ASSIGNMENT NO.3(PART B-With Pointers)

TITLE: Implement following operations on string with / without pointers (without using library functions)

i. Length

ii. Palindrome

iii. String comparison

iv. Copy

v. Reverse

vi. Substring

Roll No.82

#include<stdio.h>

#include<ctype.h>

voidlen(char \*p,char \*q);

void palindrome(char \*p);

void compare(char \*p,char \*q);

void copy(char \*p);

void reverse(char \*p);

void substring(char \*p);

void main()

{

char str1[100],str2[100];

char \*p=&str1[0];

char \*q=&str2[0];

intch,i;

printf("\n Enter the first string=");

scanf("%s",&str1);

printf("\n Enter the second string=");

scanf("%s",&str2);

do

{

printf("\n1.length 2.palindrome 3.compare 4.copy 5.reverse 6.substring");

printf("\nEnter your choice=");

scanf("%d",&ch);

switch(ch)

{

case 1:len(p,q);

break;

case 2:palindrome(p);

break;

case 3:compare(p,q);

break;

case 4:copy(p);

break;

case 5:reverse(p);

break;

case 6:substring(p);

break;

default:printf("\nInvalid choice");

break;

}

}while(ch!=0);

}

voidlen(char \*p,char \*q)

{

int i,j,length1=0,length2=0;

while(\*(p+i)!='\0')

{

length1++;

i++;

}

printf("\n length of 1st string is= %d",length1);

while(\*(q+j)!='\0')

{

length2++;

j++;

}

printf("\n length of 2nd string is= %d",length2);

}

void palindrome(char \*p)

{

int i=0,j=0,f=0;

while(\*(p+j)!='\0')

{

j++;

}

j--;

while(\*(p+i)!='\0')

{

if(\*(p+i)!=\*(p+j))

{

f=1;

}

i++;

j--;

}

if(f==0)

{

printf("\n String is palindrome");

}

else

{

printf("\n String is not palindrome");

}

}

void compare(char \*p,char \*q)

{

int i,f1=0;

for(i=0;\*(p+i)!='\0'&&\*(q+i)!='\0';i++)

{

if(\*(p+i)==\*(q+i))

{

f1=1;

}

else

{

f1=0;

}

}

if(f1)

{

printf("\nTwo strings are equal");

}

else

{

printf("\nTwo strings are not equal");

}

}

void copy(char \*p)

{

char str2[100],\*q;

q=&str2[0];

int i;

for(i=0;\*(p+i)!='\0';i++)

{

\*(q+i)=\*(p+i);

}

\*(q+i)='\0';

printf("\n Original string 1 is =%s",p);

printf("\nCopied string is =%s",q);

}

void reverse(char \*p)

{

int i=0,j=0;

while(\*p!='\0')

{

p++;

i++;

}

printf("\n");

while(i>=0)

{

printf("%c",\*p);

i--;

p--;

}

printf("= is reversed string ");

}

void substring(char \*p)

{

char str2[100],\*q;

printf("\nEnter substring you want to find=");

scanf("%s",str2);

q=&str2[0];

int i=0,j=0,f1=0,a=0,b=0;

while(\*(p+i)!='\0')

i++;

i--;

while(\*(q+j)!='\0')

j++;

j--;

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

{

for(b=a;b<a+j;b++)

{

f1=1;

if(\*(p+b)!=\*(q+(b-a)))

{

f1=0;

break;

}

}

if(f1==1)

{

break;

}

}

if(f1)

{

printf("\nSecond string is a substring of 1st");

}

else

{

printf("\nSecond string is not a substring of 1st");

}

}

**OUTPUT:**

[admin@localhost ~]$ gccAsg3B.c

[admin@localhost ~]$ ./a.out

Enter the first string=psop

Enter the second string=psoop

1.length 2.palindrome 3.compare 4.copy 5.reverse 6.substring

Enter your choice=1

length of 1st string is= 4

length of 2nd string is= 5

1.length 2.palindrome 3.compare 4.copy 5.reverse 6.substring

Enter your choice=2

String is not palindrome

1.length 2.palindrome 3.compare 4.copy 5.reverse 6.substring

Enter your choice=3

Two strings are not equal

1.length 2.palindrome 3.compare 4.copy 5.reverse 6.substring

Enter your choice=4

Original string 1 is =psop

Copied string is =psop

1.length 2.palindrome 3.compare 4.copy 5.reverse 6.substring

Enter your choice=5

posp= is reversed string

1.length 2.palindrome 3.compare 4.copy 5.reverse 6.substring

Enter your choice=6

Enter substring you want to find=ps

Second string is a substring of 1st

1.length 2.palindrome 3.compare 4.copy 5.reverse 6.substring

Enter your choice=