CS 133U: C Programming

Assignment 2: Radon Levels

Academic Integrity

You may NOT, under any circumstances, begin a programming assignment by looking for completed code on StackOverflow or Chegg or any such website, which you can claim as your own. Please check out the Student Code of Conduct at PCC.

The only way to learn to code is to do it yourself. The assignments will be built from examples during the lectures, so ask for clarification during class if something seems confusing. If you start with code from another source and just change the variable names or other content to make it look original, you will receive a zero on the assignment.

I may ask you to explain your assignment verbally. If you cannot satisfactorily explain what your code does, and answer questions about why you wrote it in a particular way, then you should also expect a zero.

A half-life is the amount of time it takes for a substance or entity to fall to half its original value. Radon-222 has a half-life of about 4 days. Given Radon amount (These levels are measured in picocuries per liter (pCi/L)) as input, output the Radon level after 4, 8, and 16 days. You will be writing a C program to prompt a user for the amount of Radon detected, and output Radon level after 4, 8, and 16 days.

Purpose

Recall, when we design a program to solve a problem, we break it down into steps a computer can execute. The purpose of this assignment is to develop an algorithm (using both **pseudocode** and a **flowchart**), and a C program to calculate the Radon level after 4, 8, and 16 days, given the Radon level (in pCi/L) as input.

After completing this assignment you will be able to:

- Correctly complete the Assignment Design Document that will be used for all assignments in this course
- Express an algorithm as pseudocode and a flowchart
- Use C syntax to read in an input from the user, store the input into an appropriately typed variable, perform arithmetic calculations, and produce output

Task

- ❏ Open the Algorithmic Design Document, make a copy, and follow the steps to create your algorithm
- ❏ You must express your algorithm as **pseudocode** and/or a **flowchart**
- ❏ Your program must read the Radon level (in pCi/L) as input
- ❏ Your program must output the Radon level after 4, 8, and 16 days, given the input Radon levels
    - ❏ Note: The average indoor radon concentration for America’s homes is about 1.3 pCi/L. EPA (Environmental Protection Agency) recommends homes be fixed if the radon level is 4 pCi/L (picocuries per liter) or more.
- ❏ Output each floating-point value with four digits after the decimal point, which can be achieved as follows:
    - ❏ printf("%0.4lf", yourValue);
- ❏ **Use only the concepts we have learned so far.**

Criteria for Success

- ❏ Test your program using the following sample runs, making sure you get the same output when using the given inputs (in **blue**):

```
Welcome to my Radon Level Calculator!

Enter the amount of Radon detected: 6

Your Radon level:

After 4 days: 3.0000 pCi/L
After 8 days: 1.5000 pCi/L
After 16 days: 0.3750 pCi/L
```

```
Thank you for using my program!
```

```
Welcome to my Radon Level Calculator!

Enter the amount of Radon detected: 9

Your Radon level:

After 4 days: 4.5000 pCi/L
After 8 days: 2.2500 pCi/L
After 16 days: 0.5625 pCi/L

Thank you for using my program!
```

- ❑ Complete zyBooks section **CS133U 2. Variables/Assignments** activities.
    - ❑ Refer to zybooks, **CS133U 2. Variables/assignments, Section 2.7** for floating point numbers.

  The new link should be as below*
    - ❑ Refer to zybooks, **CS133U 2. Variables/assignments, Section 2.7** for floating point numbers.
- ❑ Ensure required coding conventions for this assignment are met:
    - ❑ Floating point data types must be used
    - ❑ 0.4lf formatting is required
    - ❑ No gotos or breaks are allowed.
    - ❑ Only concepts from Weeks #1-2 can be used.
- ❑ Complete all sections of your Algorithmic Design Document.
    - ❑ Include **pseudocode** or a **flowchart** in part d of the design document.
    - ❑ Include at least one screenshot of your code successfully running under Sample Runs.
- ❑ Please open and compare your work with the grading rubric before submitting.
- ❑ Remember to follow all style guidelines.
- ❑ Download your Algorithmic Design Document as a PDF (File -> Download -> PDF), rename it to `a02.pdf,` and upload it to the D2L assignment by **Wednesday**.
- ❑ Upload your `a02.c` C source file to the D2L assignment by **Sunday**.
- ❑ Do your own work. Consult the syllabus for more information about academic integrity..