

Clayton Price

 Search this site[Courses](#) [Missouri S&T](#) [CS Department](#) [Help](#) [Guides](#)[Courses](#) > [CS 1570 \(formerly 053\) - Intro to Programming C++](#) > [Homework Assignments](#) > [2017](#) > [Spring](#) >

Assignment 03

Due: Friday, Feb. 10, 2016 at noon 100 points

For this assignment, you will submit a single C++ compilable file containing a program written in C++. Remember, to submit a file for this course electronically, from the directory in which the file resides type in at the UNIX prompt the command: `cssubmit 1570 section_letter assignment_number`. Be sure that only the file you want to submit is in that directory - make a directory for every assignment! The submit system will deliver every .cpp file in the current directory to me. Name your file a meaningful name and give it a .cpp extension since you will be compiling it. Also, make sure that you compile and run your program using the GNU (g++) compiler before submitting to make sure that it will work for the submit script.



Background: Poor Milhouse. Life just isn't fair sometimes. And Milhouse's life in Springfield is complicated further by his poor eyesight. But help is on the way! for our hero. You are going to write a program that will help kids like him diagnose their sight problems on-line. Your program will take inputs from the user, and it will generate "inputs" from a new device called the Photo-Honest Optic-Neuron Yadi Yadi Yadi (PHONEYYY), and use these values to produce a string of characters (the code) that the user can then take to a licensed optician who will interpret the results so they can grind lenses for working glasses. The user no longer has to drive to a competent optometrist, be evaluated by a licensed professional, diagnosed by a qualified expert, or even obtain sound medical advice! Who needs it....you have GLASS-O-MATIC working for you.

Specifications: Your program will ultimately output to the screen a code that is a sequence of letters that represents the user's condition. The user would write this down, and then take (email, phone, whatever) it to a lens grinder for new glasses. Start by prompting the user for his/her name, and then use that name in additional prompts and appropriate outputs. The first input for determination of prescription is from the user. They are to be prompted for a measure of their cornea (right eye only) in millimeters (mm). How the user accomplishes this task is not our problem. They can use a tape measure for sure. Based on this input, which should fall between 1 mm and 20 mm, the **first character** of the prescription code follows this table:



▪ 1 - 5 mm ==> "P"

- 6 - 10 mm ==> "B"
- 11 - 20 mm ==> "C"

For the **next two letters** of the prescription, you will start by using these formulas for calculating x:

$$x_1 = [(start * 7919 + 104729)] \bmod 101 + 150,$$

$$x_2 = [(x_1 * 7919 + 104729)] \bmod 101 + 150,$$

$$x = \text{average} (x_1 , x_2)$$

Now, "start" is determined by the results of the first input. If it is "P", start = 1; if "B", then start = 2; otherwise start = 3. Once x is determined,

If

- $150 \leq x \leq 200 \implies \text{"iN"}$
- $200 < x \leq 250 \implies \text{"Li"}$


To determine the **next part** of the code, the user is supposed to self-administer a 9-line Snellen chart - the one with all the **E**'s on it and you tell the direction of the **E**. So, your code should prompt for and enter the lowest line number where a mistake was made. (For example, if the user were to say the first **E** on the second line points up, they would then enter 2...assuming they got the top **E** right.) With this input, your code then determines the next part using:

- 1 ==> "N"
- 2 ==> "Ke"
- >= 3 ==> "P"

Finally, the last substring of the code is found as follows:

- if part #1 is "P" or "B" and part #3 is 1 ==> "d"
- if part #1 is "C" and part #3 is 1 ==> "ky"
- if part #2 is "iN" and part #3 is "P" ==> "h"
- otherwise ==> "ye"

Once the code for the prescription is completed, it should be output for the user in a user-friendly manner. As an example: if the first input is 12, the first substring is "C"; this results in the second part being "iN"; if the input for part 3 is 4, the third substring is "P"; and thus the fourth substring is "h". So the final code is "CiNPh". Try user inputs of 3 and 2.



Details: By next Monday, you will have been exposed to at least one kind of loop in c++. Use this new-found knowledge to write your code to insure that the user inputs acceptable values for the prompts. For example, if you prompt for cornea width, then a negative response or one greater than 20 should NOT be accepted; the user should be re-prompted for correct input. Do this for all applicable inputs (not for input of name!).



Remember: Use good coding standards and practices. This means **you should use constants!!** Think carefully about any quantity/variable/whatever that should be a constant....including strings and characters.

When you submit:

- first user input 8 (mm)
- second user input 1 (as in line #1)

As usual, don't hesitate to ask any question of your instructor if you have any. Please don't ask help from your cs 1580 (lab) instructor as this is not their assignment.

E-mail: price@mst.edu | Phone: 573-341-4491 | Fax: 573-341-4501 | Address: 325G Computer Science Building, Rolla, MO 65409-0350

[Report Abuse](#) | [Print Page](#) | [Remove Access](#) | Powered By [Google Sites](#)