+

Name	Adwait S Purao
UID no.	2021300101
Experiment No.	7

AIM:	Implement various text processing problems.
Program 1	
PROBLEM STATEMENT :	Write a program to count the number of vowels, consonants, total characters and words in the given string.
ALGORITHM:	1. START 2. Define function void findvowelconsonants with character input str 3. Set integer variables vow_count, count, consonants to zero 4. while (str[count] != '\0') if (str[count] == 'a' || str[count] == 'e' || str[count] == 'i' || str[count] == 'o' || str[count] == 'u' || str[count] == 'A' || str[count] == 'E' || str[count] == 'I' || str[count] == 'O' || str[count] == 'U') vow_count++ 5. else consonants++ 6. count++ 7. Print the number of vowels and consonants 8. Define function int string length with char input str 9. Initialize len to zero 10. while (str[len] != '\0') 11. len++ 12. Return len 13. Define function int word_num with character input str 14. Define two integer variables i and count=0 15. for (i = 0; str[i] != '\0'; i++) 16. if (str[i] == ' ' && str[i + 1] != ' ') count++ 17. Return count+1 18. Define function int main 19. Initialize character variable str[100]

	20. Initialize integer variables count=0 and i 21. Input the string 22. Call functions findvowelconsonants , string_length, word_num 23. Print number of vowels, consonants,string length,number of words 24. STOP

PROGRAM:	#include<stdio.h>
	void
	findvowelconsonants (char str[])
	{
	int vow_count = 0;
	int count = 0;
	int consonants = 0;
	while (str[count] != '\0')
	{
	
	if (str[count] == 'a' || str[count] == 'e' || str[count] == 'i'
	|| str[count] == 'o' || str[count] == 'u' || str[count] == 'A'
	|| str[count] == 'E' || str[count] == 'I' || str[count] == 'O'
	|| str[count] == 'U')
	{
	vow_count++;
	
	}
	else
	{
	consonants++;
	}
	count++;
	}
	printf ("\nThe string has %d vowels\n", vow_count);

```
  printf ("\nThe string has %d consonants\n ", consonants);
}

int
string_length (char str[])
{

  int len = 0;
  while (str[len] != '\0')
    {
      len++;

    }
  return len;

}

int
word_num (char str[])
{
  int count = 0, i;


  for (i = 0; str[i] != '\0'; i++)
    {
      if (str[i] == ' ' && str[i + 1] != ' ')
            count++;
    }
  return count + 1;
}

int
main ()
{
  char str[100];
  int count = 0, i;
  printf ("Enter the string:\n");
  scanf (" %[^\n]s ", str);
  findvowelconsonants (str);
  string_length (str);
```

```
  printf ("\nLength of string:%d\n", string_length (str));
  word_num (str);
  printf ("\nNumber of words in given string are: %d\n", word_num (str));

return 0;
}
```

RESULT:

```
Enter the string:
I am currently studying at spit

The string has 8 vowels

The string has 23 consonants

Length of string:31

Number of words in given string are: 6
```

Program 2

PROBLEM STATEMENT :	Write a Menu driven Program to i)copy one string to another one by one character. ii) Find the string length iii) compare two strings iv) reverse the string v) Concatenate one string to another string. vi) lower case to upper
ALGORITHM:	1. START 2. Define function void cop with two string inputs str 1 and str 2 3. Initialize integer variable i 4. for(i=0; str1[i]!='\0'; ++i) store the value of str 1 in str 2 5. Put null character at the end of string 2 6. Define function int len with character input str1

	7. Initialize integer variable len to zero 8. for(int i=0; str1[i]!='\0'; ++i) length++ 9. return length 10. Define function int com with two string inputs str 1 and str 2 11. Initialize integer variable dif to zero 12. for(int i=0; str1[i]!='\0' || str2[i]!='\0'; ++i) 13. dif = str1[i] - str2[i] 14. return dif 15. Define function char rev with string input str1 16. Initialize integer variables i,j 17. Call function len, int l = len(str1) 18. Initialize string char tempstr[l] 19. j=l-1 20. for(i=0; i<l; i++) 21. Store str1 in tempstr 22. j— 23. End tempstr with null character 24. Call function cop(tempstr, str1) 25. Define function void cat with two string input str1 and str2 26. Initialize integer variables i,j 27. Call function len, int l = len(str1) 28. for(i=0, j=l; str2[i]!='\0'; ++i, ++j) 29. str1[j] = str2[i] 30. End str1 with null character 31. Define function void up with two string inputs str and strn 32. Initialize integer variable i 33. for(i=0;str[i]!='\0';i++) 34. if(str[i] >= 97 && str[i] <= 122) strn[i] = str[i] – 32 35. else strn[i] = str[i] 36. End strn with null character 37. Define function int main 38. Initialize integer variables ch and trash 39. Initialize strings str 1,str 2,strn1,strn2 40. Read two strings 41. Do while(ch!=7) 42. Print Copy first string to second\n2) Length of string\n3) Compare two strings\n4) Reverse string\n5) Concatenate second string to first

	string\n6) Convert string to upper case\nEnter choice: 43. Define switch case statement 44. Case 1 45. Read string 46. Call function cop 47. Print copied string 48. break 49. Case 2 50. Read string 51. Call function len(str) 52. Break 53. Case 3 54. Read two strings 55. Call function com(str 1,str 2) 56. Break 57. Case 4 58. Read string 59. Call function rev(str 1) 60. Print reversed string 61. Break 62. Case 5 63. Read two strings 64. Call function 65. cat(str 1,str2) 66. print catenated string 67. break 68. Case 6 69. Read two strings 70. Call function up(str1,strn1) up(str2,strn2) 71. Print strn1 and strn2 72. Break 73. If input is not between 1 to 6 then print default statement Invalid input 74. STOP

PROGRAM:	#include <stdio.h>
	void cop(char str1[], char str2[])
	{
	int i;
	for(i=0; str1[i]!='\0'; ++i)
	{
	str2[i] = str1[i];
	}
	str2[i] = '\0';
	}
	int len(char str1[])
	{
	int length=0;
	for(int i=0; str1[i]!='\0'; ++i)
	{
	++length;
	}
	return length;
	}
	int com(char str1[], char str2[])
	{
	int dif = 0;
	for(int i=0; str1[i]!='\0' || str2[i]!='\0'; ++i)
	{
	dif = str1[i] - str2[i];
	}
	return dif;
	}
	char rev(char str1[])
	{
	int l = len(str1), i, j;
	char tempstr[l];
	j=l-1;

```
    for(i=0; i<l; i++)
    {
        tempstr[i] = str1[j];
        j--;
    }
    tempstr[i] = '\0';
    cop(tempstr, str1);
}
void cat(char str1[], char str2[])
{
    int l = len(str1), i, j;
    for(i=0, j=l; str2[i]!='\0'; ++i, ++j)
    {
        str1[j] = str2[i];
    }
    str1[j] = '\0';
}
void up(char str[],char strn[])
{
    int i;
    for(i=0;str[i]!='\0';i++)
    {
        if(str[i] >= 97 && str[i] <= 122)
            strn[i] = str[i] - 32;
        else
            strn[i] = str[i];
    }
    strn[i]='\0';
}

int main()
{
    int ch;
    int trash;
    char str1[1024];
    char str2[1024];
   char strn1[100],strn2[100];
   printf("Enter string 1: ");
    scanf("%[^\n]",str1);
    printf("Enter string 2: ");
```

```
   scanf(" %[^\n]",str2);
  do{
   printf("1) Copy first string to second\n2) Length of string\n3) Compare
two strings\n4) Reverse string\n5) Concatenate second string to first
string\n6) Convert string to upper case\nEnter choice: ");
   scanf("%d", &ch);
   switch (ch)
   {
   case 1:
         printf("Enter string: ");
         scanf(" %[^\n]s ", str1);
         cop(str1, str2);
         printf("Copied string: %s", str2);
      break;
   case 2:
         printf("Enter string: ");
         scanf(" %[^\n]s ", str1);
         printf("Length: %d", len(str1));
      break;
   case 3:
         printf("Enter string 1: ");
         scanf(" %[^\n]s ", str1);
         printf("Enter string 2: ");
         scanf(" %[^\n]s ", str2);
         printf("%d", com(str1, str2));
      break;
   case 4:
         printf("Enter string: ");
         scanf(" %[^\n]s ", str1);
         rev(str1);
         printf("%s", str1);
      break;
   case 5:
         printf("Enter string 1: ");
         scanf(" %[^\n]s ", str1);
         printf("Enter string 2: ");
         scanf(" %[^\n]s ", str2);
         cat(str1, str2);
         printf("%s", str1);
      break;
```

```
   case 6:
         {
            scanf("%[^\n]", str1);
            scanf("%[^\n]", str2);
            up(str1,strn1);
            up(str2,strn2);
            printf("%s\n",strn1);
            printf("%s\n\n",strn2);
            break;
         }
   default: printf("Invalid input");
      break;
   }
   } while(ch!=7);

   return 0;
}
```

RESULT:

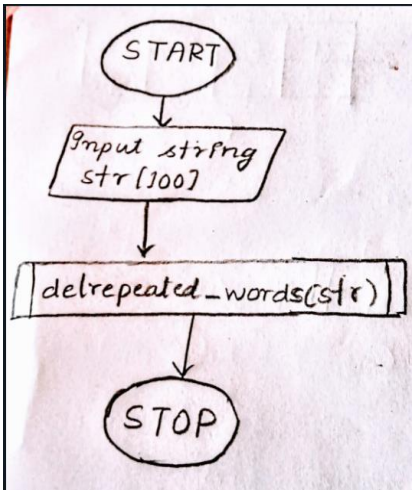
```
Enter string 1: I am Adwait
Enter string 2: I like trekking
1) Copy first string to second
2) Length of string
3) Compare two strings
4) Reverse string
5) Concatenate second string to first string
6) Convert string to upper case
Enter choice: 6
I AM ADWAIT
I LIKE TREKKING
```

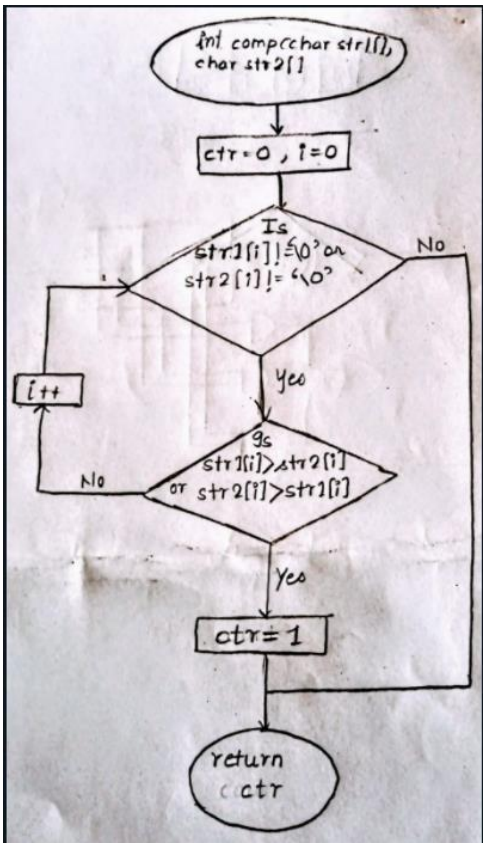
```
Enter string 1: I like to play cricket
Enter string 2: I like to walk
1) Copy first string to second
2) Length of string
3) Compare two strings
4) Reverse string
5) Concatenate second string to first string
6) Convert string to upper case
Enter choice: 7
Invalid input
```

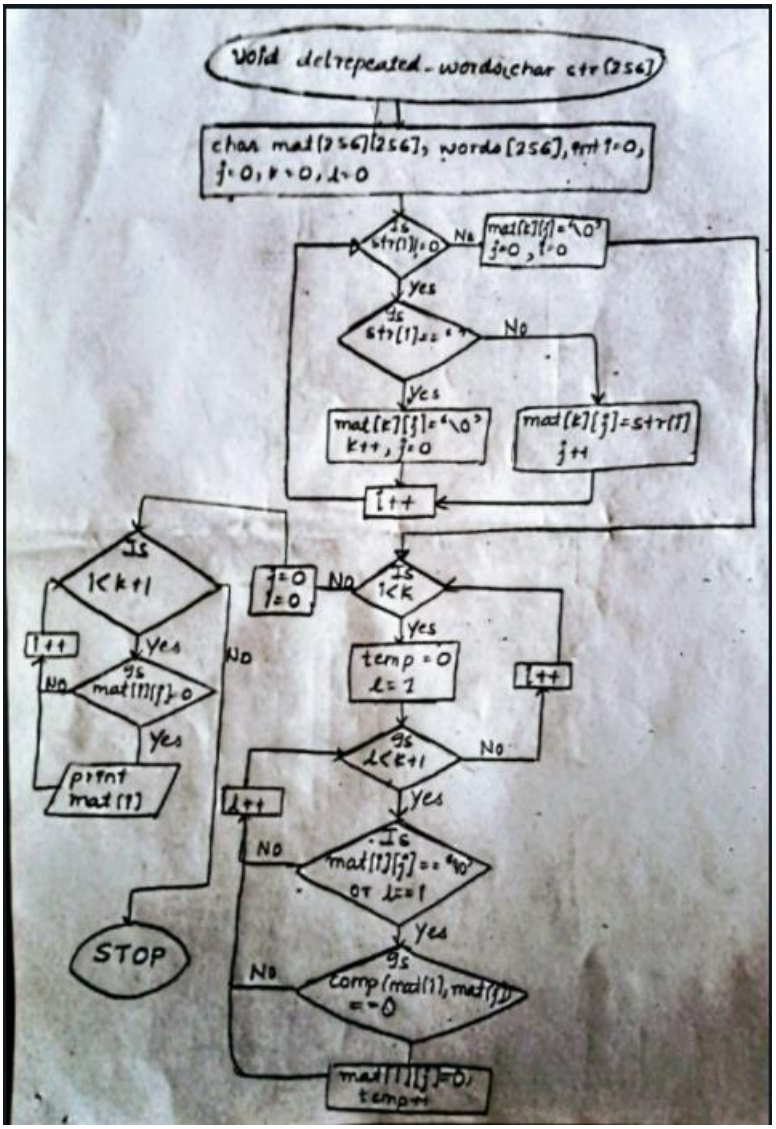
Program 3	
PROBLEM STATEMENT:	Write a program to delete all repeated words in string. Input: welcome to C programming class , welcome again to C class ! Output: welcome to C programming class , again !
ALGORITHM:	1. START 2. Define integer function comp with two character arrays str1[] and str2[] as parameters 3. Ctr = 0, I=0 4. If (str1[i]>str2[i] or str1[i]<str2[i]) count = 1 Go to step 7 5. i++ 6. Repeat 4 and 5 till str1[i] = 0 or str2[i] =0 7. Return count 8. Define void function delrepeated_words with a character array as parameters 9. char mat[100][100], words[100] 10.int i=0,j=0,k=0,l=0 11.If (str[i]==32) mat[k][j]=0 k++ j=0 Else mat[k][j]=str[i] j++

	12.i++ 13.Repeat 11 and 12 till str[i]!=0 14.mat[k][j]=0 j=0 i=0 15.temp=0 l=1 16.If(mat[i][j]==0 or l==1) Check if (comp(mat[i],mat[l])==0) mat[i][j] = 0 temp++ 17.l++ 18.Repeat 16 and 17 till l<k+1\ 19.i++ 20.Repeat 15,16,17,18 and 19 till i<k 21.i=0 j=0 22.Is mat[i][j]!=0? Print mat[i] 23.i++ 24.Repeat 22 and 23 till i<k+1 25.STOP

FLOWCHART:


START
Input string
str[100]
delrepeated_words(str)
STOP


int compr(char str1[],
char str2[])
ctr=0, i=0
Is
str1[i]!='\0' or
str2[i]!='\0'
No
Yes
i++
Is
str1[i]>str2[i]
or str2[i]>str1[i]
No
Yes
ctr=1
return
ctr



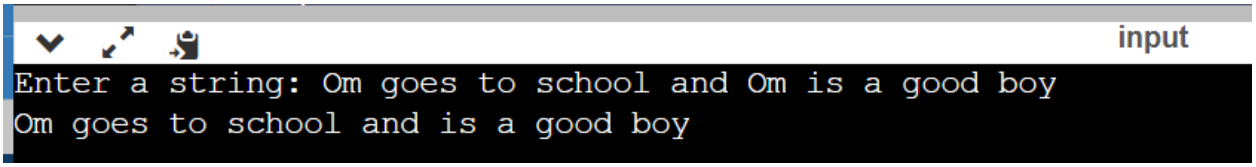
PROGRAM:

```
#include<stdio.h>
int comp(char str1[256],char str2[256])
{
   int ctr=0;
   for(int i=0;(str1[i]!='\0' || str2[i]!='\0');i++)
   {
```

```
        if((str1[i]>str2[i]) || (str1[i]<str2[i]))
        {
            ctr=1;
            break;
        }
    }
    return ctr;
}
void delrepeated_words(char str[256])
{
    char mat[256][256], words[256];
    int i=0,j=0,k=0,l=0;
    for(i=0;str[i]!='\0';i++)
    {
        if(str[i]==' ')
        {
            mat[k][j] = '\0';
            k++;
            j=0;
        }
        else
        {
            mat[k][j] = str[i];
            j++;
        }
    }
    mat[k][j]='\0';
    j=0;
    for(i=0;i<k;i++)
    {
        int temp=0;
        for(l=1;l<k + 1;l++)
        {
            if(mat[l][j] == '\0' || l == i)
            {
                continue;
            }
            if(comp(mat[i],mat[l])==0)
            {
                mat[l][j] = '\0';
```

```
            temp++;
          }
        }
    }
    j=0;
    for(i=0;i<k + 1;i++)
    {
        if(mat[i][j] == '\0')
            continue;
        else
            printf("%s ",mat[i]);
    }
}
int main()
{
    char str[256];
    printf("Enter a string: ");
    scanf(" %[^\n]",str);
    delrepeated_words(str);
    return 0;
}
```

RESULT:



CONCLUSION:	In the above experiment we learned about the basic syntax of strings and how to initialize them and how to read and print them. We learnt how we can pass them to arrays and functions. We learned about some functions like reversing, concatenating, finding the length of strings etc.