



Computer Science 381 Programming Unix in C

The College of Saint Rose
Winter Immersion 2016

Lab 2: C Basics

Due: Wednesday, December 23, 2015

In this lab you will gain more experience with the Unix tools and techniques from last week, but the focus is on some of the basics of C programming.

Before you start, create a directory on a Unix or Mac system to contain your work for this lab, and open a file `lab2.txt` where you will place your answers for this lab's questions.

C Basics

You may work on any C-capable Unix-like system (such as a lab Mac, mogul, or your own Unix-like system).

If you have not already done so, finish reading Chapter 1 of K&R.

? Question 1:

Consider any C program that uses the `printf` function. What happens if you leave out the `#include <stdio.h>` line? Explain briefly. (2 points)

? Question 2:

Suppose one of the programs from the running example about conversion from Fahrenheit to Celsius is to be modified to print conversions for every degree Fahrenheit from -100 to 1000 and you wish to print 3 digits after the decimal point for each Celsius temperature. What `printf` function call could be used to print this so that the printed temperatures are aligned nicely? (2 points)

Practice Program:

Write a program `charcount.c`, similar to that in Exercise 1-14 on p. 24 of K&R. Your program should deal only with numeric and alphabetic characters, and should treat upper or lower case characters as equivalent. You also need not print a histogram, you can just print a table of the results. Like the example program on p. 22, your program should read characters from the standard input and print the results to standard output. (10 points)

To run your program, we'll use Unix redirection like we did for the previous lab's output captures. Those used only output redirection with `>` at the end of the command line. We can also perform *input redirection* with `<` after our program name. This lets you run a program that expects input from the keyboard but instead that input will come from the file. For this case, copy the file on mogul in `/home/cs381/labs/c-basics/whosonfirst.txt` to your directory for this

lab. Then run your practice program, redirecting its input from `whosonfirst.txt` and its output to a file `whocounts.txt` with a command such as

```
./charcount < whosonfirst.txt > whocounts.txt
```

Output Capture:

| `whocounts.txt` for 2 point(s)

Practice Program:

| Write the program in Exercise 1-19 on p. 31 of K&R. Call your program `revlines.c`. (10 points)

Note: the `getline` function is now defined in `stdio.h`. (check text for what it was previously)

Also run this program on the `whosonfirst.txt` file, this time saving the output in `whorev.txt`.

Output Capture:

| `whorev.txt` for 2 point(s)

Question 3:

| Briefly describe the major differences between arrays in C, as presented in Chapter 1 of K&R, and arrays in Java. (2 points)

Programming Assignment

Write the program in Exercise 1-20 on p. 34 of K&R. Call your program `detab.c`. Use a defined constant for the value n , the number of columns between tab stops. Include This program is worth 20 points, broken down as shown at the end of this document.

Reference solutions to all programs are available on mogul in `/home/cs381/labs/c-basics`.

Submission

Please submit all required files as email attachments to terescoj@strose.edu by Wednesday, December 23, 2015. Be sure to check that you have used the correct file names and that your submission matches all of the submission guidelines listed on the course home page. In order to email your files, you will need to transfer them from mogul to the computer from which you wish to send the email. There are a number of options, including the `sftp` command from the Mac command line.

Grading

This lab is graded out of 50 points.

Grading Breakdown	
Lab questions and output captures	10 points
Practice program <code>charcount.c</code> correctness	10 points
Practice program <code>revlines.c</code> correctness	10 points
<code>detab.c</code> correctness	10 points
<code>detab.c</code> design	3 point
<code>detab.c</code> documentation	3 points
<code>detab.c</code> style	3 point
<code>detab.c</code> efficiency	1 point
Total	50