Programmer Name: Harsh Bhat

Batch: E5

Problem Statement: Perform the following string operations without using the library functions

A. With pointers to arrays b. Without pointers to arrays

1. Substring, 2. Palindrome, 3. Compares, 4. Copy,5. Reverse

CODE

```
/*Problem Statement: Perform the following string operations without using the library functions
// With pointers to arrays
//  1.Substring, 2.Palindrome,3. Compares, 4. Copy,5. Reverse*/

// Name : Harsh Bhat
// Roll No : 22108

#include <stdio.h> #include
<conio.h> int length(char*);
void substring(char*, char*);
void palindrome(char*);
void compare(char*, char*);
void copy(char*); void
reverse(char*); void main()
{ char a[50];
char b[50];
int choice;
```

DS_Prog\0

```
  printf("enter a string a : ");
gets(a);
  printf("enter a string b : ");
gets(b);    do
  {

    printf("which operation do you want to perform \n1) substring \n2) palindrome \n3)
compare
\n4) copy \n5) reverse \n");
scanf("\n%d",&choice);      switch
(choice)
    {
    case 1:
      printf("\nthis checks string b is substring of a or not");
substring(a,b);          break;      case 2:
      printf("select string \n1)string a \n2)string b \n");
scanf("%d",&choice);          switch (choice)
      {          case 1:
palindrom(a);
break;          case 2:
palindrom(b);
break;          default:
        printf("\nenter proper choice");
break;
      }
break;      case 3:
compare(a,b);
break;      case 4:
      printf("\nwhich string do you want to copy \n1)string a \n2)string b \n");
scanf("%d",&choice);          switch (choice)
      {
case 1:
        copy(a);
        break;
case 2:
copy(b);
```

DS Prog I/O

```
break;
default:
            printf("\nenter proper choice");
break;
        }
break;
case 5:
        printf("\nwhich string do you want to reverse \n1)string a \n2)string b \n");
scanf("%d",&choice);          switch (choice)
        {          case
1:
reverse(a);
break;          case
2:
reverse(b);
break;
default:
            printf("\nenter proper choice");
break;
        }
     }
     printf("\nif you want more operations then press 1 otherwise 0 \n");
scanf("%d",&choice);
  }
  while (choice==1);
getch();
}
int length(char* str)
{
  char* ptr=str;
char* t=ptr;    while
(*ptr!='\0')
  {
     ptr++;
  }
  return (ptr-t) ;
```

DS Prog I/O

```
}
void substring(char* a, char* b)
{
   char* ptra=a;
char* ptrb=b;
while (*ptra!='\0')
   {
      if (*ptra==*ptrb)
      {
ptrb++;
      }
else
      {
         ptrb=b;
      }
      ptra++;       if
(ptrb==(b+length(b)))
      {
         printf("\nstring found");
break;
      }
   }
   if (ptrb!=(b+length(b)))
   {
      printf("\nstring not found");
   }
}
void palindrom(char* str)
{    char* ptra=str;    char*
ptrl=str+(length(str)-1);    while
((ptrl-ptra)>0)
   {
      if (*ptra==*ptrl)
      {
ptra++;
```

DS_Prog I/O

```
ptrl--;       }
else
     {
        printf("\nstring is not palindrome");
break;
     }
  }
  if((ptrl-ptra)==0 || (ptrl-ptra)==-1)
  {
     printf("\nstring is palindrome");
  }
 }
void compare(char* x,char* y)
{
   char* ptrx=x;    char* ptry=y;
char* ptrl=ptrx+(length(x)-1);
while (*ptrx!='\0')
  {
     if (*ptrx!=*ptry)
     {
        printf("\nstrings are not same");
break;
     }
     else if (ptrx==ptrl)
     {
        printf("\nstrings are same");
break;
     }
else
     {
ptrx++;
ptry++;
}
  }
}
void copy(char *str)
```

DS_Prog I/O

```
{
   char copy_str[50];
   int i = 0;    while
(str[i] != '\0')
   {
      copy_str[i] = str[i];
i++;
   }

   printf("The copied string is ");

   int j = 0;    while
(str[j] != '\0')
   {
      printf("%c", str[j]);
j++;
   }
}
void reverse(char *str)
{
   int los = length(str);

   printf("\nThe string before reversing is %s", str);
char temp[los + 1];    for (int i = 0; i < los; i++)
   {
      temp[i] = str[(los - 1) - i];
   }

   for (int i = 0; i < los; i++)
   {
      str[i] = temp[i];
   }

   printf("\nThe string after reversing is %s", str);
}
```

DS Prog I/O

OUTPUT

Enter the string a:
RACE
Enter the string b: RUN String
a is :
RACE
String b is:
RUN

Enter the operation you want to perform :
1.Substring
2.Palindrome
3.Compare
4.Copy
5.Reverse
1
Substring not found:
(Do you want to perform again. If yes enter 1 otherwise 0

Enter the operation you want to perform :
1.Substring
2.Palindrome
3.Compare
4.Copy

5.Reverse 2 Enter the string with which you wish to proceed:
1.
String a.
2.String b.
1

DS Prog I/O

no string is not a palindrome
Do you want to perform again
. If yes enter 1 otherwise 01
Enter the operation you want to perform :
1.Substring
2.Palindrome
3.Compare
4.Copy
5.Reverse
3 Strings are not same.
Do you want to perform again.
If yes enter 1 otherwise 01 Enter the operation you want to perform
1.Substring
2.Palindrome
3.Compare
4.Copy
5.Reverse
4
Enter the string with which you wish to proceed:
1
String a.
2.
String b.
1 COPIED STRING RACED²b

Do you want to perform again. If yes enter 1 otherwise 01

Enter the operation you want to perform :
1.Substring
2.Palindrome
3.Compare
4.Copy
5.Reverse

DS_Prog I/O

5
Enter the string with which you wish to proceed:
1
.String a.
2.
String b.
1
4AR
Do you want to perform again. If yes enter 1 otherwise 0

/* Problem statement : perform following string operations without using library functions
//Without pointer to array.
// 1.Substring, 2.Palindrome,3. Compares, 4. Copy,5. Reverse */

```
// Name : Sanika Gaikwad
// Roll No. : 22121

#include <stdio.h> int
length(char[]); void
substring(char[], char[]); void
palindrom(char[]); void
compare(char[], char[]); void
copy(char[], char[]); void
reverse(char[]); void main()
{    char
a[50];    char
b[50];    int
choice;
   printf("enter a string a : ");
gets(a);
   printf("enter a string b : ");
gets(b);    do
```

DS_Prog I/O

```
  {
    printf("which operation do you want to perform \n1) substring \n2) palindrome \n3)
compare
\n4) copy \n5) reverse\n");
scanf("\n%d",&choice);  switch
(choice)
    {
    case 1:
      printf("\nthis checks string b is substring of a or not");
substring(a,b);  break;  case 2:
      printf("select string \n1)string a \n2)string b \n");
scanf("%d",&choice);  switch (choice)
      {  case 1:
palindrom(a);
break;  case 2:
palindrom(b);
break;  default:
        printf("\nenter proper choice");
break;
      }
break;  case 3:
compare(a,b);
break;  case 4:
      printf("\nwhich string do you want to copy \n1)string a \n2)string b \n");
scanf("%d",&choice);  switch (choice)
      {  case
1:
copy(a,b);
break;  case
2:
copy(b,a);
break;
default:
        printf("\nenter proper choice");
break;
```

DS Prog I/O

```
        }
break;
case 5:
        printf("\nwhich string do you want to reverse \n1)string a \n2)string b \n");
scanf("%d",&choice);          switch (choice)
        {          case
1:
reverse(a);
break;          case
2:
reverse(b);
break;
default:
            printf("\nenter proper choice");
break;
        }
      }
      printf("\nif you want more operations then press 1 otherwise 0 \n ");
scanf("%d",&choice);
   }
   while (choice==1);
}
int length(char str[])
{    int
i=0;
   while (str[i]!='\0')
   {
i++;
   }
   return i;
}
void substring(char a[], char b[])
{    int i=0,j=0;
while (a[i]!='\0')
   {       if
(a[i]==b[j])
```

DS_Prog I/O

```
    {       j++;
}     else     {
j=0;     }     i++;
if (j==length(b))
    {
      printf("\nstring found");
break;
    }
  }
  if (j!=length(b))
  {
    printf("\nstring not found");
  }
}
void palindrom(char str[])
{   int i=0;   int
j=length(str)-1;   while
(i<length(str)/2)
  {     if
(str[i]==str[j])
    {
i++;
j--;     }
else
    {
      printf("\nstring is not palindrome");
break;
    }
  }
  if (i==length(str)/2)
  {
    printf("\nstring is palindrome");
  }
}
void compare(char x[],char y[])
```

DS Prog I/O

```
{    int i=0;
while (x[i]!='\0')
  {       if
(x[i]!=y[i])
     {
        printf("\nstrings are not same");
break;
     }
     else if (i==(length(x)-1))
     {
        printf("\nstrings are same");
break;
     }
else       {
i++;
     }
  }
}
void copy(char a[], char b[])
{    int
i;
  for (i=0;i<=(length(a)-1);i++)
  {
b[i]=a[i];
if (a[i]=='\0')
     {
break;
     }
  }
  printf("\nstring a is : ");
puts(a);    printf("\nstring
b is : ");    puts(b);
}
void reverse(char a[])
```

DS_Prog I/O

```
{    int i,j=length(a)-1;
char t;    for
(i=0;i<=(j/2);i++)
  {       t=a[i];
a[i]=a[j-i];
a[j-i]=t;
  }
  printf("\nreverse string is : ");
puts(a);  }
```

Output

1.Substring Searching
2.Check for Palindrome
3.String Comparison
4.Copy string
5.Reverse String
6.Quit Enter
Your Choice:
1 Enter 1st
string:
A Enter 2nd string:
B Press a Character
1.Substring Searching
2.Check for Palindrome
3.String Comparison
4.Copy string
5.Reverse String
6.Quit Enter Your
Choice: 2 Enter a
String:
CODE Not a palindrome

Press a Character
1.Substring Searching
2.Check for Palindrome
3.String Comparison

DS_Prog I/O

4.Copy string
5.Reverse String
6.Quit
Enter Your Choice
:3 Enter 1st
string:
MANGO
Enter 2nd string:
ORANGE
1st
1.Substring Searching
2.Check for Palindrome
3.String Comparison
4.Copy string
5.Reverse String
6.Quit Enter Your
Choice:
4
Enter a String:
RACE
Result=RACE
Press a
Character
1. Substring Searching
2. Check for Palindrome
3. String Comparison
4. Copy string
5. Reverse String
6. Quit
Enter Your Choice:
5 Enter a String:
run
Result= nur
Press a
Character

DS_Prog I/O

PUNE INSTITUTE OF COMPUTER TECHNOLOGY

PUNE - 411043

Department of Electronics & Telecommunication Engineering

ASSESMENT YEAR: 2021-2022 CLASS: SE V/VI/VII/VIII

SUBJECT: Data Structure and Algorithm

Practice Assignment **Roll No: 22108** **Date: 8/9/2021**

No: 1

DS_Prog I/O