String Practice Sheet

1. Write a C program to find length of a string with and without function.

#include‹stdio.h>

int main()

int i;

char str[100]; printf(”Enter a string\n”); gets(str); for(i=0;str[i]!='\0';i++); printf(”Length = %d”,i);

### Write a C program to copy one string to another string with and without function.

#include‹stdio.h>

int main()

int i;

char strl[100],str2[100]; printf(”Enter a string\n”); gets(str1); for(i=0;str1[i]!='\0';i++)

str2[i]=str1[i];

str2[i]='\0' ;

printf(”%s”,str2);

### Write a C program to concatenate two strings with and without function.

#include‹stdio.h>

int main()

int i,j;

char strl[100],str2[100]; printf(”Enter a string\n”); gets(str1);

gets(str2); for(i=0;str1[i]!='\0';i++);

for(j=8;str2[j]!='\8';j++)



str1[1]= ' \B ' ;

#### puts (str1);

##### Write a C program to compare two strings with and without function.

#include<stdio.h> int main()

#### 1nt 1, f=B;

char strl[100],str2[100]; printf(”Enter two strings\n”); gets(str1);

### gets( svr2) ;

for ( i=e; sort [i] != ' \8 ' j i++)

#### if(str1[i]==str2[i]);

else

break;

else

printf(”Different”);

printf(”Same”);

1. Write a C program to convert lowercase string to uppercase.

#include<stdio.h> int main()

1nt 1;

char str[100]; printf(”Enter a string\n”);

gets(str);

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

if(str[i]>=97&&str[i]‹=122) //if(str[i]>='a'&&str[i]‹='z')

st :n[ 1]=st:n[ 1] -32;

puts ( str);

1. Write a C program to convert uppercase string to lowercase.

#include<stdio.h> int main()

1nt 1;

char str[100]; printf(”Enter a string\n”); gets(str); for(i=0;str[i]!='\0';i++)

#### if(str[i]>=65&&str[i]‹=90) //if(str[i]>='A'&&str[i]‹='Z')

st :n[ 1]=st:n[ 1] +32;

puts ( str);

1. Write a C program to toggle case of each character of a string.

#include<stdio.h> int main()

1nt 1;

char str[100]; printf(”Enter a string\n”); gets(str); for(i=0;str[i]!='\0';i++)

#### if(str[i]>=97&&str[i]‹=122) str[i]=str[i]-32;

else if(str[i]>=65&&str[i]<=90) str[i]=str[i]+32;

puts(str);

1. Write a C program to find total number of alphabets, digits or special character in a string.

#include<stdio.h> int main()

int i,c=0,d=0,e=0; char str[100];

printf(”Enter a string\n”); gets(str); for(i=0;str[i]!='\0';i++)

if((str[i]>=97&&str[i]‹=122)||(str[i]>=65&&str[i]‹=90))

C ++ j

else if(str[i]>=48&&str[i]<=57) d++;

else

e++;

printf(”No. of Alphabets = %d\n”,c); printf(”No. of Digits = %d\n”,d);

printf(”No. of Special Characters = %d\n”,e);

1. WAP to input a string and copy all the vowels , consonants , digits and special characters in 4 different strings then display those 4 strings.

#include<stdio.h> int main()

int i,c=0,d=0,e=0,f=0;

char str[100],strl[100],str2[100],str3[100],str4[100]; printf(”Enter a string\n”);

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

1f ( str [ 1]== ' a ' | | str [ 1]== ' e ' | | str [ 1]== ' 1 ' | | str [ 1]== ' o ' | | str [ 1]== ' u ' | | str [ 1

]== ' A' | | str [ 1]== ' E ' | | str [ 1]== ' I ' | | str [ 1]== ' 0' | | str [ 1]== ' U ' ) str1[c++]=str[ 1] ;

else if((str[i]>=97&&str[i]‹=122)||(str[i]>=65&&str[i]‹=90))

str2 [d++]=str[1] ;

e1se 1f(str[1]›=48&&str[1]‹=57) str3 [e++]=str[1] ;

e1se

str4[f++]=str[1];

str1[ c ]=str2 [d]=str3 [e] =str4[I]= ' \0 ' ; printf ("Vowe1 s = " ) ;

puts(strl); printf(”Consonants = ”); puts(str2); printf(”Digits = ”);

put s ( str3) ;

pri ntf (" Spec 1a1 Cha ra cte rs = " ) ; put s ( str4) ;

1. Write a C program to count total number of vowels and consonants in a string.

#include<stdio.h> int main()

int i,c=0,d=0; char str[100];

printf(”Enter a string\n”); gets(str); for(i=0;str[i]!='\0';i++)

1f ( str [ 1]== ' a ' | | str [ 1]== ' e ' | | str [ 1]== ' 1 ' | | str [ 1]== ' o ' | | str [ 1]== ' u ' | | str [ 1

]== ' A' | | str [ 1]== ' E ' | | str [ 1]== ' I ' | | str [ 1]== ' O' | | str [ 1]== ' U ' ) c++;

else if((str[i]>=97&&str[i]‹=122)||(str[i]>=65&&str[i]‹=90))

d++;

printf(”No. of Vowels = %d\n”,c); printf(”No. of Consonants = %d\n”,d);

1. Write a C program to count total number of words in a string.

#include<stdio.h> int main()

int i,c=0,f=0; char str[100];

printf(”Enter a string\n”); gets(str); for(i=0;str[i]!='\0';i++)

if(str[i]==' ' | |str[i]=='\n' ||str[i]=='\t') f=0;

else if(f==0)

c++;

printf(”No. of Words = %d\n”,c);

## Write a C program to find reverse of a string with and without function.

#include‹stdio.h> int main()

int i,j=0,temp; char str[100];

printf(”Enter a string\n”); gets(str); for(i=0;str[i]!='\0';i++); l=l-1;

#### while(j‹i)

temp=str[i]; str[i]=str[j]; str[j]=temp;

1--j

puts(str);

1. Write a C program to check whether a string is palindrome or not.

#include‹stdio.h>

int main()

int i,j=0,f=0; char str[100];

printf(”Enter a string\n”); gets(str); for(i=0;str[i]!='\0';i++); l=l-1;

#### while(j‹i)

if(str[i]==str[j]); else

printf(”Not Palindrome\n”); break;

1--j

if(f==0)

printf(”Palindrome\n”);

### Write a C program to reverse order of words in a given string.

#include‹stdio.h>

int main()

int i,j=0,temp; char str[100];

printf(”Enter a string\n”); gets(str); for(i=0;str[i]!='\0';i++); l=l-1;

#### while(j‹i)

temp=str[i]; str[i]=str[j]; str[j]=temp;

1--j

puts(str);

## Write a C program to find first occurrence of a character in a given string.

#include<stdio.h> int main()

int i,t=-i; char str[100],a;

printf("Enter a string\n”); gets(str);

printf("Enter a character to search its 1st occurence\n”); scanf(”%c”,&a);

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

if (str[ i]==a)

break;

else

printf(”Character found at %d portion\n”,t+1);

pr1ntf ( "Character not found \ n" );

1. Write a C program to find last occurrence of a character in a given string.

#include<stdio.h> int main()

int i,j,t=-1; char str[100],a;

printf(”Enter a string\n”); gets(str);

printf(”Enter a character to search its last occurence\n”); scanf(”%c”,&a);

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



if(str[j]==a)

break;

else

printf(”Character found at %d portion\n”,t+1);

pr1ntf ( "Character not found \ n" );

1. Write a C program to search all occurrences of a character in given string.

#include<stdio.h> int main()

int i,t=0;

char str[100],a; printf("Enter a string\n”); gets(str);

printf(”Enter a character to search its occurences\n”); scanf(”%c”,&a);

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

if (str[ i]==a)

printf(”Character found at %d portion\n”,i+1);

##### If (t==0)

pr1ntf ( "Character not found \ n" );

1. Write a C program to count occurrences of a character in given string.

#include<stdio.h> int main()

int i,c=0;

char str[100],a; printf("Enter a string\n”); gets(str);

printf(”Enter a character to search its occurences\n”); scanf(”%c”,&a);

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

### if (str[ i]==a)

else

pr1ntf ( "Character not found \ n" );

printf(”Character present %d times\n”,c);

1. Write a C program to find highest frequency character in a string.

#include<stdio.h> int main()

int i,t,max; char str1[100]; int p[128]=(0};

printf(”Enter a string 1\n”); gets(str1); for(i=0;str1[i]!='\e'; i++)

max=p[0]; for(i=0;i<=127;i++)

### if(p[i]>max)

max=p[i];

printf(”Character = %c Frequency = %d\n”,t+1,max); for(i=t+1;i<=127;i++)

#### if(p[i]==max)

printf(”Character = %c Frequency = %d\n”,i+1,max);

1. Write a C program to find lowest frequency character in a string.

#include<stdio.h>

int main()

int i,t,min; char str1[100]; int p[128];

printf(”Enter a string l\n”); gets(str1); for(i=0;i<=127;i++)

#### p[i]=101;

p[ sari[i] -1]=l;

else

#### min=p[0];

for(i=0;i<=127;i++)

### if(p[i]‹min)

min=p[i];

printf(”Character = %c Frequency = %d\n”,t+1,min); for(i=t+1;i<=127;i++)

### if(p[i]==min)

printf(”Character = %c Frequency = %d\n”,i+1,min);

1. Write a C program to count frequency of each character in a string.

#include<stdio.h> int main()

int i,t,min; char strl[100]; int p[l28]=(0};

printf("Enter a string l\n”); gets(str1); for(i=e;str1[i]!='\e';i++)

for(i=0;i<=127;i++)

1f(p[1]›0)

pr1ntf( "Character = be Frequency = id\n",1+1, p[1] );

1. Write a C program to remove first occurrence of a character from string.

#include<stdio.h> int main()

int i,j,t=0; char str[100],a;

printf(”Enter a string\n”); gets(str);

printf("Enter a character to delete its 1st occurence\n”); scanf("%c”,&a);

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

if (str[ i]==a)



break;

else

pr1ntf ( "Character not present. \ n" );

printf(”String after deletion: %s\n”,str);

1. Write a C program to remove last occurrence of a character from string.

#include<stdio.h> int main()

int i,j,x,t=0;

char str[100],a; printf(”Enter a string\n”); gets(str);

printf("Enter a character to delete its last occurence\n”);

s canf ( "gc ", &a);

for (x=B; str [x] != ' \e ' ; x++); x=x -1;

for ( 1=x;1›=B; 1- -)

### if (str[ i]==a)



**break;**

#### if(t==0)

printf(”Character not present.\n”);

else

printf(”String after deletion: %s\n”,str);

##### Write a C program to remove all occurrences of a character from string.

#include<stdio.h> int main()

int i,j,t=0; char str[100],a;

printf(”Enter a string\n”); gets(str);

printf(”Enter a character to delete its all occurence\n”); scanf(”%c",&a);

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

if (str[ i]==a)



**else**

pr1ntf ( "Character not present . \ n" );

printf(”String after deletion: %s\n”,str);

##### Write a C program to remove all repeated characters from a given string.

#include<stdio.h> int main()

int i,j,t=0; char str[100],a;

printf("Enter a string\n”); gets(str); for(i=0;str[i]!='\0';i++)



printf(”String after deletion of all repeated charcters: %s\n”,str);

##### Write a C program to replace first occurrence of a character with another in a string.

#include <stdio.h> int main()

char str[100],a,b; printf(”Enter a string\n”); gets(str);

printf(”Enter a character you want to get replaced its 1st occurence\n”);

s canf ( "'Xc'\*c", &a);

printf("Enter a character you want to replace with\n”); scanf(”%c%\*c”,&b);

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

### if (str[ i]==a)

st:r [ 1]=b;

##### break;

**else**

pr1ntf ( "Character not present . \ n" );

printf(”String after replacing: %s\n”,str);

##### Write a C program to replace last occurrence of a character with another in a string.

#include <stdio.h> int main()

char str[100],a,b; printf(”Enter a string\n”); gets(str);

printf(”Enter a character you want to get replaced its last occurence\n");

s canf ( "gcg\*c ", &a);

printf("Enter a character you want to replace with\n”);

s canf ( "'Xc'\*c ", &b);

for (x=0; str [x] != ' \e ' ; x++);

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

### if (str[ i]==a)

st:r [ 1]=b;

##### break;

else

pr1ntf ( "Character not present . \ n" );

printf(”String after replacing: %s\n”,str);

##### Write a C program to replace all occurrences of a character with another in a string.

#include <stdio.h> int main()

int i,t=0;

char str[100],a,b; printf("Enter a string\n”); gets(str);

printf(”Enter a character you want to get replaced its all occurences\n”);

s canf ( "'Xc'\*c ", &a);

printf("Enter a character you want to replace with\n”); scanf("%c%\*c”,&b);

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

### if (str[ i]==a)

st:r [ 1]=b;

else

pr1ntf ( "Character not present . \ n" );

printf(”String after replacing: %s\n",str);

##### Write a C program to find first occurrence of a word in a given string.

#include<stdio.h>

#include<string.h> int main()

int i,j,d,x,c=0,f=0;

char strl[100],str2[100]; printf("Enter a string\n”); gets(str1);

for(i=strlen(str1)+1;i>=1;i--)

str1[0]=' ; x=strlen(strl); strl[x+1]='\0';

str1[x]=' ;

printf(”Enter 2nd string\n”); gets(str2); for(i=0;str1[i]!='\0';i++)

if(str1[i]==' '&&str1[i+1]!=' ')

c++;

if(str1[i]==str2[0])

d=0;

for(j=0;str2[j]!='\0';j++) if(str1[i+j]==str2[j])

d++;

if(d==strlen(str2)&&str1[i+j]==' '&&str1[i-1]==' ')

printf(”Found at %d position\n”,c); break;

if(f==0)

printf(”String not present.\n”);

### Write a C program to find last occurrence of a word in a given string.

#include‹stdio.h>

#include‹string.h> int main()

int i,j,d,x,c=0,f=0;

char strl[100],str2[100]; printf(”Enter a string\n”); gets(str1);

if(str1[0]!=' ') for(i=strlen(str1)+1;i›=1;i--)

strl[i]=str1[i-1];

str1[0]=' ; x=strlen(strl); strl[x+1]='\0';

str1[x]=' ;

printf(”Enter 2nd string\n”); gets(str2);

for(i=strlen(str1)-1;i>=0;i--)

if(str1[i]==' '&&str1[i-1]!=' ')

c++;

if(str1[i]==str2[0])

d=0;

for(j=0;str2[j]!='\0';j++) if(str1[i+j]==str2[j])

d++;

if(d==strlen(str2)&&str1[i+j]==' '&&str1[i-1]==' ')

printf(”Found at %d position from last\n”,c); break;

if(f==0)

printf(”String not present.\n”);

### Write a C program to search all occurrences of a word in given string.

#include‹stdio.h>

#include‹string.h> int main()

int i,j,d,x,c=0,f=0;

char strl[100],str2[100]; printf(”Enter a string\n”); gets(str1);

if(str1[0]!=' ') for(i=strlen(str1)+1;i›=1;i--)

strl[i]=str1[i-1];

str1[0]=' ; x=strlen(strl); strl[x+1]='\0';

str1[x]=' ;

printf(”Enter 2nd string\n”); gets(str2); for(i=0;str1[i]!='\0';i++)

if(str1[i]==' '&&str1[i+1]!=' ')

c++;

if(str1[i]==str2[0])

d=0;

for(j=0;str2[j]!='\0';j++) if(str1[i+j]==str2[j])

d++;

if(d==strlen(str2)&&str1[i+j]==' '&&str1[i-1]==' ')

printf(”Found at %d position\n”,c);

if(f==0)

printf(”String not present.\n”);

### Write a C program to count occurrences of a word in a given string.

#include‹stdio.h>

#include‹string.h> int main()

int i,j,d,x,c=0,f=0,count=0; char strl[100],str2[100]; printf(”Enter a string\n”); gets(str1);

if(str1[0]!=' ') for(i=strlen(str1)+1;i›=1;i--)

strl[i]=str1[i-1];

str1[0]=' ; x=strlen(strl); strl[x+1]='\0';

str1[x]=' ;

printf(”Enter 2nd string\n”); gets(str2); for(i=0;str1[i]!='\0';i++)

if(str1[i]==str2[0]) d=0;

for(j=0;str2[j]!='\0';j++) if(str1[i+j]==str2[j])

d++;

if(d==strlen(str2)&&str1[i+j]==' '&&str1[i-1]==' ') count++;

printf(”%s is present %d times\n”,str2,count);

### Write a C program to remove first occurrence of a word from string.

Rishabh Singh D1

#include<stdio.h>

#include<string.h> int main()

int i,j,d,x,c=0,f=0,z,y; char strl[100],str2[100]; printf(”Enter a string\n”); gets(str1);

if(strl[0]!=' ') for(i=strlen(strl)+1;i>=1;i--)



str1[0]= ' ; x=st rlen( str1) ; strl[x+1]='\0';

S £ P1 [ X ] = ' ' j

printf(”Enter 2nd string\n”); gets(str2); for(i=0;strl[i]!='\0';i++)

### if(str1[i]==' '&&str1[i+1]!=' ')

c++;

if(strl[i]==str2[0])

d=0;

for(j=0;str2[j]!='\0';j++)

#### if(strl[i+j ]==str2[j])

d++;

if(d==strlen(str2)&&str1[i+j]==' '&&str1[i-1]==' ')

for(z=0;z‹=strlen(str2);z++) for(y=i-1;strl[y]!='\0';y++)



printf(”String after deleting its 1st occurence:%s\n”,str1); break;

if(f==0)

printf(”Word not present.\n”);

### Write a C program to remove last occurrence of a word in given string.

#include‹stdio.h>

#include‹string.h> int main()

int i,j,d,x,c=0,f=0,z,y; char strl[100],str2[100]; printf(”Enter a string\n”); gets(str1);

if(str1[0]!=' ') for(i=strlen(str1)+1;i›=1;i--)

strl[i]=str1[i-1];

str1[0]=' ; x=strlen(strl); strl[x+1]='\0';

str1[x]=' ;

printf(”Enter 2nd string\n”); gets(str2);

for(i=strlen(str1)-1;i>=0;i--)

if(str1[i]==' '&&str1[i+1]!=' ')

c++;

if(str1[i]==str2[0])

d=0;

for(j=0;str2[j]!='\0';j++) if(str1[i+j]==str2[j])

d++;

if(d==strlen(str2)&&str1[i+j]==' '&&str1[i-1]==' ')

for(z=0;z<=strlen(str2);z++) for(y=i-1;str1[y]!='\0';y++)

strl[y]=str1[y+1];

printf(”String after deleting its 1st occurence:%s.”,str1); break;

if(f==0)

printf(”Word not present.\n”);

### Write a C program to remove all occurrence of a word in given string.

#include‹stdio.h>

#include‹string.h> int main()

int i,j,d,x,c=0,f=0,z,y; char strl[100],str2[100]; printf(”Enter a string\n”); gets(str1);

if(str1[0]!=' ')

for(i=strlen(str1)+1;i›=1;i--) strl[i]=str1[i-1];

str1[0]=' ; x=strlen(strl); strl[x+1]='\0';

str1[x]=' ; printf(”Enter 2nd string\n”);

gets(str2); for(i=0;str1[i]!='\0';i++)

if(str1[i]==' '&&str1[i+1]!=' ')

c++;

if(str1[i]==str2[0])

d=0;

for(j=0;str2[j]!='\0';j++) if(str1[i+j]==str2[j])

d++;

if(d==strlen(str2)&&str1[i+j]==' '&&str1[i-1]==' ')

for(z=0;z<=strlen(str2);z++) for(y=i-1;str1[y]!='\0';y++)

strl[y]=str1[y+1];

if(f==0)

printf(”Word not present.\n”);

else

printf(”String after deleting its 1st occurence:%s\n”,str1);

### Write a C program to trim leading white space characters from given string.

#include‹stdio.h>

int main()

int i,j,t=0; char str[100];

printf(”Enter a string\n”); gets(str); for(i=0;str[i]==' ' ;i++)



printf(”String after deletion of leading white spaces:%s\n”,str);

1. Write a C program to trim **trailing** white space characters from given string.

#include<stdio.h> int main()

int i,j,x; char str[100];

printf(”Enter a string\n”); gets(str); for(x=0;str[x]!='\e';x++);

for(i=x-1;str[i]==' ';i--)



printf(”String after deletion of trailing white spaces:%sOK”,str);

1. Write a C program to trim both leading and **trailing** white space characters from given string.

#include<stdio.h> int main()

int i,j,x; char str[100];

printf("Enter a string\n”);

### gets( sir ) ;

for ( i=0; str [ i]== ' ' ; 1++ )



for (x= 0; str [x] != ' \6 ' ; x++);

//leading

for(i=x-l;str[i]==' ';i--) //trailing



printf(”String after deletion of leading & trailing white spaces:%sOK”,str);

1. Write a C program to remove all extra blank spaces from given string.

#include<stdio.h> int main()

int i,j,x; char str[100];

printf(”Enter a string\n”);

### gets( sir ) ;

for ( i=0; str [ i]== ' ' ; 1++ )

//leading



for (x= 0; str [x] != ' \6 ' ; x++);

for(i=x-l;str[i]==' ';i--) //trailing



for(i=0;str[i]!='\e';i++) // midle



printf(”String after deletion of extra white spaces:%sOK”,str);

1. Write a C Program to input a string and check whether the string is pangram or not. A string is said to be pangram if it contains all the alphabets of English ( Could be upper or lower )

#include<stdio.h> //The Quick brown fox jumps over the lazy dog. = Panagram int main()

int i,t=0; char str[100]; int f[26]={0};

printf(”Enter a string\n”); gets(str); for(i=0;str[i]!='\0';i++)

if(str[i]>=97&&str[i]‹=122)

### f[str[i]-97]++;

else if(str[i]>=65&&str[i]‹=90)

### f[str[i]-65]++;

for(i=0;i‹=25;i++)

#### if(f[i]==0)

printf(”Not Panagram\n”);

brea k;

if(t==0)

printf(”Panagram\n”);

1. Write a C Program to input 2 strings and check whether the strings are anagram or not. 2 strings are said to be anagram if they contains same set of letters and length.

#include<stdio.h> // listen silent are anagrams. int main()

int i,t=0,x,y;

char strl[100],str2[100]; int p[128]={0},q[128]={0};

printf(”Enter a string I\n”);

gets(str1);

printf(”Enter a string 2\n”); gets(str2); for(x=0;str1[x]!='\0';x++);

for(y=0;str2[y]!='\0';y++); if(x==y)



q[str2[i]-1]++; for(i=0;i<=127;i++)

printf(”Not Anagram\n”);

break;

printf(”Anagram\n”);

else

printf(”Not Equal lengh therfore not anagram”);

OR

#include<stdio.h> int main()

int i,j,t,temp=0,x,y; char strl[100],str2[100];

printf(”Enter a string l\n”); gets(str1);

printf(”Enter a string 2\n”); gets(str2); for(x=0;str1[x]!='\0';x++);

for(y=0;str2[y]!='\0';y++); if(x==y)



if(str1[j]>str1[j+1]) //sorting 1st string

t=str1[j]; strl[j]=str1[j+1]; strl[j+1]=t;

if(str2[j] >str2[j+1]) //sorting 2nd string

t=str2[j]; str2[j]=str2[j+1]; str2[j+1]=t;

for(i=0;str1[i]!='\0' ;i++)

if(str1[i]!=str2[i])

printf(”Not Anagram\n”); temp=1;

break;

if(temp==0)

printf(”Anagram\n”);

else

printf(”Not Equal lengh therfore not anagram”);

## Write a program to input a word from the user and print it in the following way. For example, if the word is PROGRAM, the program will print it as-

### P R

P R 0

P R 0 G

P R 0 G R

P R 0 G R A

### P R 0 G R A M

#include‹stdio.h> int main()

let l,X,

char str[100];

printf(”Enter a string 1\n”); gets(str); for(x=0;str[x]!='\0';x++); for(i=l;i<=x;i++)

printf(”%.\*s\n”,i,str);

1. Write a program to search a middle name in the name consisting of first name, middle name and last name.

#include<stdio.h>

int main()

int i,x,j,k;

char str[100],str2[100]; printf(”Enter a string 1\n”); gets(str); for(x=0;str[x]!='\0';x++);

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

for (j=1+1, k=0; s tr [j ] != ' ' ; j ++, k++)

str2 [ k]=str [j ];

str2 [ k]=' \0 ' ; brea k;

printf(”Middle name = %s”,str2);

1. Write a menu driven program to perform the following task- o Find length of a string
   * Copy of one string into another
   * Capitalize all letters of a string
   * Reverse of string
   * Comparison of two strings

#include <stdio.h>

#include <string.h> int main()

int n,i,j,t;

char str[100],str1[100],str2[100],str3[100]; printf(”Enter a string 1\n”);

gets(str);

printf(”Enter\nl To find length\n2 To copy string into another\n3 Capitalize all letters\n4 Reverse string\n5 Compare 2 strings\n”);

scanf(”%d%\*c”,&n); switch(n)

case 1:

pr1ntf ( " Length = id ", str1en( str) );

##### break;

case 2:

strcpy(strl,str);

printf(”Copied String =%s”,strl); break;

case 3:

for(j=0;str[j]!='\0';j++)

if(str[j]›='a'&&str[j]‹='z') str2[j]=str[j]-32;

else

str2[j]=str[j];

str2[j]='\0';

printf(”String after capitalizing= %s”,str2); break;

case 4:

pr1ntf ( " Reversed Str 1ng= is ", strrev ( str) ); break;

case 5:

printf(”Enter another String to Compare\n”);

# gets(str3);

#### t=strcmp(str,str3); if(t==0)

printf(”Same\n”);

**else**

printf(”Different\n”);

brea k;

**default :**

printf(”Wrong Input”);

1. Define string. Differentiate between ’a’ and "a".

A string is a data type used in programming, such as an integer and floating point unit, but is used to represent text rather than numbers. It is comprised of a set of characters that can also contain spaces and numbers.

" is an arrays of chars, ' ' is a char.

"a" is a string literal and a is a char primitive

Double quotes is for string and single quotes for character

1. **Differentiate between character array and string. Give one example of each.**

**String** is a sequence of characters that is treated as a single data item and terminated bv null character' 1e ' . C language does not support strings as a data tvpe. A **string** is actuallv one- dimensional arrav of characters in C language. These are often used to create meaningful and readable programs.

**For example** : The string "hello world" contains 12 characters including ' 1e' character which is automaticallv added bv the compiler at the end of the string.

1. **What are the functions used for reading a string? If you want to read your full name, which function you will prefer** ? **Why?**

**We can use scanf(“%[^\n] ”,str); OR gets(str); both have same meaning. I will prefer gets(string\_name); as it is easy to use.**

1. **What are the ways to initialize 1D and 2D string?**

char str[5] = {”Java”};

char str[5][10] = {”Java”, "Python", "C++", "HTML", ”SQL”};

#### What will be the output of following program? void main()

char str1[] = "abcd"; char str2[] = "abcd";

if(str1==str2)

printf("Equal");

else

printf("Unequal");

) OUTPUT: Unequal

#### What will be the output of the following program? void main()

( char s[]= “Hello, World”;

printf(“>>P»s<<\n”,s); printf(“>>P»20s<<\n”,s); printf(“>>P»-20s<<\n”,s);

printf(“>>P».4s<<\n”,s);

printf(“>>P»-20.4s<<\n”,s);

printf(“>>P»20.4s<<\n”,s);

>>HeIIo,WorId<<

>> HeIIo,WorId<<

>>HeIIo,WorId <<

>>HeII<<

>>HeII

HeII<<

#### What are wrong initializations of the following string arrays?

(i) char str[]={’h’, ’e’, ’I’, ’I’, ’o’}; (ii) char str[5]={’h’, ’e’, ’I’, ’I’, ’o’, “\0’};

(iii) char str[]={’h’, ’e’, ’I’, ’I’, ’o’, ’\0’}; (iv) char str[6]={’h’, ’e’, ’I’, ’I’, ’o’, ’\0’};

1. char str[]= “hello”;
2. char str[][]={“hi”, “hello”, “good”, “bad”, “better”};
3. char str[5][]= {‘1i”, ‘1eIIo”, “good”, “bad”, “better”};
4. char str[5][10]= {“hi”, “hello”, “good”, “bad”, “better”};

OUTPUT: error in vi and vii, rest are correct (warning in ii)

1. **Write a** C **program that reads the name of a person as input and print the name in an abbreviated fashion, e.g., Dennis Ritchie as D.R.**

#include‹stdio.h> int main()

int i,x;

char str[100];

printf(”Enter a string 1\n”); gets(str); printf(”%c.”,str[0]);

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

if(str[i]==' '&&str[i+1]!=' ')

printf(”%c.”,str[i+1]);

1. **Write a program to store name of ten cities and rewrite it in alphabetical order.**

#include ‹stdio.h>

#include ‹string.h> int main()

int i,j,x;

char str[10][100],temp[10]; printf(”Enter name of 10 cities\n”); for(i=0;i<=9;i++)

scanf(”%[’\n]%\*c”,str[i]); printf(”Cities before sorting:\n”); for(i=0;i<=9;i++)

printf(”%s\t”,str[i]); for(i=1;i<=10-1;i++)

for(j=0;j<=10-i-1;j++) if(strcmp(str[j],str[j+1])>0)

strcpy(temp,str[j]); strcpy(str[j],str[j+1]); strcpy(str[j+1],temp);

printf(”\nCities after sorting:\n”); for(i=0;i<=9;i++)

printf(”%s\t”,str[i]);

### Write a C program to remove the white spaces from a string.

#include‹stdio.h>

int main()

int i,j,x; char str[100];

printf(”Enter a string\n”); gets(str); for(i=0;str[i]!='\0';i++)

if(str[i]==' ')

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

str[j]=str[j+1];

1--j

printf(”String after deletion of all white spaces:%s0K”,str);

### What will be the output of following programs?

1. #include<stdio.h> #include<string.h> int main()

char str1[20] = "Hello", str2[20] = "World"; printf("%s\n", strcpy(str2, strcat(str1, str2))); return 0;

### } OUTPUT:Hello World

1. #include<stdio.h>

int main()

char p[] = "%d\n";

P[!] - 'c';

printf(p, 65);

return 0;

) OUTPUT:A

1. #include<stdio.h> #include<string.h> int main()

printf("%d\n", strlen("123456"));

### return 0;

#### ) OUTPUT:6

1. #include<stdio.h>

int main()

char s[25] = "The cocaine man";

int i=0; char ch;

ch = s[++i]; printf("%c", ch); ch = s[i++]; printf("%c", ch); ch = i++[s]; printf("%c", ch); ch = ++i[s]; printf("%c", ch); return 0;

} OUTPUT: hhe!