Langara College

# Department of Computing Science & Information Systems

# CPSC1150 – Program Design

###### **Lab5: Strings and Loops**

**Objectives:**

* Solving problems and writing algorithms
* Processing strings
* Implementing algorithms withloop statements

**Problem [45 marks]**

**Instructions:**

1. Create a folder named **Lab5** to store all the files from this lab
2. Create an external documentation file (filename: **Lab5Ext.docx**) to store the summary, algorithm(s), and sample input and output for each problem.
3. All your programs must have good internal and external documentations

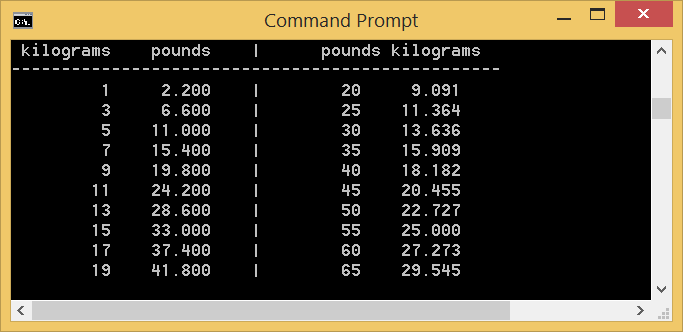
**Problem 1: [5 marks] Find a person’s middle name** (filename: **FindMiddleName.java**)

Design an algorithm and then write a program that asks a person to enter his/her first name, middle name, and last name in this order. Assume that the names are separated by single spaces. The program finds and displays the middle name of the person. You must use the Scanner’s ***nextLine()*** method to read the three names together and store them into a single string variable.

**Problem 2: [10 marks] Conversions between kilograms and pounds**

(filename: **KilogramsAndPounds.java**)

Design an algorithm and then write a program (You must use a ***while*** loop) that displays the following two tables on the **console** (note that 1 kilogram is 2.2 pounds. Declare a constant to store this value). The four columns in the table should be properly formatted as shown below. ***Hint***: use the System.out.printf method.



**Problem 3: [10 marks] Display Pyramid** (filename: **DisplayPyramid.java**)

Design an algorithm and then write a program that prompts the user to enter an integer from 1 to 15 (assume that the user always enters a valid integer) and displays a pyramid, as shown in the following sample run:

Enter a number between 1 and 15: 12

1

2 1 2

3 2 1 2 3

4 3 2 1 2 3 4

5 4 3 2 1 2 3 4 5

6 5 4 3 2 1 2 3 4 5 6

7 6 5 4 3 2 1 2 3 4 5 6 7

8 7 6 5 4 3 2 1 2 3 4 5 6 7 8

9 8 7 6 5 4 3 2 1 2 3 4 5 6 7 8 9

10 9 8 7 6 5 4 3 2 1 2 3 4 5 6 7 8 9 10

11 10 9 8 7 6 5 4 3 2 1 2 3 4 5 6 7 8 9 10 11

12 11 10 9 8 7 6 5 4 3 2 1 2 3 4 5 6 7 8 9 10 11 12

**Problem 4: [20 marks] Simulate the movement of an elevator** (filename: **Elevator.java**)

Design an algorithm and then write a program called Elevator, which models a simplified elevator.

1. The elevator operates in a building with 9 floors. The labelling of the floors begins at 1.
2. The user will request a floor by typing in an integer in the appropriate range. The elevator

you are modelling needs to “move” to the requested floor, displaying floor numbers and a

picture of an up/down arrow as it goes.

1. The program should continue moving to new floors and prompting for a floor selection until the user enters a sentinel value of 0 instead of a floor number. When the sentinel value is entered, the program should exit.
2. If the entered floor number is not valid, the user should be warned, and the user is asked to enter a floor number again.

A sample runs is given below:

o-------o

| 1 |

o-------o

The elevator is at Floor 1 now

Enter a floor number between 1 and 9, or enter 0 to quit: 5

The elevator is moving up from Floor 1 to 5

o-------o /\

| 5 | //\\

o-------o // \\

o-------o /\

| 4 | //\\

o-------o // \\

o-------o /\

| 3 | //\\

o-------o // \\

o-------o /\

| 2 | //\\

o-------o // \\

o-------o /\

| 1 | //\\

o-------o // \\

The elevator is at Floor 5 now

Enter a floor number between 1 and 9, or enter 0 to quit: 12

Invalid floor number. Try again.

Enter a floor number between 1 and 9, or enter 0 to quit: 3

The elevator is moving down from Floor 5 to 3

o-------o \\ //

| 5 | \\ //

o-------o \/

o-------o \\ //

| 4 | \\ //

o-------o \/

o-------o \\ //

| 3 | \\ //

o-------o \/

The elevator is at Floor 3 now

Enter a floor number between 1 and 9, or enter 0 to quit: 0

Thank you for using the elevator program. Goodbye!

**What to hand in**

Zip the folder **Lab5** which contains all the Java source files and the external documentation file from this lab and upload the zip file to BrightSpace.

**When to hand in**

By 8:29 am, Thursday, Feb. 17, 2022