**SSN COLLEGE OF ENGINEERING (Autonomous)**

**Affiliated to Anna University**

**DEPARTMENT OF CSE**

**UCS 1211 PROGRAMMING IN C LABORATORY**

**A4: String Operations in C** ========================================================

**REGISTRATION NO: 185001112**

**NAME: PRATHYUSH S**

**CLASS: CSE-B (SEMESTER-2)**

**========================================================**

**1. Implement any three of the following functions. Write the required functions such as strlen(str), isUpper (ch), toLower (ch), toUpper (ch) without using library functions. You are encouraged to implement all the functions.**

**a. strcat (str1, str2) Appends str2 to str1**

**b. strncpy (dest, src, n) Copies up to n characters from src to dest string c. strchr (str1, ch) Scans the string str1 for the first occurrence of the character ch and returns the position**

**d. strset (str1, ch) Sets all characters in the string str1 to the character ch**

**e. strcmpi (str1, str2) Compares str1 and str2 ignoring the case sensitivity and returns -1/0/1**

**f. strncat (dest, src, n) Copies atmost n characters of src to the end of dest string**

#include<stdio.h>

void concat(char a[],char b[]);

int len(char a[]);

void append();

void copy();

void cp(char a[],char b[],int m); //function prototypes

void find();

void replace();

void endcopy();

void cpend(char a[],char b[],int m);

void compare();

void tolower(char a[]);

void main()

{

int o,count=1;

while(count!=0)

{

printf("Enter\n1.String Append\n2.String Copy\n3.Find Character\n4.Find a character and replace\n5.Compare 2 Strings\n6.Copy Sting 2 to end of Stirng 1\n7.Exit\n");

scanf("%d",&o);

//switch case

switch(o)

{

case 1:

append();

count=1;

break;

case 2:

copy();

count=1;

break;

case 3:

find();

count=1;

break;

case 4:

replace();

count=1;

break;

case 5:

compare();

count=1;

break;

case 6:

endcopy();

count=1;

break;

case 7:

count=0;

break;

default:

printf("Invalid input\n");

}

}

}

//a

void append()

{

//to read 2 strings

int n,m;

printf("Enter the number of elements in first string:\n");

scanf("%d",&n);

printf("Enter the number of elements in second string:\n");

scanf("%d",&m);

char s1[n],s2[m];

printf("Enter the first string: ");

scanf("%s",s1);

printf("Enter the second string: ");

scanf("%s",s2);

//printf strings

printf("String 1: %s\nString 2: %s\n",s1,s2);

concat(s1,s2);//function call

printf("Appended String : %s\n",s1);

}

void concat(char a[],char b[])

{

int i,len1,len2,j;

len1=len(a);//length of a

len2=len(b);//length of b

j=0;

//loop to concatenate 2 strings

for(i=len1-1;i<(len1+len2);i++)

{

a[i]=b[j];

j++;

}

}

//function to find string length

int len(char a[])

{

int i=0;

while(a[i]!='\0')

{

i++;

}

return (i+1);

}

//b

void copy()

{

int n;

int m;

//read string

printf("Enter the number of elements:\n");

scanf("%d",&n);

char a[n],b[n];

printf("Enter the string:\n");

scanf("%s",&a);

//read number of char to be copied

printf("Enter the number of characters to be copied:\n");

scanf("%d",&m);

cp(a,b,m);//function call

printf("Copied String is : %s\n",b);

}

//function to copy strings

void cp(char a[],char b[],int m)

{

int i;

for(i=0;i<m;i++)

{

b[i]=a[i];

}

b[m]='\0';

}

//c

void find()

{

int i,n,count;

//to read a string and character to find

printf("Enter the number of elements:\n");

scanf("%d",&n);

char a[n],b;

printf("Enter the string:\n");

scanf("%s",&a);

printf("Enter the character to be found : ");

scanf(" %c",&b);

//to find the postion of the string

for(i=0;i<n;i++)

{

if(a[i]==b)

{

printf("%c is at %d index\n",b,i);

count=0;

break;

}

else

{

count=1;

}

}

if(count==1)//if that char is not found

printf("Not Found\n");

}

//d

void replace()

{

//to read a string

int i,n,count;

printf("Enter the number of elements:\n");

scanf("%d",&n);

char a[n],b;

printf("Enter the string:\n");

scanf("%s",&a);

//to read the replacement character

printf("Enter the replacement character : ");

scanf(" %c",&b);

//loop to replace the elements with char

for(i=0;i<n;i++)

{

a[i]=b;

}

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

}

//e

void compare()

{

//to read a string

int n,m,i,min,count;

printf("Enter the number of elements in first string:\n");

scanf("%d",&n);

printf("Enter the number of elements in second string:\n");

scanf("%d",&m);

char s1[n],s2[m];

printf("Enter the first string: ");

scanf("%s",s1);

printf("Enter the second string: ");

scanf("%s",s2);

tolower(s1);//to convert s1 to lowercase

tolower(s2);//to convert s2 to lowercase

printf("String 1: %s\nString 2: %s\n",s1,s2);

min=(n<m)?n:m;//to find min

//to compare the strings

for(i=0;i<min;i++)

{

if(s1[i]<s2[i])

{

count=-1;

break;

}

else if(s1[i]>s2[i])

{

count=1;

break;

}

else

count=0;

}

if(count==-1)

printf("String1 is smaller\n");

else if(count==1)

printf("String1 is larger\n");

else

printf("Both strings are same\n");

}

//function to conver to lowercase

void tolower(char a[])

{

int i,k,num;

k=len(a);

for(i=0;i<k;i++)

{

num=a[i];

if((num>64) && (num<91))

{

a[i]=num+32;

}

}

}

//f

void endcopy()

{

//to read 2 strings

int n,m,d;

printf("Enter the number of elements in first string:\n");

scanf("%d",&n);

printf("Enter the number of elements in second string:\n");

scanf("%d",&m);

char s1[n],s2[m];

printf("Enter the first string: ");

scanf("%s",s1);

printf("Enter the second string: ");

scanf("%s",s2);

printf("String 1: %s\nString 2: %s\n",s1,s2);

//to read number of characters

printf("Enter the atmost number of characters to be copied:");

scanf("%d",&d);

cpend(s1,s2,d);

printf("New string = %s\n",s1);

}

//to copy the strings

void cpend(char a[],char b[],int m)

{

int i,k;

k=len(a)-1;

for(i=0;i<m;i++)

{

a[k]=b[i];

k++;

}

b[m]='\0';

}

**Output**:

cseb112@jtl-10:gcc ass41.c –o ass41

cseb112@jtl-10:./ass41

Enter

1.String Append

2.String Copy

3.Find Character

4.Find a character and replace

5.Compare 2 Strings

6.Copy Sting 2 to end of Stirng 1

7.Exit

1

Enter the number of elements in first string:

5

Enter the number of elements in second string:

4

Enter the first string: abcde

Enter the second string: abcd

String 1: abcde

String 2: abcd

Appended String : abcdeabcd

**Output**:

cseb112@jtl-10:gcc ass41.c –o ass41

cseb112@jtl-10:./ass41

Enter

1.String Append

2.String Copy

3.Find Character

4.Find a character and replace

5.Compare 2 Strings

6.Copy Sting 2 to end of Stirng 1

7.Exit

2

Enter the number of elements:

3

Enter the string:

abc

Enter the number of characters to be copied:

2

Copied String is : ab

**Output**:

cseb112@jtl-10:gcc ass41.c –o ass41

cseb112@jtl-10:./ass41

Enter

1.String Append

2.String Copy

3.Find Character

4.Find a character and replace

5.Compare 2 Strings

6.Copy Sting 2 to end of Stirng 1

7.Exit

3

Enter the number of elements:

3

Enter the string:

abc

Enter the character to be found : b

b is at 1 index

**output**:

cseb112@jtl-10:gcc ass41.c –o ass41

cseb112@jtl-10:./ass41

Enter

1.String Append

2.String Copy

3.Find Character

4.Find a character and replace

5.Compare 2 Strings

6.Copy Sting 2 to end of Stirng 1

7.Exit

4

Enter the number of elements:

3

Enter the string:

abc

Enter the replacement character : c

string : ccc

**output**:

cseb112@jtl-10:gcc ass41.c –o ass41

cseb112@jtl-10:./ass41

Enter

1.String Append

2.String Copy

3.Find Character

4.Find a character and replace

5.Compare 2 Strings

6.Copy Sting 2 to end of Stirng 1

7.Exit

5

Enter the number of elements in first string:

3

Enter the number of elements in second string:

3

Enter the first string: abc

Enter the second string: def

String 1: abc

String 2: def

String1 is smaller

**Output**:

cseb112@jtl-10:gcc ass41.c –o ass41

cseb112@jtl-10:./ass41

Enter

1.String Append

2.String Copy

3.Find Character

4.Find a character and replace

5.Compare 2 Strings

6.Copy Sting 2 to end of Stirng 1

7.Exit

Enter the number of elements in first string:

3

Enter the number of elements in second string:

3

Enter the first string: abc

Enter the second string: def

String 1: abc

String 2: def

Enter the atmost number of characters to be copied:2

New string = abcde

**2. Write a program to search the last occurrence of a substring in a given string.**

#include<stdio.h>

#include<string.h>

void main()

{

char s[100],c[100];

int l1,l2,i,j,count=0,m,k;

printf("Enter a sentence:\n");//to

gets(s);

printf("Enter the word to search:\n");//to read word to search

gets(c);

l1=strlen(s);//length of s

l2=strlen(c);//length of c

k=l1-1;

for(i=l1-1;i>=0;i--)

{

m=l2-1;

for(j=l2-1;j>=0;j--)

{

if(s[k]==c[j])

{

count++;

if(count==l2)//to find the word

{

printf("Last occurrence of %s is at index %d\n",c,i-l2+1);//to print the index

goto stop;

}

k--;

}

else

{

count=0;

k--;

break;

}

}

}

stop:

if(count==0)

printf("Word not present\n");

}

**Output**:

cseb112@jtl-10:gcc ass42.c –o ass42

cseb112@jtl-10:./ass42

Enter a sentence:

hi how r u hi

Enter the word to search:

hi

Last occurrence of hi is at index 11

**3. Write a program which replaces a substring with another in a given line of text.**

#include<stdio.h>

#include<string.h>

void rep(char [],char [],char []);//function prototype

void main()

{

char str[100],sub[100],new[100];

printf("Enter a string:\n");

gets(str);

printf("Enter substring to be replaced:\n");

gets(sub);

printf("Enter new substring:\n");

gets(new);

rep(str,sub,new);//function call

}

void rep(char str[],char sub[],char new[])

{

int a,b,c,i,j,k,f,start,end,count=0;

a=strlen(str);

b=strlen(sub);

c=strlen(new);

for(i=0;i<a;i++)

{

f=0;

start=i;

for(j=0;str[i]==sub[j];j++)

{

if(j==b-1)

{

f=1; //finding the word

count++;

}

i++;

}

end=i;

if(f==0)

i-=j;

else

{

for(j=start;j<end;j++)

{

for(k=start;k<a;k++) //deleting the found word

str[k]=str[k+1];

a--;

i--;

}

for(j=start;j<start+c;j++)

{

for(k=a;k>=j;k--)

str[k+1]=str[k]; //creating space for the new word

str[j]=new[j-start]; //inserting new word

a++;

i++;

}

}

}

if(count!=0)

printf("The string after replacing : %s\n",str);

else

printf("Word not found\n");

}

**Output**:

cseb112@jtl-10:gcc ass43.c –o ass43

cseb112@jtl-10:./ass43

Enter a string:

hi how are you

Enter substring to be replaced:

hi

Enter new substring:

hai

The string after replacing : hai how are you

**4. Write a program to reverse a string without using the library function. No extra string should be used and the source string itself should be modified to store the reversed string. Number of exchanges should be minimum.**

#include<stdio.h>

#include<string.h>

void main()

{

char s[100],temp;

int i,k,l;

printf("Enter a string:\n"); //to read a string

gets(s);

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

k=strlen(s)-1;

l=strlen(s)/2;

for(i=0;i<l;i++)

{

temp=s[i];

s[i]=s[k]; // to switch the values in elements

s[k]=temp;

k--;

}

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

}

**Output**:

cseb112@jtl-10:gcc ass44.c –o ass44

cseb112@jtl-10:./ass44

Enter a string:

ssn

Original string : ssn

Reveresed string : nss

**5. Write an interactive C program that will encode or decode a line of text. To encode a line of text, proceed as follows.**

**1. Convert each character, including blank spaces, to its ASCII equivalent.**

**2. Generate a positive random integer. Add this integer to the ASCII equivalent of each character. The same random integer will be used for the entire line of text.**

**3. Suppose that N1 represents the lowest permissible value in the ASCII code, and N2 represents the highest permissible value. If the number obtained in step 2 above (i.e., the original ASCII equivalent plus the random integer) exceeds N2, then subtract the largest possible multiple of N2 from this number, and add the remainder to N1. Hence the encoded number will always fall between N1 and N2, and will therefore always represent some ASCII character.**

**4. Display the characters that correspond to the encoded ASCII values.**

**The procedure is reversed when decoding a line of text. Be certain, however, that the same random number is used in decoding as was used in encoding.**

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

void main()

{

char a[100],b[100],d[100];

int i,n,k[100],r,c,n1,n2;

printf("Enter a sentence:\n");

gets(a);

i=0;

n1=a[0];

n2=a[0];

while(a[i]!='\0')

{

k[i]=a[i];

if(k[i]<n1)

n1=k[i]; //to find n1 and n2

if(k[i]>n2)

n2=k[i];

i++;

}

//Encoding

i=0;

srand(time(0));

r=rand();

while(a[i]!='\0')

{

n=r%n2;

k[i]+=n;

b[i]=k[i];

i++;

}

printf("Encoded String: %s\n",b);

//Decoding

i=0;

while(a[i]!='\0')

{

n=r%n2;

k[i]-=n;

d[i]=k[i];

i++;

}

printf("Dencoded String: %s\n",d);

}

**Output**:

cseb112@jtl-10:gcc ass45.c –o ass45

cseb112@jtl-10:./ass45

Enter a sentence:

abcd

Encoded String: jklm

Dencoded String: abcd