X414.20 Fundamentals of Software Development

Coding Lab 5 – Looping

Please complete these lab steps only while in the Zoom meeting.

It is easiest to print these instructions so that you can refer to them during the lab. Your screen will be busy enough with both Zoom window and your Amazon Workspace window.

Task 1 – Change the interactive Blood Pressure program to prompt for multiple users and blood pressures

Two weeks ago, you wrote a program that read in a name and age and gave a blood pressure advisement. This week you will change that program so that it does this repeatedly, until the user indicates no more.

The original blood pressure program prompted the user for a name and an age, and then it displayed a short advisory report. In this new interactive version, it will do the same, but then it will prompt the user if they want to stop. For this program, do this without void functions.

The source code should be stored under: **c:\class\lab5task1.c**.

The sample dialog is:

```
Blood Pressure Processing Program

This is just a guideline program.

-----
Please enter patient name: Roberta Janss
Please enter your age: 22

Roberta Janss, age 22 -- Warning level is 131.0 mm Hg or higher

Are you finished with processing (y/n)? n

-----
Please enter patient name: Jack Pawlazi
Please enter your age: 50

Jack Pawlazi, age 50 -- Warning level is 145.0 mm Hg or higher

Are you finished with processing (y/n)? y

Thank you. You processed 2 patients.
```

Some things to think about:

1. This is an interactive program. What type of loop would be most appropriate here?
2. Because you are counting the number of patients, what do we need to do? Variables? Initialization?

Keep working on this until you think it is correct. Then **Ask for Help** so that I can check both your pseudocode design and your C program (for each student). **Do not proceed until I have checked and approved your program.**

Task 2 – Add more to the summary

In this exercise, you will take the Task 1 program and do what is necessary so that two additional lines are added to the summary. These lines will show the average age and the average blood pressure warning level.

The sample dialog is the same as before, but the summary section will look like this:

```
Thank you. You processed 2 patients.
Average age: 36.0
Average blood pressure warning level: 138.0
```

The source code should be stored under: **c:\class\lab5task2.c**.

Some things to think about:

1. What extra variables will you need?
2. Where will you be doing the math necessary to determine these averages.

Keep working on this until you think it is correct. Then **Ask for Help** so that I can check both your pseudocode design and your C program (for each student). **Do not proceed until I have checked and approved your program.**

Task 3 – Change this program into a file input program

In this task, you will change this to a file input version. The input file will be called **c:\class\bpdata.txt**. and you will create it manually. The source code should be in **c:\class\lab5task3.c**. The output file will be stored in **c:\class\lab5task3out.txt**.

Input file will be:

```
Roberta Janss     22
Jack Pawlazi      50
Gary Lee          42
Marj Fariss       67
```

The names above start in column 1 and the ages start in column 18.

The input file could have **any number of lines of names and ages**. Names will be no more than 16 in length. The maximum age is 140 and is always a whole number.

The sample output for the above would be:

```
Blood Pressure Report

Patient Name            Age     Bp Warning Level

Roberta Janss            22           131.0
Jack Pawlazi             50           145.0
Gary Lee                 42           141.0
Marj Fariss              67           153.5

Average                  45           142.6
```

The Patient Name header (and the names below) starts in column 1. The "e" in Age and the last digit of each ages is in column 26. The "BP Warning Level" header starts in column 32. The Bp Warning Levels last digit (the decimal digit) is in column 42.

Some things to think about:

1. This is a file input program based on end of file. What type of loop structure should we be using here?
2. This is a columnar report, so think again about what I said in the lecture about your formats. What should they NOT be.
3. Think about how you can use your Dev C++ editor to determine column positions and column widths (if you were copying these from a Word or HTML file).
4. Once again, think about the extra variables you are going to need to help you process the averages.

Keep working on this until you think it is correct. Then **Ask for Help** so that I can check both your pseudocode design and your C program (for each student). **Do not proceed until I have checked and approved your program.**

Task 4 – Submit your files in Canvas (breakout session)

1. Close Dev C++.
2. Re-open your Google Chrome. Be sure to choose Chrome from inside your AWS, not the one on your physical computer.
3. You should be in Canvas. If not, browse there and login.
4. Go to **Modules**. Then **Module 5**. If it doesn't display automatically, click the wedge next to it to open things up. Click on **Coding Lab – M5** – **Looping**.
5. Click the big **Submit Assignment** button.
6. Under **File Upload**, click **Choose File**.
7. In the File name box, type C: and press ENTER. Then double-click the class folder to open it.
8. For each of the C files you generated tonight, highlight each of them (one at a time), click it and then click **Open** (or you could simply double-click the file).
9. The file should appear next to the **Choose File** button.
10. Do this for each additional C file from tonight plus your single output file. You will need to choose **+Add Another File** for each of these. **There should be a total of 4 files**: 1 output files, 3 source code files. No need to submit the data file.
11. You may add a comment to the **Comments…** area if you wish.
12. Then click **Submit Assignment**. The button will change to **Submitting** and then you will be returned to the Coding Lab screen. Assignment should be marked as Submitted! In the upper right corner.
13. Later, to check your score, click on **Grades** in the second-level menu on the left.
14. If I haven't graded your program yet, you will see an icon. Other, you will see a score. You can click on the name of the assignment to see my comments and to issue further comments or respond to mine.
15. Coding Labs will typically receive a score of 5 if you completed the lab and your submission was substantially correct. You may receive a score of 4 if there are some problems, and 3 if it is incomplete. You will receive a 0 if you did not participate in the lab. Ignore any "Total" scores that Canvas may show. They are meaningless.

Congratulations on completing your fourth Coding Lab!!! Please exit the breakout session to rejoin the entire class for final announcements.