**CSCI 1100 – 2017**

**Laboratory Report 7**

**Name:**

**Student ID:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Please indicate your registered lab room number:** | | | |
| **Rm 133** | **Rm 134** | **Rm 143** |  |

|  |  |  |
| --- | --- | --- |
| **Declaration**  **Complete this section by filling in the shaded column to accurately reflect your work** | | |
| 1 | This document is entirely my own work. | *Yes/no* |
| 2 | I obtained some help to complete this document. | *Yes/no.*  *If yes, from whom? Give details.*  *It is reasonable to obtain help from any person as long as you acknowledge the source.* |
| 3 | This document contains some material from the Internet or another document, file, or program. | *Yes/no.*  *If you got help from a source you need acknowledge this (e.g., the URL or a book reference). You must detail how you used this source.*  *Your lab work must still be your own. You should* ***not*** *copy and paste partial or full solutions.* |

Your task is to complete this report using Word and JGrasp. You may use your own computer, or one of the lab computers provided.

Your submission should be a **PDF** copy of this document that includes your completed work as well as a **ZIP** file containing your source code files. You should submit both your **PDF document** and **ZIP file** on Brightspace:

[http://dal.brightspace.com](http://dal.brightspace.com/)

**Submission Deadlines (firm):**

Monday Labs due: Wednesday by 12:00pm (**noon**)

Friday Labs due: Sunday by 12:00pm (**noon**)

NB:

* Try to submit this report *during* the lab so that your TA can check it for you before you submit!
* Attendance is mandatory in all labs, and will form part of your overall lab grade
* Acknowledge any help that you obtained from friends, TAs, the Learning Centre, etc., by completing the Declaration on the first page of this document. Obtaining help is fine, *so long as you acknowledge it!*
* Any students who cannot log on to the lab computers should speak to the Help Desk to set up their account.
* Textbooks, class handouts, and any other materials are welcomed and encouraged in all labs!
* Food and drink are not permitted in the computer labs
* Late labs are ***not*** accepted! It is known that computer errors, power outages, and network lag are 105% more likely to occur between 11:55-11:59am, in the moment they can do the most damage. Account for this, and give yourself the chance to make a timely submission!

**Exercise 1.**

1. Write a method called promptForInt that takes a single String parameter and returns an integer value. This method should use the JOptionPane class to generate a showInputDialog pop-up prompt for the user. The dialog in this input dialog prompt should use both the String parameter passed to promptForInt as well as a fixed String that specifies the user should enter an integer value. Your method should convert the user’s input to an integer value that is then returned.
2. Write a method called promptForDouble that takes a single String parameter and returns a double value. Follow the same outline as in part (a).
3. Write a main method that uses your code from parts (a) and (b) to do the following:
   1. Prompt the user for an integer, and print to output the next largest integer
   2. Prompt the user for a value, and print to output the square root of its absolute value
   3. Prompt the user for two integers, and print to output their product
   4. Prompt the user for three values, and print to output their average

Ensure that each prompt uses descriptive language, and makes it clear to the user what they should be giving as input.

For this question and all following questions:

* remember to use comments in your code where appropriate!
* include your source code in a class named after the exercise number (e.g., Exercise1)

**Exercise 2.**Review Exercise 4 from Lab 2.

1. Write a method called convertDistance that takes a double and a Boolean parameter and returns a double value that is calculated by converting between kilometres and miles. The method should convert the double parameter into either kilometres or miles, depending on the value of the Boolean parameter. Your method should convert from kilometres into miles if the Boolean parameter is true, and should convert from miles into kilometres otherwise.
2. Write a main method that prints to output two prompts for the user, and after each prompt pauses to collect input from the user using the Scanner class. Prompt the user once for a number of kilometres, and convert this into miles, and also prompt the user for a number of miles, which should be converted into miles. Use nicely formatted output that makes clear both the input and output. For instance, your main method could print out something like:  
     
    There are 6.21371 miles in 10.0 kilometres!  
     
    There are 8.04672 kilometres in 5.0 miles!

**Exercise 3.**

Review Exercise 5 from Lab 3.

1. Write a method called circleCirc that takes a single double parameter and returns a double value that is equal to the circumference of a circle. The method should calculate the circumference using the absolute value of the double parameter as the circle’s radius.
2. Write a method called circleArea that takes one double parameter and returns a double value that is equal to area of a circle. The method should calculate the area using the absolute value of the double parameter as the circle’s radius.
3. Write a method called sphereVolume that takes one double parameter and returns a double value that is equal to volume of a sphere. The method should calculate the volume using the absolute value of the double parameter as the sphere’s radius.
4. Write a main method that prints to output an appropriate prompt to the user, then uses the Scanner class to collect the radius of a circle. Print out both the circumference and area of the circle, using a nicely formatted message that also includes the original radius value, followed by the volume of a sphere having the same radius.

**Exercise 4.**

Review Exercise 1 from Lab 4.

1. Write a method getAverage that takes a single integer as a parameter and returns a double value. The method should use showInputDialog to prompt the user for a number of double values and then return their average. Your method should use the integer parameter as the number of prompts that are needed (and so also the number of values that need to be collected).
2. Write a main method that prints to output a prompt for the user. Use the Scanner class to read in a single integer value that will represent the number of values that will be provided to average. Then, use this value in a call to the method from part (a). Use the returned value from this method to print to output a nicely formatted message that gives both the average value, and also indicates the number of values that were collected.

**Exercise 5.**

1. Write a method called makeLine that takes one integer and two char parameters (element and separator) and returns a String. The method should return a String representing a single line that is formed from the two characters given as parameters. Your method should construct this line by using the element character alternating with the separator character. The integer parameter gives the number of element characters there should be in the String, and these should each be separated by one separator character. Additionally, there should be one separator character at the start and end of the line, and the whole String should have an end-line character at the end.  
     
   For example, if your method were called with integer parameter 3 and char parameters ‘A’ and ‘b’ respectively, then the returned String would look like this:  
     
    bAbAbAb
2. Write a method called makeGrid that takes one integer as a parameter and returns a String. The method should return a String representing a grid of squares, made up of – (hyphen), | (vertical line), and space characters. Your method should use the result of part (a) to generate a square grid using the integer parameter as the size. The grid should look like an array of squares, like on a checkerboard. For example, if called with the parameter 3, then the returned String would look like this:  
     
    - - -

| | | |

- - -

| | | |

- - -

| | | |

- - -

1. Write a main method prints to output a prompt for the user that asks for a single integer value, then use *either* the Scanner class or showInputDialog method to accept a single integer value as input. Use the method from part (b) to print to output a square grid of the appropriate size.

Before submitting, check that:

1. You have properly filled in your name, ID, and Declaration on the first page,
2. You have included your solution for each question that requires one,
3. Your solutions are easy to read and formatted appropriately,
4. You have secured your guest right by sharing bread and salt with your TAs (optional),
5. You have saved your submission (which should be a completed copy of **this file**) as a **PDF**,
6. You have included *all* of your source files (one per question) in a **ZIP** file,
7. You are preparing to submit **both** your **PDF** and **ZIP** file on Brightspace, and
8. You have logged off of any lab computers.