# LAB 4: Decision Making –Part 2 & LAB 5: Looping – Part 1

For each problem below:

a) Analyze the problem by identifying input, output, formula, and constraint
b) Design an algorithm to solve the problem using pseudocode (so that you could include the pseudocode in your program)
c) Prepare several, appropriate number of test data to verify the correctness of your program
d) Prepare, compile, link, and execute the program to solve the problem
e) Test your program using the prepared test data
f) Write proper documentation in the program. Include the following information to form a banner at the beginning of your program:

```
/*******************************************************
* Author: your name and student ID                     *
* Course: the course code only                         *
* Section: your specific section number                *
* Date: of lab session                                 *
* Brief description:    of what problem the            *
*                       program tries to solve         *
* Pseudocode: write the algorithm to solve the         *
*             problem                                  *
* Test data: provide a set of test data                *
*             - input & expected output                *
*******************************************************/
```

**QUESTIONS**

DECISION MAKING – IF…ELSE IF, SWITCH

1. Write a program that reads your carry mark and prints a message on the screen based on the following table:

| Carry mark | Message |
|---|---|
| 40 to 50 | Hooray!<br>I am an extraordinary student. |
| 30 to 39 | Yeay!<br>I am good and will work hard to get an A for this course. |
| 20 to 29 | Hmmm…<br>Not good enough but I won't give up easily. |
| 10 to 19 | Oops!<br>What have I done this semester? |
| 0 to 10 | OMG!<br>Perhaps I should repeat this course. |

(Hint: use if…else if statements)

The valid range of the carry mark is in between 0 to 50. Apply proper data validation in this program.

2. Using a `switch` statement, write a program that reads an integer number in between 1 to 7 and prints on the screen "`Day `" followed by the number and the name of a day in a week based on the input value. For example, if you enter 3, "`Day 3 is Wednesday`" will be displayed on the screen. Assume that a week begins on Monday and ends on Sunday. Apply proper input data validation in this program.

3. Re-write the program in written for Question (2) using `if…else if` statements.

4. Using `if…else if`, write a basic calculator program that takes two real numbers as input and allows the user to select which operation they would like to perform on the two numbers, either:
    1. Addition
    2. Subtraction
    3. Multiplication
    4. Division

    Print out the result of the selected operation on the screen. Format the program's output based on (a) and sample input/output is shown in (b):

```
-----------------------------------------
              Basic Calculator
       A: Addition
       S: Subtraction
       M: Multiplication
       D: Division
-----------------------------------------
Enter two real numbers: x y
Select an operation: z

x operator-based-on-z-value y = t

[End-of-program]
```

(a)

```
-----------------------------------------
              Basic Calculator
       A: Addition
       S: Subtraction
       M: Multiplication
       D: Division
-----------------------------------------
Enter two real numbers: 30.5 2.3
Select an operation: D

30.5 / 2.7 = 13.26

[End-of-program]
```

(b)

5. Re-write the program written for question (4) using `switch` statement.

LOOPING – WHILE

6. Using `while` structure, write a program to repeatedly display the first two letters of your name 2 times. Format the program's output in the following manner:

```
\----------------------------------------/
\       BBBBB                  A         /
\       B    B                A A        /
\       BBBB                 A   A       /
\       B    B              AAAAAAA      /
\       BBBBB              A       A     /
\----------------------------------------/

\----------------------------------------/
\       BBBBB                  A         /
\       B    B                A A        /
\       BBBB                 A   A       /
\       B    B              AAAAAAA      /
\       BBBBB              A       A     /
\----------------------------------------/
```

7. Modify the program written for Question (6) so that it allows the user to determine how many times the two letters of your name will be printed on the screen. A sample input and output of the program is as follows:

```
How many times?: 3

\----------------------------------------/
\       BBBBB                A          /
\       B    B              A A         /
\       BBBB               A   A        /
\       B    B            AAAAAAA       /
\       BBBBB            A       A      /
\----------------------------------------/

\----------------------------------------/
\       BBBBB                A          /
\       B    B              A A         /
\       BBBB               A   A        /
\       B    B            AAAAAAA       /
\       BBBBB            A       A      /
\----------------------------------------/

\----------------------------------------/
\       BBBBB                A          /
\       B    B              A A         /
\       BBBB               A   A        /
\       B    B            AAAAAAA       /
\       BBBBB            A       A      /
\----------------------------------------/
```

8. Using `while` structure, modify the program written for Lab4 Question 5. In the new program repeatedly displays the selected face 3 times. Refer to below sample input/output to guide you in writing the new program.

```
MY SMILING/ANGRY FACE
Enter option [S/A]: A

     vvvvvvvvvvvvv
  / vvvvvvvvvvvvvvv \
  |   vvvvvvvvvvv   |
  |                 |
  |    /\       /\  |
 (|     *       *   |)
  |         |       |
   \     \----/     /
     \            /
       \________/

     vvvvvvvvvvvvv
  / vvvvvvvvvvvvvvv \
  |   vvvvvvvvvvv   |
  |                 |
  |    /\       /\  |
 (|     *       *   |)
  |         |       |
   \     \----/     /
     \            /
       \________/
```

```
     vvvvvvvvvvvvvv
  / vvvvvvvvvvvvvvvv \
  |   vvvvvvvvvvvv   |
  |                  |
  |    /\       /\   |
 (|     *       *    |)
  |         |        |
   \     \----/     /
     \             /
      \_________/

      Angry Face

Rate this face
How many LIKE?: 4

The angry face has 4 LIKE
```

9. Modify the program written for Question (8) so that it allows the user to determine how many times the smiling of angry face will be displayed on the screen. Rate the face's LIKE for every display. Before the program ends, display the total LIKE given to the face(s). A sample input and output of the program is as follows:

```
MY SMILING/ANGRY FACE
How many times?: 2

Enter option [S/A]: A
     vvvvvvvvvvvvvv
  / vvvvvvvvvvvvvvvv \
  |   vvvvvvvvvvvv   |
  |                  |
  |    /\       /\   |
 (|     *       *    |)
  |         |        |
   \     \----/     /
     \             /
      \_________/

      Angry Face

Rate this angry face
How many LIKE?: -2

Enter option [S/A]: S
     vvvvvvvvvvvvvv
  / vvvvvvvvvvvvvvvv \
  |   vvvvvvvvvvvv   |
  |                  |
  |   -----   -----  |
 (|     +       +    |)
  |         |        |
   \      \___/     /
     \             /
      \_________/

      Smiling Face

Rate this smiling face
How many LIKE?: 10

The face(s) received 8 LIKE
```