**Working with Strings**

**Mitchell Wadle**

*Lab 3*

9/22/17

**Problem:**

In this week’s lab, we were working with strings and how to compare and print strings in different ways. Our problems were to first copy the source code from blackboard and add code to it. Instructions were commented into the code already so all we had to do was read the comments and add code as needed to make the program run correctly.

**Analysis:**

Problem 1: From problem 1, they had use test the program using a few different symbols to see if we could get the same result. The use of the ( ! ) symbol meant that that thing was going to get printed next. It doesn’t matter if the ( ! ) symbol is there or not, the if else statement runs and prints the result just fine. When you get to the end if a string has been entered that is not recognized by the if else statement then another string should be printed that says that the major entered was not recognized. This needs fixed because the printf statement that prints the line prints automatically even if a string is recognized. Within the if statement that reads if an unrecognized string has been entered, they had just flag in the () when they need to say flag == 0. This reads the integer value for flag and if its 0 then the statement will print if not then it won’t. We needed the value flag to determine if a recognized major was entered or not. Flag gets updated when a recognized major is entered but remains zero if an unrecognized one is entered.

Problem 2: Frist asked to add a printf that asked a user for a source string. The code would then take an input from the keyboard and store it in sourceString. Then we had to write a string that printed both strings together. The final task was to print the sourceString after it had been replaced by the destinationString.

Problem 3: had use add a printf for the sourceString and the destinationString. They then had use strcat to add the strings together instead of using 2 %s.

**Design:**

Problem 1: You first need a printf to ask for the users major. Then you need a scanf to take an input from the keyboard and save it to major. This then gets compared in the if else statement already written. The error is in the final if statement. It needs to see if flags is equal to zero which is written as (flag == 0) not just (flag). Once this is changed they code runs fine with no other errors.

Problem 2: First you needed to add the printf to ask for an input from the user. The next thing is a scanf that takes the input from the keyboard for sourceString. Next we need to make a printf that prints both the sourceString and the destinationString together. After that a strcpy is used to copy destinationString into sourceString. Finally we print just sourceString and will have changed to the string from destinationString.

Problem 3: You need to add a printf and a scanf that asks for a sourceString and save the string to sourceString. You then need another one that asks for destinationString and takes a string and saves it to destinationString. The strcat uses string concatenation to add one of the strings to the other one so there isn’t 2 %s.

**Testing:**

Problem 1: For testing I first added the printf and scanf and the following printf to make sure that everything was print correctly and that the scanf was reading correctly. Once that was done I ran it multiple times to see where the error was coming from, changing things every run. Eventually I fixed the problem and the code ran fine.

Problem 2: First added the printf and scanf then ran the code. The code doesn’t run properly until you add the final printf at the end that shows the need value for sourceString.

Problem 3: Run the code with the printf and scanf’s in there to make sure they’re taking data correctly. Then add the printf after the strcat to print the new value for sourceString.

**Comments:**

Lab was fairly easy. Only problem I had was with problem 1. I was trying a different approach to get the flag if statement to work properly and it wasn’t working. With problems 2 and 3 maybe be a little clearer on what exactly why want as far as a result. A little unclear as to what the difference was between some if the printf statements. In the end everything worked out.





