CS 240 Summer 2017

Lab 2: Pointers, passing by value vs. reference, and 1-D arrays (205 pts)

Due: 06/27/2017 (Tue), 11:59 PM

Objective

The objective of this lab is to practice using pointers as they are commonly used in passing by reference and manipulation of 1-D arrays.

Reading

Read chapters 2 and 3 from K&R (textbook).

Lab 2 Code Base

The C code base for lab2 is available as a tarball

/u/data/u3/park/pub/cs240/lab2.tar

on the Linux machines in LWSN B148. Copy it to your working directory and untar it using

% tar xvf lab2.tar

Under lab2, you will find subdirectories that contain versions of code discussed in class and additional code needed for Lab 2.

Problems [205 pts]

Problem 1 (15 pts)

Compile and run the program in v1 multiple times. What output do you observe, and why?

Problem 2 (15 pts)

Explain why after calling changeval1() in v2 the value of s printed on stdout continues to be 5 (not 3).

Problem 3 (20 pts)

Explain why after calling changeval() in v3 the value of x printed on stdout changes to 3. What happens if we change the assignment "*a = 3" in changeval() to "a = 3"? Explain what happens upon compilation and at run-time when executing a.out.

Problem 4 (15 pts)

Modify the code in v3 so that the function changeval() is placed in a separate file changeval.c. Put the declaration of the function in main() in a separate header file main.h. Verify that everything works as it should by (1) compiling the two .c files separately using the -c option, (2) running gcc on the resultant .o files, and (3) executing a.out. If you don't like the filename a.out for the executable, what are two methods for changing it to a different name?

Problem 5 (30 pts)

Compile and run the code in v4. Explain the output produced by the program. Suppose we add the assignment "z = &y" to the end of main(). How must z be declared in the program? How do we print the value of x using z? Make the changes to main.c and submit the revised code after checking that it compiles and runs correctly.

Problem 6 (20 pts)

Compile and execute the code in v6. What is the meaning of segmentation fault? What causes the fault to arise in main() when executing a.out?

Problem 7 (20 pts)

Describe how the 1-D integer array s[3] in v7 is laid out in main memory (i.e., RAM).

Problem 8 (15 pts)

Why does compiling and running the program in v8 result in segmentation fault? Make changes to main.c so that a segmentation fault does not arise. Submit the revised code of v8.

Problem 9 (40 pts)

What is the silent (or hidden) run-time error in v9? What programming practice reduces the likelihood of introducing such run-time bugs? What happens if you change the limit of the third for-loop from 6 to 7? What happens if you run gcc with option -fno-stack-protector? Explain what is going on.

Problem 10 (15 pts)

What does the layout of character array a[6] in v10 look like in memory? Use the string processing library function strcpy() (check its usage using man) to assign the string "abcde" to a[6] and print its value. Can a[6]

be used to store the string "abcdef"? Explain.

Bonus Problem (20 pts)

The BUGS section of strcpy()'s man pages warns against a potential bug of using strcpy() incorrectly. How is this related to the silent run-time error of Problem 9?

Turn-in Instructions

Provide written answers in a separate writeup, lab2.pdf and place the pdf file in lab2/.

Electronic turn-in instructions:

i) Any written answers submit your write-up as a pdf file, lab2.pdf. You can use your favorite editor subject to that it is able to export pdf files which most editors, including freeware, do. The TAs need to be able to spend their time evaluating the content of your submission, not switching between editors or hunting down obscure document formats which wastes valuable time and is in no one's interest. Thus only pdf files are accepted and other formats receive zero credit, even if a pdf file is submitted after the due date.

ii) For code submissions, follow the directions specified in the problems.

iii) We will use turnin to manage lab assignment submissions. Rename lab2 to mylab2. Go to the parent directory of your mylab2 directory and type the command

turnin -v -c cs240 -p lab2 mylab2 -v

Please note the assignment submission policy specified in the course home page.

Back to the CS 240 web page