Lab 1 - Setup, Strings

CS 251, Fall 2024

Copyright Notice

Labs in CS 251

Labs are intended to be completed **during** your lab section. You aren't expected to look at this before your lab section.

For labs in CS 251, you must work in pairs. The main goal of lab is to act as a bridge between lecture and exams, and lectures and projects. To accomplish this, most labs will involve some theoretical or code-reading questions, and an "intermediary" programming task of some kind. Since labs are time-limited, all labs are collaborative, and your lab TAs will be *extremely* helpful. **If you get stuck, you should immediately ask for help rather than floundering**. Lab is only 2 hours, and we want you to optimize the time you spend for maximum learning.

At the end of the lab, you **must** call over a member of course staff to "check off" that you were present at lab.

Personal Computer Setup

If you have not done the setup, follow the instructions in [FA24] CS 251 Software Setup so that your computer is set up to work on CS 251. **Both** partners should follow these instructions.

Depending on your computer, this may not finish during lab. That's fine, as long as one member of your group is able to get started writing code.

Welcome Survey

Once you're done with software installation, complete our Welcome Survey.

VSCode + Terminal Presentation

At some point during the lab, your TAs will give a presentation on using the terminal and VSCode. If you aren't familiar with either one of these tools, we recommend waiting until the presentation before beginning the programming task.

> During the presentation, you should follow along on your own computer! We recommend taking notes. Using our tools effectively is an important skill.

Programming

lab01-starter.zip

In lab, we ask that, as much as possible, you "pair program". Pair programming is a style of programming in which two programmers work **collaboratively** on one design, algorithm, or piece of code. The intent is that the two partners "act as one": one person "drives" (has control of the keyboard) and the other "navigates" (reviews and actively questions the code). Traditional pair programming happens **with one computer** and partners **switch roles periodically**.

The navigation role is **not** just to passively watch the driver write code. Their job is to think about design and strategy: where is the development going? is there a potential bug? is there something we should design differently? The navigation role should also **not** just be dictating code to the driver.

Importantly, pair programming is not dividing the work into two parts, with each student doing half.

Now for your task: in `lab1.cpp`, implement the two functions:

- `numWordsIn`, which counts the number of words in a given sentence.
- `main`, which prompts the user for a sentence, then outputs the number of words in that sentence.

There are many possible approaches here! Use the commands `make run_lab1` to run your program's `main`, and `make test_lab1` to run the test suite. Remember, ask a TA if you're having trouble; it's what we're here for.

Deliverables

To record your attendance at the lab, please show a TA:

- Welcome survey completion
- Optionally, your VSCode setup
- Your group’s implementation and passing tests