# Claude's Custom Counters, Inc.

*** NOTE: DUE DATES MAY BE DIFFERENT FOR DIFFERENT SECTIONS ***

Background

Claude has built a booming business designing high-end custom counters for discerning customers. Claude's motto, *right angles are just wrong*, guides his every design. While most vendors deliver boring rectangular designs, Claude will only produce parallelogram counters having interior angels that are not 90˚. As a boutique designer for eccentric customers Claude was previously able to make all calculations in his head. However, demand for his custom designs recently went mainstream and he is now in serious need of automation. You must develop a computer program to determine how much one of Claude's custom countertops will cost.
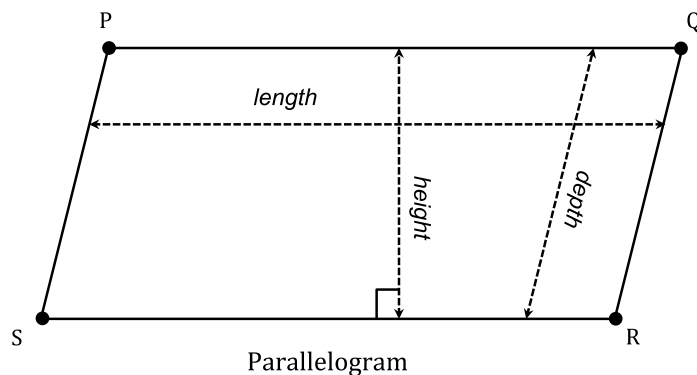
$$Area = length \times height$$

Parallelogram

Each countertop piece is cut from stone blocks. Materials currently offered are:

- Marble, at $92.99/sq ft installed
- Granite, at $78.99/sq ft installed
- Quartz, at $56.99/sq ft installed

The initial cost of a counter is based upon the area of material required for fabrication. Because of wastage when pieces are cut, we must add 26% to the area of the finished piece and then round that up to the nearest whole number.

Exposed edges can be finished by smoothing and polishing for $4.99 a linear foot.

Program Input

- The type of stone (first letter of the stone name)
- The length of the countertop
- The depth of the countertop
- The height of the countertop
- How many length edges and depth edges are to be finished

Each item of input requires validation. The minimum value for both length and depth is 5 feet. The maximum value for both length and depth is 25 feet. The length must be greater than or equal to the depth (if the length and depth are equal, the counter is a rhombus). The height must be between 58% and 80% of the shortest side:
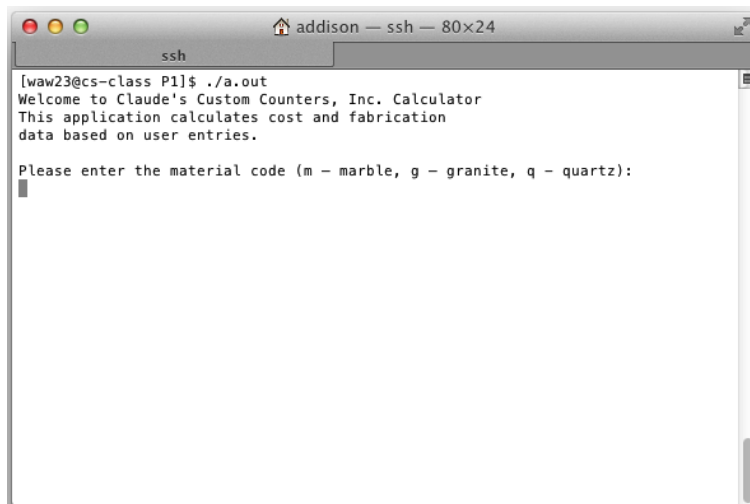
$$5.0 \leq length \leq 25.0$$
$$5.0 \leq depth \leq 25.0$$
$$depth \leq length$$
$$0.58\,(depth) \leq height \leq 0.80\,(depth)$$

The type of stone must be one of the three options specified above. The number of finished length edges must be an integer between 0 and 2. The number of finished depth edges must be an integer between 0 and 2. After each value is entered, the value shall be tested to ensure that it is valid and/or within the appropriate numeric range. If any entry is invalid an error message shall be displayed and the program will end. You are not required to test for data type errors. This means that if a character value is expected you may assume that the user entered a character, if a number is expected, you may assume that the user entered a number. For character values (such as the first letter of stone names) your software shall accept both upper case and lower case letters as valid. Your software shall also disregard any extraneous information typed after a valid entry. For example the lower case letter g would be valid for granite, the upper case letter G would be valid for granite, as would the entire name Granite (all characters after the G are ignored). If all values entered are valid, then your software shall make the required calculations and output the results specified below.

Program Output

- The length, depth, and height of the countertop
- The number of square feet of material needed to begin fabrication
- The cost of the stone
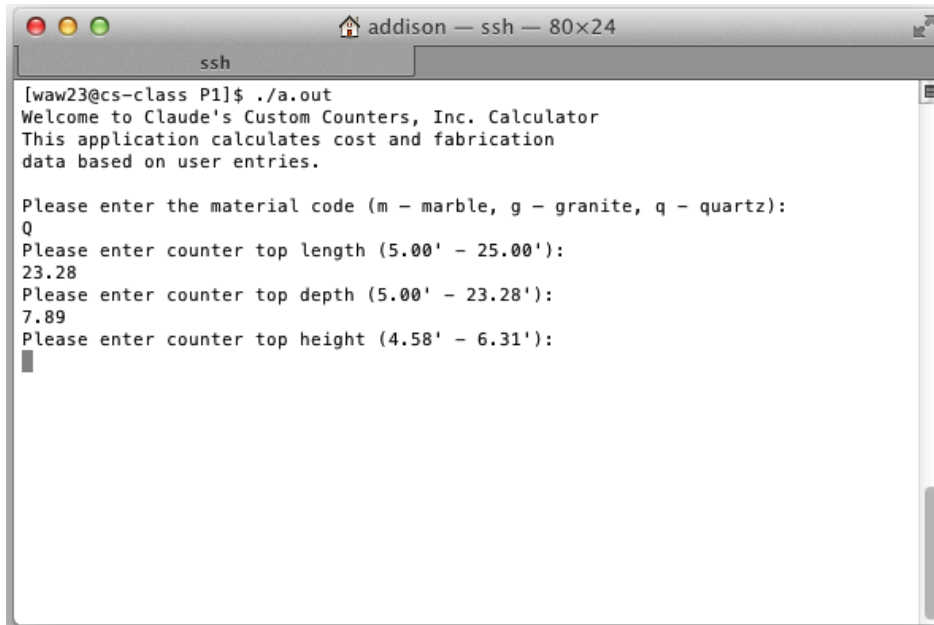- The cost of the edge finishing
- The total cost of the above

Screen Captures

Your software shall have "smart quotes" that inform the user of acceptable entries:

```
●●●                       🏠 addison — ssh — 80×24                    ↙↗
              ssh
[waw23@cs-class P1]$ ./a.out
Welcome to Claude's Custom Counters, Inc. Calculator
This application calculates cost and fabrication
data based on user entries.

Please enter the material code (m — marble, g — granite, q - quartz):
Q
Please enter counter top length (5.00' - 25.00'):
23.28
Please enter counter top depth (5.00' - 23.28'):
7.89
Please enter counter top height (4.58' - 6.31'):
▌
```

Your software shall accept upper case and lower case for character data entry. It shall also ignore any extraneous characters entered after a valid value is entered. Note in the screen capture below, the entry Granite is accepted. The upper case character G is stored as the user's entry. The remaining characters are ignored.

```
●●●                       🏠 addison — ssh — 80×24                    ↙↗
              ssh
[waw23@cs-class P1]$ ./a.out
Welcome to Claude's Custom Counters, Inc. Calculator
This application calculates cost and fabrication
data based on user entries.

Please enter the material code (m — marble, g — granite, q - quartz):
Granite
Please enter counter top length (5.00' - 25.00'):
12.54
Please enter counter top depth (5.00' - 12.54'):
8.32
Please enter counter top height (4.83' - 6.66'):
5.23

How many length edges should be finished and polished (0 - 2)
▌
```

Your software shall display clear error messages for any invalid entries. If an invalid value is entered, the program should exit after displaying the error message.
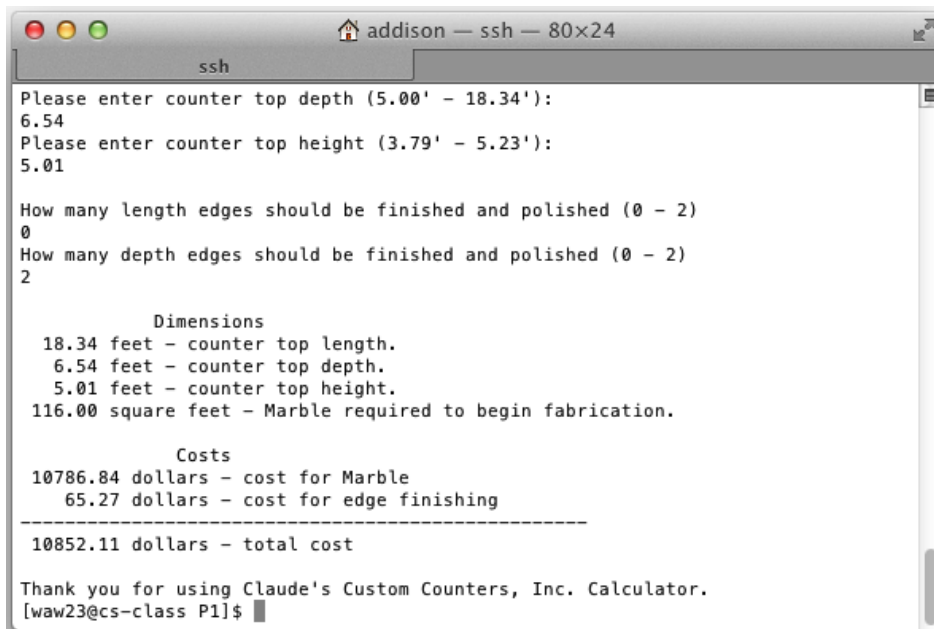
```
 ● ● ●                    🏠 addison — ssh — 80×24
              ssh
[waw23@cs-class P1]$ ./a.out
Welcome to Claude's Custom Counters, Inc. Calculator
This application calculates cost and fabrication
data based on user entries.

Please enter the material code (m — marble, g — granite, q — quartz):
R
The material selected is not a valid value.
Program will exit now, Goodbye...
[waw23@cs-class P1]$ ▌
```
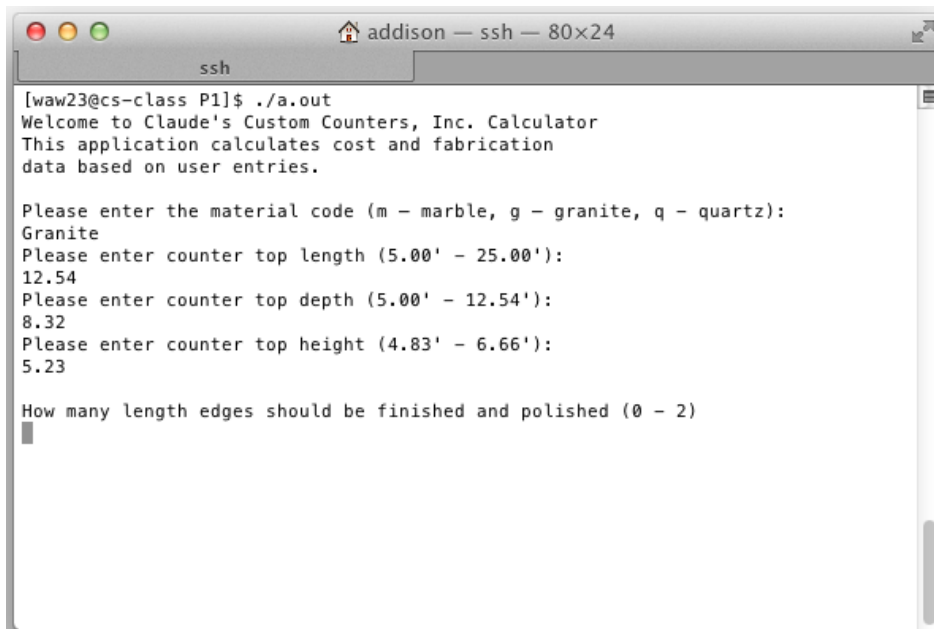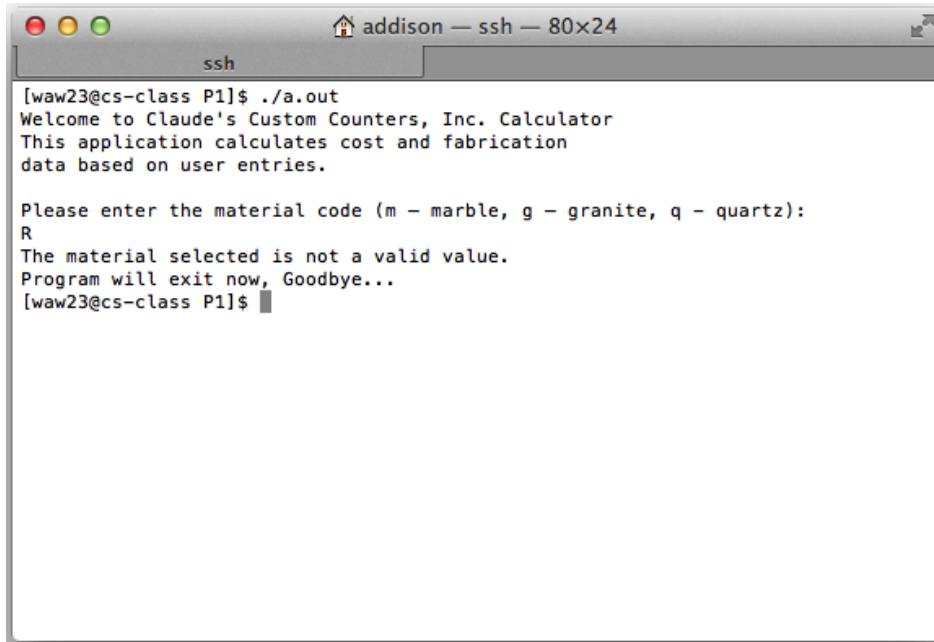
Your software shall display neatly formatted, accurately calculated output.

```
 ● ● ●                    🏠 addison — ssh — 80×24
              ssh
Please enter counter top depth (5.00' — 18.34'):
6.54
Please enter counter top height (3.79' — 5.23'):
5.01

How many length edges should be finished and polished (0 — 2)
0
How many depth edges should be finished and polished (0 — 2)
2

          Dimensions
  18.34 feet — counter top length.
   6.54 feet — counter top depth.
   5.01 feet — counter top height.
 116.00 square feet — Marble required to begin fabrication.

           Costs
 10786.84 dollars — cost for Marble
    65.27 dollars — cost for edge finishing
 ----------------------------------------------------
 10852.11 dollars — total cost

Thank you for using Claude's Custom Counters, Inc. Calculator.
[waw23@cs-class P1]$ ▌
```

Academic Integrity

This is an individual project and all work must be your own. Refer to the guidelines specified in the *Academic Honesty* section of this course syllabus or contact me if you have any questions.

Part A - Design Document   (Submit by deadline as per professor instructions)

Create a flow diagram which illustrates the flow of execution for your proposed algorithmic solution. (As discussed in class), Use standard flow diagram format: boxes, diamonds and arrows. You may use a digital tool (e.g. draw.io or Google Drawings) or you may draw the diagram by hand and digitize.

Please include enough detail so that the implementation should follow from the direct translation of the diagram. ***Specifically, each box (task) should be performable using 1 – 3 lines of C++ code***. When in doubt, add more detail. (Please also include constant and variable declarations / definitions needed. You may group many of these into one box.)
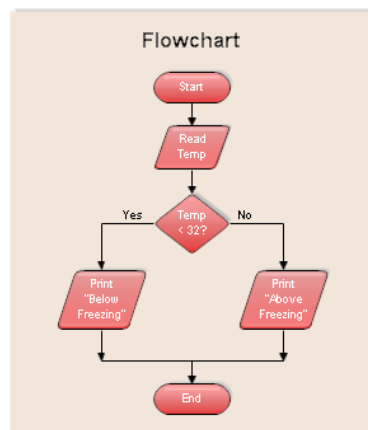
`Simple Example:`

Part A - Submission Details

Upload (as instructed by your professor) a .pdf file containing your design using the language described above. Use the following file name for your file: `<netID>P1.pdf` (replace `<netID>` with your netID and remove the angle brackets). Late submissions will be penalized heavily – see rubric for details. If you are late you may turn in the project to receive feedback but the grade may be zero. In general requests for extensions will not be considered.

Include the following Ethics Statement in the submission comments within Canvas, when submitting the Design:

COSC 051 <term-year>
  Project #1

  Author: <your name>

  In accordance with the class policies and Georgetown's
 Honor Code, I certify that, with the exception of the
  class resources and those items noted below, I have neither
  given nor received any assistance on this project.

 References not otherwise commented within the program source code.
  Note that you should not mention any help from the TAs, the professor,
   or info taken from the class textbooks.

Part B - Program Source Code (Submit by deadline as per professor instructions)

Important: Your output and input should be very similar to that shown in the sample output. Please ask for the input in **exactly** the same order shown and only request the same items shown - do not ask for any other input. This will assist in grading your program. Some content must also be included in your program **<u>exactly</u>** as specified.

Include the following comments at the start of your source code file:

```
/*
* <FileName>.<file extension>
 *
 *  COSC 051 <put year and term here>
 *  Project #1
 *
 *  Due on: <put due date here>
 *  Author: <your name>
 *
 *
 *
 *  In accordance with the class policies and Georgetown's
 *  Honor Code, I certify that, with the exception of the
 *  class resources and those items noted below, I have neither
 *  given nor received any assistance on this project.
 *
 *  References not otherwise commented within the program source code.
 *  Note that you should not mention any help from the TAs, the professor,
 *  or any code taken from the class textbooks.
 */
```

These comments must appear **exactly** as shown above. The only difference will be values that you replace where there are "place holders" within angle brackets such as `<netID>` which should be replaced by your own `netID`. For example, I would replace `<netID>P1.cpp` with `waw23P1.cpp`.

Part B - Submission Details

Upload (as instructed by your professor) a `.cpp` file containing your <u>source code</u>. Do **NOT** post your executable file. You should ensure that **<u>your source file compiles on the server</u>** and that the executable file runs and produces the correct output. Use the following file name for your file: `<netID>P1.cpp`. Late submissions will be penalized heavily – see rubric for details. If you are late you may turn in the project to receive feedback but the grade may be zero. In general, requests for extensions will not be considered.

Part A – Grading Rubric

Grade Standards - Missing: 0%, Poor: up to 50%, Fair: up to 67%, Good: up to 82%, Excellent: up to 99%, Perfect: 100%

Detailed Rubric (Code)			Correct Flow Diagram Format	Detail	Correctness	
	100.00	<-- TOTAL				100.00
Correct Flow Diagram Format	*30.00*	<--sub total				30.00
Boxes, Diamonds, and arrows	10.00					
Boxes: Tasks/Instructions clearly indicated	10.00					
Diamonds: conditions clearly indicated, branches clearly indicated	10.00					
Detail	*30.00*	<--sub total				30.00
Boxes: each task can be executed using 1 – 3 lines of code	15.00					
Diamonds: each condition can be executed using 1 – 3 lines of code	15.00					
Correctness	*40.00*	<--sub total				40.00
Flow and logic are correct	20.00					
Calculations are shown and are correct	20.00					

Part B – Grading Rubric

Detailed Rubric (Code)	100.00	<-- TOTAL			100.00
Code Quality and Formatting	*14.00*	<--sub total			
proper indentation					
good variable and constant names					
good use of constants (no "magic numbers" in calculations)					
good use of comments				**Code Quality and Formatting**	14.00
good use of vertical white space to separate code					
good use of horizontal white space to improve readability					
line length less than 100 characters					
User interface / data input	*17.00*	<--sub total			
outputs a brief greeting message					
outputs a brief greeting message					
outputs prompt for shape code					
outputs prompt for radius					
if appropriate for the shape, outputs prompt for height				**User interface / data input**	17.00
outputs prompt for material					
values entered by user are input into named variables of appropriate data type					
any extraneous characters entered after a valid entry are ignored					
error messages are clear and descriptive					
for character input, both uppercase and lowercase are accepted as valid					
Data validation algorithms	*20.00*	<--sub total			
all input data are validated to ensure they are valid and/or within limits					
prompts for data input are in reasonable order, test for errors and exit as soon as possible (don't make the user keep entering data if there has already been a fatal error)				**Data validation algorithms**	20.00
if any input data fail validation error message(s) are displayed					
processing terminates (gracefully) if any data fail validation, "abnormal" exits will be allowed (in moderation) for Project #1, but will eventually be prohibited					
Calculation algorithms	*37.00*	<--sub total			
surface area and volume are accurately calculated using the correct equations for each different shape				**Calculation algorithms**	37.00
manufacturing costs and sales price are accurately calculated for each shape/material combination					
Output	*12.00*	<--sub total			
outputs shape name and dimensions corresponding to the user's entries					
outputs calculated surface area and volume				**Output**	12.00
outputs all calculated cost and price values					
output is neatly arranged on screen and is consistent with the output shown in the example program					

Common Deductions (Code)		
Program does not compile ON THE CLASS SERVER (deduction varies depending on how bad, value listed is max)	-100.00	
Program compiles but has warnings ON THE CLASS SERVER (deduction varies depending on how bad, value listed is max)	-100.00	
Program crashes during execution ON THE CLASS SERVER (deduction varies depending on how bad, value listed is max)	-100.00	
Code uses any global variables	-40.00	
Filename does not follow conventions specified	-20.00	
Required comments and honor statement not included at start of file exactly as specified	-30.00	
Late penalty for each 15 minutes late	-2.50	