

LAB 5: Looping – Part 2

For each problem below:

- Analyze the problem by identifying input, output, formula, and constraint
- Design an algorithm to solve the problem using pseudocode (so that you could include the pseudocode in your program)
- Prepare several, appropriate number of test data to verify the correctness of your program
- Prepare, compile, link, and execute the program to solve the problem
- Test your program using the prepared test data
- 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

- Re-write the program written for Lab4-2 Question 6 using `do..while` loop.
- Re-write the program written for Lab4-2 Question 7 using `for` loop.
- Re-write the program written for Lab4-2 Question 8 using `for` loop.
- Re-write the program written for Lab4-2 Question 9 using `do..while` loop.
- Improve the program written for Lab4-2 Question 2 so that it repeatedly asks for an integer number and display the day message as long as the user does not enter value -999. A sample input and output of the program is as follows:

```

Enter any number between 1 to 7 [or -999 to exit]: 7

Day 7 is Sunday

Enter any number between 1 to 7 [or -999 to exit]: -999

[end-of-program]

```

- Improve the program written for Lab4-2 Question 4 so that the program exits only after the user chooses not to continue with it. Format the program's input/output in the following manner:

```
-----  
      Basic Calculator  
A: Addition  
S: Subtraction  
M: Multiplication  
D: Division  
-----  
Enter two real numbers: a b  
Select an operation: z  
  
a operator-based-on-z-value b = t  
  
Continue [ y or n ]?: y  
  
Enter two real numbers: a b  
Select an operation: z  
  
a operator-based-on-z-value b = t  
  
Continue [ y or n ]?: n  
  
[End-of-program]
```