Lab Partner 1 Name
Lab Partner 2 Name

CS 115 Fall 2019 Lab #2

Due: **Thursday, September 12th, MIDNIGHT**
Points: **20**

Instructions:

1. Use this document template to report your answers. Enter all lab partner names at the top of first page.
2. You don't need to finish your lab work during the corresponding lab session.
3. Name the complete document as follows:

`LastName_FirstName_CS115_Lab2_Report.doc`

4. Submit the final document to Blackboard Assignments section before the due date. No late submissions will be accepted.
5. ALL lab partners need to submit a report, even if it is the same document.

Objectives:

1. (10 points) Demonstrate the ability to break a basic problem down into inputs, process and outputs.
2. (10 points) Demonstrate the ability to design test cases for your problem.

Problem 1:

Break a basic problem down into **inputs**, **process** and **outputs** and write **pseudocode** (step by step sequence of necessary actions) to solve the problem. **NO Java CODE IS NEEDED**

Answer the following questions for the problems listed below. Populate provided tables (enter as many rows as you find necessary) and pseudocode boxes with your answers. Feel free to add extra tables, boxes, comments, etc. if needed.

INPUTS: What are the inputs?

- What format / data type are they? (integer, real number, single character, string - a sequence of characters)
- Any valid / invalid / illegal / special values? (positive, negative, valid range, etc.)
- How do you get them? (enter manually, ask user, read from file, etc.)

PROCESS: How do you get from inputs to the outputs you want?

- What are the calculation steps?
- To follow these steps, what else do you need? (formulas, etc.)
- Other variables, constants, conversions (besides input and output variables)

OUTPUTS: What are the outputs?

- What format / data type are they in? (integer, floating-point, character, or string)

- Any valid / invalid / illegal / special values? (positive, negative, valid range, etc.)
- How do you output them? (display on screen, save to a file, plot, tabularize, etc.)

1. A teenager gets retained by a neighborhood association to distribute fliers, collect dues, and do miscellaneous chores. Just to make sure he is around when needed, he gets 40 dollars a month (he doesn't have to work for that). Plus he gets $11.25 per hour for any time he actually works in a month. Calculate how much the teenager earns if he works H hours in a month. **[2.5 points]**

Inputs and outputs (use "N/A", "undefined", "none", etc. If necessary)					
Variable name	Input or Output ?	Data type / format	Constraints	Special cases	Comments

Pseudocode:

What are the inputs? H hours worked in a month
What format are they in? (e.g., integer, floating-point, character, or string) real number, you can work decimal parts of an hour
Any valid or invalid values? (e.g. positive, negative, valid range, certain values) positive, some upper limit less than 200 hours maybe (5 weeks, 40 hours per week)
How do you get them? (e.g., prompt user or read from file) prompt user

How do you get from inputs to the outputs you want (process)? What are the calculation steps?

Salary = 40 + 11.25* H

To follow these steps, what else do you need?
Other variables, constants, conversion? (besides input and output variables)
40 hours base pay constant, 11.25/hour pay rate constant

Libraries (e.g., for calculations). NONE
What are the outputs? Salary earned
What format are they in? real number to two decimal points.
How do you output them? (e.g., on screen or to a file) screen

2. Consider the 3 dimensional barbell shown below.

Now:
a) Find the volume of the figure if the radius of each sphere is given, and the length of the bar connecting them is given, and the diameter of the bar is given (**all in the same units**),
b) Find the surface area of the figure. **[2.5 points]**

Inputs and outputs (use "N/A", "undefined", "none", etc. If necessary)					
Variable name	Input or Output ?	Data type / format	Constraints	Special cases	Comments

Pseudocode:

What are the inputs? Radius of each sphere, length of the bar
What format are they in? real, all greater than or equal to zero
How do you get them? prompt user or arguments

How do you get from inputs to the outputs you want (process)? What are the calculation steps?

volume_sphere=4/3*pi*r^3;
volume_bar=pi*(d/2)^2*length;
total_volume=2*volume_sphere +volume_bar
sa_sphere=4*pi*r^2;
sa_bar=pi*d*length;
total_sa=2*sa_sphere + sa_bar % subtract the surface area of the joint if you want more accuracy

To follow these steps, what else do you need?

Other variables, constants, conversion? (besides input and output variables)
PI
Libraries (e.g., for calculations). NONE

What are the outputs? total_volume, total_sa
What format are they in? floating point to 2 decimal digits
How do you output them? screen

3. Most stop watches allow you to display the time elapsed as a number of seconds, or as hours:minutes: seconds. So 5437 seconds or 1 hour:30 minutes:37 seconds Given an integer number of seconds, calculate the equivalent integer number of hours, integer number of minutes, and integer number of seconds. **[2.5 points]**

Inputs and outputs (use "N/A", "undefined", "none", etc. If necessary)					
Variable name	Input or Output ?	Data type / format	Constraints	Special cases	Comments

Pseudocode:

What are the inputs? seconds
What format are they in? positive integer
How do you get them? prompt user or arguments

How do you get from inputs to the outputs you want (process)? What are the calculation steps?

hours = floor(seconds/3600)
leftover = mod(seconds, 3600) or leftover=leftover-(hours*3600)
minutes = floor(leftover/60)
leftover = mod(leftover,60) or leftover=leftover-(minutes*60)

To follow these steps, what else do you need?
Other variables, constants, conversion? (besides input and output variables)

The number of seconds in an hour, the number of seconds in a minute
Libraries (e.g., for calculations).

What are the outputs? hours minutes seconds
What format are they in? integers
How do you output them? screen

4. A person suffering from type II diabetes injects insulin based on measurements of their blood sugar level. If the blood sugar level is less than 115, they don't need to inject any insulin at all. For a value of 115, they inject 1 unit of insulin. For every additional increase of 20 in her blood sugar level, they get one additional unit of insulin. (Thus, for a blood-sugar level of 134, they get 1 unit; for a blood-sugar level of 135, they get 2 units.) Create a formula for calculating the insulin injections and output a table that shows the number of units of insulin injected for a user input range of blood sugar values from 115 upwards. **[2.5 points]**

Inputs and outputs (use "N/A", "undefined", "none", etc. If necessary)					
Variable name	Input or Output ?	Data type / format	Constraints	Special cases	Comments

Pseudocode:

What are the inputs?
blood_sugar_range (start 115 end ?? increment ??)

What format are they in? (e.g., integer, floating-point, character, or string) Integer
Any valid or invalid values? (e.g. positive, negative, valid range, certain values) positive, some upper limit less than 500 maybe

How do you get them? (e.g., prompt user or read from file) prompt user
How do you get from inputs to the outputs you want (process)? What are the calculation steps?

insulinUnits = 1 + floor ((blood_sugar - base_blood_sugar_level) / 20)

To follow these steps, what else do you need?
Other variables, constants, conversion? (besides input and output variables) constants base_blood_sugar_level=115

Libraries (e.g., for calculations). NONE
What are the outputs? Table of insulin values
What format are they in? integer
How do you output them? (e.g., on screen or to a file) screen

Problem 2 (10 points):

Develop a test plan (a set of test cases) for the following problem. The goal of testing is to determine if the solution (a computer program) to your problem:

- Behaves correctly / produces correct results when given legal input values,
- Handles illegal input values correctly (for example: preventing division by zero),
- Behaves as planned when inputs are assigned "boundary" values.

In simple words, your test plan, at the very least, should:

- Consider "what could go wrong",
- Check if your problem solution behaves as expected.

Also, don't forget to make sure that you are using correct data types, for example: **number of children HAS TO be an INTEGER and you should have a test case that verifies that!**

Be VERY thorough!

NO Java CODE IS NEEDED. Figure out the calculation process first (feel free to include it as pseudocode. It is not necessary but may help to understand your thought process) and come up with test cases.

1. The bank wants to be sure you can afford to pay them back before they give you a mortgage. One way they consider your ability to repay is by making sure your total debt doesn't exceed a certain percentage of your income, usually 36-42%. This percentage is called the debt ratio. Given your monthly income (a real number) and other monthly debt payment(a real number), you can use the following formulas to determine the lower and upper limits on your monthly mortgage payments **[2 points]**:

- Lower limit = (36% of your income) minus your other monthly debt
- Upper limit = (42% of your income) minus your other monthly debt

Test case name	Input data set for this test	Explain why you chose this test

(ex. "negative height", "typical conditions", etc.)	case	case

This is the process answers and test cases should be based on:

INPUT – monthlyIncome real>0 monthlyDebt real>0
PROCESS need to also know the 36% and 42% constants
Lower limit = (36% of your income) minus your other monthly debt
Upper limit = (42% of your income) minus your other monthly debt
OUTPUT
Lower limit real>0, 2 decimal places
Upper limit real>0, 2 decimal places

Test cases that should be considered:

- Valid / invalid input data,
- High Debt (sample data: income: 4000 | monthly debt:1000),
- Low Debt (sample data: income: 6000 | monthly debt: 500),
- No Debt (sample data: income: 3000 | monthly debt: 0).

2. A jet aircraft is flying 'height' feet above a level plain at 'speed' mph. Suddenly the ground begins to rise at a 4 degree slope. Calculate the amount of time in seconds (to two decimal places) the pilot has to raise the nose before the aircraft strikes the ground. **[2 points]**:

Test case name (ex. "negative height", "typical conditions", etc.)	Input data set for this test case	Explain why you chose this test case

Solution should be based on this process (doesn't have to be included):

set height and speed identifiers

find other side of triangle (distance) with height and opposite angle 4 degrees
convert speed from mph to feet per sec
divide distance by speed to get time
format and output to two decimal places

Test cases that should be included:
- Valid / invalid input data
- small height / high speed
- large height / high speed
- small height / low speed
- large height / low speed

3. You are planning a picnic. Given the number of children and adults attending the picnic, compute the number of full 1.0 lb packages of hamburger needed (you can only purchase full 1.0 lb packages) **[2 points]**. You know:

- Children eat one burger,
- That adults eat two burgers,
- That each burger weighs ¼ pound,
- Hamburger comes only in one pound packages.

Test case name (ex. "negative height", "typical conditions", etc.)	**Input data set for this test case**	**Explain why you chose this test case**

Solution should be based on this process (doesn't have to be included):

Input: #children>=0, #adults>=0 both integers
Process: #hamb=#children+2*#adults
LBhamb=#hamb/4
#packages=ceil(LBhamb)
Output: #packages>=0, integer

Test cases that should be included:
- valid / invalid input data,
- no children,
- no adults,
- more children than adults,

- more adults than children,
- equal number of adults and children,
- whole number of lbs needed,
- fractional number of lbs needed and round up.

4. Given the constant flow rate of a faucet into the sink in volume/sec, the volume of the sink, and the constant drain rate of the drain in volume/sec, determine if the faucet is left running when (if ever) the sink will overflow. Output when it will overflow in seconds, or a message stating it will not overflow. **[2 points]**:

Test case name (ex. "negative height", "typical conditions", etc.)	**Input data set for this test case**	**Explain why you chose this test case**

Needs to consider:

- valid / invalid input data,
- all inputs should be greater than zero (constant flow rate in, sink volume, constant drain rate),
- test a case for in rate being very small compared to out rate so no water fills at all,
- test the the case with two rates being equal,
- test the case with the in rate being greater than out rate.

Those three last cases should be tested on both small sinks and large sinks.

5. One of the more interesting statistics for comparing power hitters in baseball is slugging percentage. A hitter's slugging percentage is calculated as follows **[2 points]**:

slugging percentage = (singles + 2*doubles + 3*triples + 4*homeruns) divided by atbats

Test case name (ex. "negative height", "typical conditions", etc.)	**Input data set for this test case**	**Explain why you chose this test case**

Needs to consider:

- slugging - Need to verify all input values are integers, 0 or greater,
- atbats has to be >= total of all hits,

Then for testing you should test all combinations of each type of hit being zero or non zero. A lot of test cases are needed