

Week 7 Lab

This Week

This week, you will work on the third version of the calc program from the calc backlog. You will attempt stories 12-14. (**Tip:** these programs are difficult. You should try to write recursive functions to implement the backlog. Start early and think hard. The thinking will be a big benefit for the rest of the class and the rest of your life. The functions become increasingly difficult so do them in order.)

You will need to download **wk3-calc.zip** into your working directory using `wget sp.nathanielgmartin.com/wk3-calc.zip`. When you unzip it in your working directory, it will produce the same files that you found in wk2-calc. However, `calc.c` will contain the stubs needed for stories 12-14. You will need to the functions you wrote last week into `calc.c` in wk2-calc because the new program builds on the old program.

Be careful: it is very easy to overwrite files in Unix. Copy last weeks work into a safe place before you begin working, to avoid copying over you work:

1. Create a new directory in your working directory called `old_week2`. You will have old copies of all of your directories if you do it this way.
2. Copy the directory you demo this week into `old_week2`.
3. Work only in `wk3_calc`. You can always download a new untouched copy, but you cannot restore you work if you lose it.

Another good practice is to keep the most recent version of your files in a separate directory. For example, if you need to quit for the day. May a copy of your directory in a new place. For example, the first time, you might use the command `cp -r calc calc.1` This command will copy all of the files in the directory `calc` in a newly created directory called `calc.2`. The next time, you can do `cp calc calc.2` This way you will have a copy of all of the files.

It is best to create a new directory with the current files in it every time you quit for more than a few hours and each time you complete a story. You can also use the directory name to give you a hint of what it contains. Such as `cp -r calc calc.4.story1` if you complete story 1 in the fourth session.

Demo

Today you will be demoing the second set of stories from the calculator backlog. You will demo stories 6-11 implementing arithmetic. You will get 15 marks for each story completed correctly.

Marks (90 marks total)

- Story 6 (15 marks)
 - Proper formatting (5 marks)
 - Compiles correctly. (i.e., without errors or warnings) (5 marks)
 - Prints “Enter integer> ” (2 marks)
 - Prints “You entered 1” if you type 1. (2 marks)
 - Prints newline (1 mark)
- Story 7 (15 marks)
 - Proper formatting (2 marks)
 - Compiles without errors or warnings. (3 marks)
 - Prints “a is not an integer, try again> ”, if you enter 'a'; (2 marks)
 - Prints “1.2 is not an integer, try again> ”, if you enter “1.2”; (2 marks)
 - Prints “1a is not an integer, try again> ”, if you enter “1a”; (2 marks)
 - Prints “You entered> 2”, if you enter “02”; (2 marks)
 - Prints “You entered> 20”, if you enter “020”; (2 marks)
- Story 8 (15 marks)
 - Proper formatting (2 marks)
 - Compiles without errors or warnings. (3 marks)
 - Test works as as described in backlog. (4 marks)
 - Add function implements according to declaration. (3 marks)
 - Add function called in printf statement. (3 marks)
- Story 9 (15 marks)
 - Proper formatting (2 marks)
 - Compiles without errors or warnings. (3 marks)
 - Test works as as described in backlog. (4 marks)
 - Subtract function implements according to declaration. (3 marks)
 - Subtract function called in printf statement. (3 marks)

- Story 10 (15 marks)
 - Proper formatting (2 marks)
 - Compiles without errors or warnings. (3 marks)
 - Test works as as described in backlog. (4 marks)
 - Multiply function implements according to declaration. (3 marks)
 - Multiply function called in printf statement. (3 marks)
- Story 11 (15 marks)
 - Proper formatting (2 marks)
 - Compiles without errors or warnings. (3 marks)
 - Test works as as described in backlog. (4 marks)
 - Division function implements according to declaration. (3 marks)
 - Division function called in printf statement. (3 marks)