the following output:

```
January 1, 2005 is Saturday
……..
December 1, 2005 is Thursday
```

Question 7:

Write an application that keeps track of a student's food purchase at the campus cafeteria. A meal card is assigned to an individual student. When a meal card is first issued, the balance is set to the number of points. If the student does not specify the number of points, then the initial balance is set to 100 points. Points assigned to each food item are a whole number. Every time food items are bought, points are deducted from the balance. If the balance becomes negative, the purchase of food items is not allowed.

Question 8:

Write an application that displays the results of a series of 50 coin tosses. Use the `rand()` function (needs `#include <cstdlib>`) to generate a number between 0 an 1. After each coin toss, display whether the toss represents "heads" or "tails". If the result is 0.5 or less, the result represents "heads"; otherwise, it represents "tails". After the 50 tosses are complete, display the percentages of heads and tails.

Question 9:

Your little sister asks you to help her with her multiplication, and you decide to write a program that test her skills. The program will let her input a starting number, such as 5. It will generate multiplication problem ranging from 5 x 1 to 5 x 12. For each problem, she will be prompted to enter the correct answer. The program should check for her answer and should not let her advance to the next question until the correct answer is given to the current question.

Question 10:

Write a program that calculates the occupancy rate for a hotel. The program should start by asking the user how many floors the hotel has. A loop should then iterate once for each floor. In each iteration, the loop should ask the user for the number of rooms on the floor and how many of them are occupied. After all the iterations, the program should display how many rooms the hotel has, how many of them are occupied, how many are unoccupied and the percentage of rooms that are occupied. The percentage may be calculated by dividing the number of rooms occupied by the number of rooms.

Input Validation: Do not accept a value less than 1 for the number of floors. Do not accept a number less than 10 for the number of rooms on a floor.

Question 11:

Write a program that will predict the size of population of organisms. The program should ask the user for the starting number of organisms, their average daily population increase (as a percentage), and the number of days they will multiply. A loop should display the size of the population for each day.

Input Validation: Do not accept a number less than 2 for the starting size of the population. Do not accept a negative number of average daily population increases. Do not accept a number less than 1 for the number of days they will multiply.

Question 12:

Write a program that diplay a series of at least four survey questions; the survey can be on any social or political topic you want, and each question should have at least three possible numeric-choice answers. At the end of the survey, ask the user whether he wants to (1) enter another set of responses to the same set of questions, or (2) quit. Continue to accept sets of responses until the user chooses to quit, and then, display the results of the survey – for each question indicate how many users chooses the first option, second option, and so on.

NOTES:

1. Each group must choose two of the assignments (as your first and second choice) listed above. I will assign the assignment using a "first-come-first-serve" policy. For example, the first group that email me with the choice of question 3 and question 9 as its second choice will get to do the question 3. But for the second group emailed me with the same choices will get to do the question 9 .
2. Please confirm with me the chosen projects together with your group members name via email (radziahm@utm.my) by **Saturday (19 November 2011).**

INSTRUCTIONS:

1. This assignment MUST be submitted by **Friday (02/12/11) before 4 p.m**. at my office.
2. Do this assignment in your Assignment 1 group.
3. For the solution, you are required to develop **pseudo code**, **flowchart** and **C++ code.**
4. Each group must submit a **hardcopy (pseudo code, flowchart, C++ code and a sample of running** - send to my office), **team evaluation form** (hardopy – enclosed in the report) and **softcopy** (via e-learning) of the C++ program.
5. Failure to submit this assignment on time will cost you **5 marks/day**.
6. **Copying/Plagiarism** in whatever form is strictly prohibited. No marks will be given (zero) to all involved parties if caught copying other's assignment(s).