**BIL 105E – Introduction to Scientific and Engineering Computing (C)**

**Spring 2015-2016**

**Homework 3 Report**

Assignment Date: 19.04.2016

Due Date: 01.05.2016, 23:00

Name: Kadir Emre Oto

ID: 150140032 CRN: 21831

**Introduction:**

As stated in description of homework, we are supposed to implement some character sequence (i.e., string) operations by using pointers and functions. Also this homework aims to consolidate the usage of pointers and improve programming skills.

**Development Environment:**

My project includes 1 C file, named 150140032.c and 1 executable file, named 150140032. I wrote my code Atom Editor on Mac OS, and compile it with GCC compiler in two different platforms (Mac OS and Linux). My command that I used to compile is

“gcc 150140032.c –o 150140032”.

**Variables:**

Defined in main function:

* choice: an integer to hold the user’s operation choice
* space: an integer to allocated memory space
* times: an integer to hold number of replacements
* \*ccs: a pointer to hold current character sequence
* \*insert, \*find, \*replace: pointers to hold some required strings
* begin\_index and end\_index: integers to hold range of indeces.

Defined in sub\_string:

* \*sub: a pointer to hold sub string

Defined in remove\_string:

* \*sub: a pointer to hold sub string
* diff: an integer to hold length of “remove”

Defined in replace\_string:

* diff: an integer to hold change of length caused by replacements
* \*temp: an char pointer to hold replacement string temporarily

\*Detailed information is in comment lines of the source code and the flowchart at next page.

**Used Funtions:**

int set\_ccs(char \*\*ccs)

char \*sub\_string (char \*ccs, int begin\_index, int end\_index)

char \*remove\_string(char \*\*ccs, int begin\_index, int end\_index)

int insert\_string(char \*\*ccs, char \*insert, int begin\_index)

int replace\_string(char \*\*ccs, char \*find, char \*replace)

**Conclusion:**

This was a beneficial homework that improved my programming skills. I fully understood the usage of char pointers and memory allocation. If it is necessary to change or allocate char sequence in other functions, the function should call by reference of pointer.

**FlowChart**

int function user\_menu {

print the menu;

read the user’s choice;

return choice;

}

int function set\_ccs{

allocate the css size of 81 char initially;

read the css from user;

reallocate the css;

If allocation is unsuccessful

return -1;

return allocated space;

}

char\* function sub\_string{

If begin\_index and end\_index are out of range

return NULL;

define and allocate \*sub just enough;

for (i = 0; i <= end\_index - begin\_index; i++)

\*(sub+i) = \*(ccs+i+begin\_index);

return sub

}

char\* function remove\_string {

diff = end\_index - begin\_index + 1;

get \*sub using sub\_string function;

if (sub == NULL) return NULL;

for (i = end\_index+1; i <= strlen(\*ccs); i++) {

// slide the css to left

\*(\*ccs+i-diff) = \*(\*ccs+i);

}

return sub

}

int function insert\_string{

expand the size of css just enough;

for (i = ccslen; begin\_index <= i; i--){

// slide the character to right

\*(\*ccs+i+inslen) = \*(\*ccs+i);}

for ( i = 0; i < inslen; i++){

// add the characters in insert into the ccs

\*(\*ccs+i+begin\_index) = \*(insert+i);}

free allocated space for insert;

return new length of ccs;

}

int function replace\_string {

int times = 0;

for (i = 0, j = 0; i < lenccs; i++){

diff = times \* (lenreplace - lenfind);

If i'th char of ccs is equal to j'th char of find

j += 1;

Else{

I -= j;

j = 0;}

if j == lenfind, so there is a match{

remove the “find” string;

insert the “replace” string;

times += 1;

j = 0;

}

}

free allocated space for find;

free allocated space for replace;

return times;

}