**CSC3320 System Level Programming**

**Lab Assignment 10 - Post-Lab**

Due at 11:59 pm on Friday, April 02, 2021

Purpose: Learn how to use the pointers to represent strings in C.

**Part 1:**

Write a function about string copy, the *strcpy* prototype "*char\* strcpy (char\* strDest, const char\* strSrc*);". Here *strDest* is destination string, *strSrc* is source string.

1) Write the function *strcpy*, don't call C string library.

#include<stdio.h>

char\* strcpy(char\* strDest, const char\* strSrc)

{

int i;

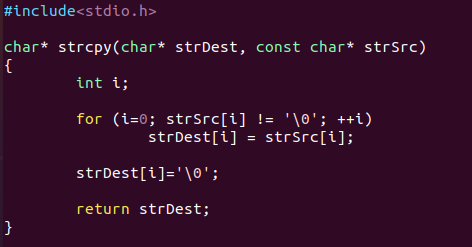
for (i=0; strSrc[i] != '\0'; ++i)

strDest[i] = strSrc[i];

strDest[i]='\0';

return strDest;

}



//Use main driver to test function.

int main()

{

char source[] ="Hello, World!";

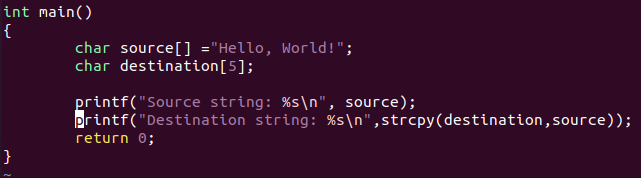
char destination[5];

printf("Source string: %s\n", source);

printf("Destination string: %s\n",strcpy(destination,source));

return 0;

}



Output:



2) Here *strcpy* can copy *strSrc* to *strDest*, but why we use *char\** as the return value of *strcpy*?

Char\* pointer is returned because the return type of function was char\* type and this address can be used by some other variable to point to destination. By returning strDest it will get easier to combine multiple classes from string library.

**Part 2:**

Write a program ***findStr.c*** that finds the "smallest" and "largest" in a series of words. After the user enters the words, the program will determine which words would come first and last if the words were listed in dictionary order. The program must stop accepting input when the user enters a four-letter word. Assume that no word is more than 20 letters long. An interactive session with the program might look like this:

Enter word: dog

Enter word: zebra

Enter word: rabbit

Enter word: catfish

Enter word: walrus

Enter word: cat

Enter word: fish

Smallest word: cat

Largest word: zebra

Hint: Use two strings named *smallest\_word* and *largest\_word* to keep track of the "smallest" and "largest" words entered so far. Each time the user enters a new word, use *strcmp* to compare it with *smallest\_word*; if the new word is "smaller", use *strcpy* to save it in *smallest\_word*. Do a similar comparison with *largest\_word*. Use *strlen* to determine when the user has entered a four-letter word.

***Questions:***

*1) Attach the source code of your C program into the answer sheet.*

#include<stdio.h>

#include<string.h>

int main()

{

char input[20];

char smallest\_word[20];

char largest\_word[20];

printf("Enter Word : ");

scanf("%s",input);

strcpy(smallest\_word,input);

strcpy(largest\_word,input);

while(strlen(input)!=4)

{

if(strcmp(input,smallest\_word)<0)

strcpy(smallest\_word,input);

else if (strcmp(input,largest\_word)>0)

strcpy(largest\_word,input);

printf("Enter Word : ");

scanf("%s",input);

}

/\*Last input (even if 4 characters long) is still an input even when program

\* stops accepting inputs after that 4 characters long input) \*/

if(strcmp(input,smallest\_word)<0)

strcpy(smallest\_word,input);

else if (strcmp(input,largest\_word)>0)

strcpy(largest\_word,input);

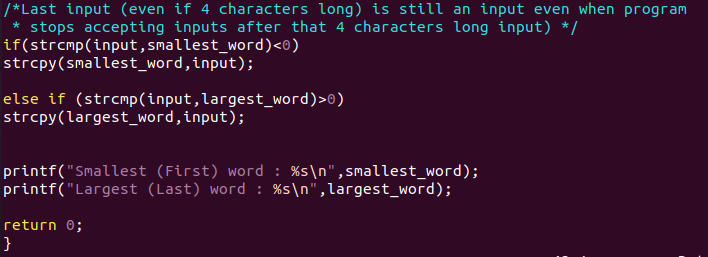
printf("Smallest (First) word : %s\n",smallest\_word);

printf("Largest (Last) word : %s\n",largest\_word);

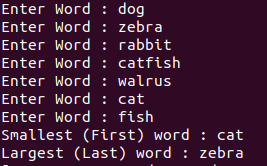
return 0;

}





*2) Run the C program, attach a screenshot of the output in the answer sheet.*



1

***Submssion***:

• Please follow the instructions below step by step, and then write a report by answering the questions and upload the report (named as

Lab10\_FirstNameLastName.pdf or Lab10\_FirstNameLastName.doc) to Google Classroom, under the rubric Lab 10 – Post Lab Assignment.

• Upload the C files findStr.c to the folder named “Lab 10 – Post Lab” in Google Classroom.

• Please add the lab assignment NUMBER and your NAME at the top of your filesheet.

2