1) read from a terminal using scanf function and print using printf function.

```
#include <stdio.h>

int main()
{
  int x;
  int args;

  printf("Enter an integer: ");
  if (( args = scanf("%d", &x)) == 0) {
     printf("Error: not an integer\n");
  } else {
     printf("Read in %d\n", x);
  }
return 0;
}
```

Output: Enter an integer: 20
Read in 20

2) read a lines of text from a terminal using fgets function and print using puts function.

```
#include<stdio.h>
int main(){
        char name[20];
   printf("Enter name: ");
   fgets(name,sizeof(name),stdin);
   printf("name: ");
   puts(name);
   return 0;
}
```

Output: Enter name: biswajeet
name: biswajeet

3) Convert

a. Upper case to Lower case
b. Lower case to Upper case
c. Toggle case
d. Sentence case

a) upper case to lower case:

```
#include <stdio.h>
#include <string.h>
int main(){
  char s[100];
  int i;

  printf("Enter a string : ");
  gets(s);

  for (i = 0; s[i]!='\0'; i++) {
    if(s[i] >= 'A' && s[i] <= 'Z') {
      s[i] = s[i] + 32;
    }
  }

  printf("\nString in Lower Case = %s", s);
  return 0;
}
```

OUTPUT:

```
Enter a string : BISWAJEET

String in Lower Case = biswajeet
```

b) lower case to upper case:

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

```
  char s[100];
  int i;
  printf("Enter a string : ");
  gets(s);

  for (i = 0; s[i]!='\0'; i++) {
    if(s[i] >= 'a' && s[i] <= 'z') {
      s[i] = s[i] - 32;
    }
  }
  printf("\nString in Upper Case = %s", s);
  return 0;
}
```

OUTPUT:

Enter a string : hello world

String in Upper Case = HELLO WORLD

c) toggle case:

```
#include <stdio.h>
#include <string.h>
int main(){
    char Str[100];
    int i;
    printf("Enter any string: ");
    gets(Str);
    for (i = 0; Str[i]!='\0'; i++){
        if(Str[i] >= 'a' && Str[i] <= 'z'){
            Str[i] = Str[i] - 32;
        }
        else if(Str[i] >= 'A' && Str[i] <= 'Z'){
            Str[i] = Str[i] + 32;
        }
    }

    printf("\n The Given String after toggle case = %s", Str);
```

```
return 0;
}
```

OUTPUT:

Enter any string: HeLlO

The Given String after toggle case = hElLo

d) sentence case:

```
#include <stdio.h>
#include <ctype.h>
int main(){
     char str[100];
     printf("Enter a string : ");
   gets(str);
   str[0] = toupper(str[0]);
   printf("The string is: %s.",str);
   return 0;
}
```

OUTPUT:

Enter a string : hello programmers

The string is: Hello programmers

4) perform String Concatenation (With and Without String Handling Functions).

a) Without using string handling function:

```
#include<stdio.h>

int main()
{
 char str1[25],str2[25];
 int i=0,j=0;
 printf("\nEnter First String:");
 gets(str1);
 printf("\nEnter Second String:");
 gets(str2);
 while(str1[i]!='\0')
```

```
 i++;
 while(str2[j]!='\0')
 {
  str1[i]=str2[j];
  j++;
  i++;
 }
 str1[i]='\0';
 printf("\nConcatenated String is %s",str1);
return 0;
}
```

output: Enter First String:21

Enter Second String:23

Concatenated String is 2123

b) With using string function:

```
#include <stdio.h>
#include <string.h>

int main()
{
   char a[100], b[100];

   printf("Enter the first string\n");
   gets(a);

   printf("Enter the second string\n");
   gets(b);

   strcat(a,b);

   printf("String obtained on concatenation is %s\n",a);
```

```
    return 0;

}
```

output: Enter the first string
myschoolis
Enter the second string
tutorials
String obtained on concatenation is myschoolistutorials

5) perform String Reversal (With and Without String Handling Functions).

a) using string handling function:

```
#include <stdio.h>
#include <string.h>
int main()
{
  char s[100];

  printf("Enter a string to reverse\n");
  gets(s);

  strrev(s);

  printf("Reverse of the string: %s\n", s);

  return 0;
}
```

output: Enter a string to reverse

computer is an amazing device
Reverse of entered string is
ecived gnizama na si retupmoc

b) Without using string handling function:

```
#include <stdio.h>
int main()
{
  char s[1000], r[1000];
  int begin, end, count = 0;

  printf("Input a string\n");
  gets(s);



  while (s[count] != '\0')
    count++;

  end = count - 1;

  for (begin = 0; begin < count; begin++) {
    r[begin] = s[end];
    end--;
  }

  r[begin] = '\0';

  printf("%s\n", r);

  return 0;
}
```

output: Input a string
computer is an amazing device

ecived gnizama na si retupmoc

6) perform Substring Extraction (With and Without String Handling Functions).

a) using string handling function:

```
#include<stdio.h>
#include <string.h>

int main() {
   char string[50] = "Hello world";
   // Extract the first token
   char * token = strtok(string, " ");
   printf( " %s\n", token ); //printing the token
   return 0;
}
```

output: Hello

b) without using string handling function:

```
#include <stdio.h>
int findSubstring(char *str, char *substring);
int main()
{
    char str[40], substr[40];
    printf("Enter the string: ");
    gets(str);
    printf("Enter the substring: ");
    gets(substr);
    printf("findSubstring(): %d\n", findSubstring(str, substr));
    return 0;
}
int findSubstring(char *str, char *substr)
{
    /* write your code here */
    int i = 0, j = 0;
    while ((str[j] != '\0')||(substr[i] != '\0')) {
        if (substr[i] != str[j]) {
```

```
            j++;
            i = 0;
        }
        else {
            i++;
            j++;
        }
    }
    if (substr[i] == '\0')
        return 1;
    else
        return -1;
}
```

output: Enter the string: you are a coder
Enter the substring: are
findSubstring(): 1

7) copy one string into another and count the no of elements copied. (With and Without String Handling Functions).

a) With using string handling function:

```
#include<stdio.h>
#include<string.h>    // for using strcpy() function

int main(){
    char str1[100];
    char str2[100];
    int i;
    printf("Enter the string: ");
    gets(str2);
    strcpy(str1,str2);
    printf("\nThe copied string is: %s", str1);
    for(i=0; str2[i]!='\0'; i++)
      str1[i]=str2[i];
      str1[i]='\0';
    printf("\nNumber of characters = %d\n", i);
```

```
   return 0;
}
```

Output: Enter the string: man

The copied string is: man
Number of characters = 3

b) Without using string handling function:

```
#include<stdio.h>
//#define N 10

int main(){
char str1[80],str2[80];
int i;
printf("input a string:");
scanf("%s",str2);
for(i=0;str2[i]!='\0';i++)
str1[i]=str2[i]!='\0';i++)
str1[i]=str2[i];
str1[i]='\0';
printf("\n");
printf("original string:%s",str1);
printf("\nnumber of characters=%d\n",i);
return 0;
}
```

Output: input a string: man
original string: man
number of characters = 3

8) read a string and prints if it is a palindrome or not.

```
#include <stdio.h>
#include <string.h>

int main(){
```

```
    char string1[20];
    int i, length;
    int flag = 0;

    printf("Enter a string:");
    scanf("%s", string1);

    length = strlen(string1);

    for(i=0;i < length ;i++){
        if(string1[i] != string1[length-i-1]){
            flag = 1;
            break;
      }
}

    if (flag) {
        printf("%s is not a palindrome", string1);
    }
    else {
        printf("%s is a palindrome", string1);
    }
    return 0;
}
```

Output: Enter a string:wow
wow is a palindrome

9) read a line of text and count all occurrences of particular word.

```
#include <stdio.h>
#include <string.h>
#include <ctype.h>

int main()
{
    char string[100], word[20], unit[20], c;
    int i = 0, j = 0, count = 0;
```

```
   printf("Enter string: ");
   i = 0;
   do
   {
      fflush(stdin);
      c = getchar();
      string[i++] = c;

   } while (c != '\n');
   string[i - 1] = '\0';
   printf("Enter the word you want to find: ");
   scanf("%s", word);
   for (i = 0; i < strlen(string); i++)
   {
      while (i < strlen(string) && !isspace(string[i]) && isalnum(string[i]))
      {
         unit[j++] = string[i++];
      }
      if (j != 0)
      {
         unit[j] = '\0';
         if (strcmp(unit, word) == 0)
         {
            count++;
         }
         j = 0;
      }
   }

   printf("The number of times the word '%s' found in '%s' is '%d'.\n", word, string,
count);
return 0;
}
```

output: Enter string: hello world hello program hello C
Enter the word you want to find: hello
The number of times the word 'hello' found in 'hello world hello program hello C' is 3

10) read a string and rewrite it in the alphabetical order.

```
#include<stdio.h>
#include<string.h>

int main()
{
char str[20], k;
int i, j;

printf("Enter a string: \n");
scanf("%[^\n]", str);
for(i=0; str[i] != '\0'; i++)
{
for(j=i+1; str[j] != '\0'; j++)
{
if(str[i] > str[j])
{
k= str[i];
str[i] = str[j];
str[j] = k;

}
}
}
printf("%s", str);
printf("\n");
return 0;

}
```

Output: Enter a string:
blueberry
bbeelrruy

11) Print the Words Ending with Letter S.

```
#include <stdio.h>
#include <string.h>

char str[100];

int main()
{
   int i, t, j, len;

   printf("Enter a string : ");
   scanf("%[^\n]s", str);

   len = strlen(str);

   str[len] = ' ';

   for (t = 0, i = 0; i < strlen(str); i++)
   {
      if ((str[i] == ' ') && (str[i - 1] == 's'))
      {
         for (j = t; j < i; j++)
            printf("%c", str[j]);
         t = i + 1;
         printf("\n");
      }
      else
      {
         if (str[i] == ' ')
         {
            t = i + 1;
         }
      }
   }
return 0;
}
```

Output: Enter a string :
Welcome to Illumin8's C Programming Class, Welcome Again to C Class ! Illumin8's Class

12) Delete All Repeated Words in the line of text.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int main()
{
	char str[100], word[100], twoD[10][30];
	int i = 0, j = 0, k = 0, len1 = 0, len2 = 0, l = 0;

	printf ("Enter the string\n");
	gets (str);


	for (i = 0; str[i] != '\0'; i++)
	{
		if (str[i] == ' ')
		{
			twoD[k][j] = '\0';
			k ++;
			j = 0;
		}
		else
		{
			twoD[k][j] = str[i];
			j ++;
		}
	}

	twoD[k][j] = '\0';

	j = 0;
```

```
	for (i = 0; i < k; i++)
	{
		int present = 0;
		for (l = 1; l < k + 1; l++)
		{
			if (twoD[l][j] == '\0' || l == i)
			{
				continue;
			}

			if (strcmp (twoD[i], twoD[l]) == 0) {
				twoD[l][j] = '\0';
				present = present + 1;
			}
		}

	}

	j = 0;

	for (i = 0; i < k + 1; i++)
	{
		if (twoD[i][j] == '\0')
			continue;
		else
			printf ("%s ", twoD[i]);
	}

	printf ("\n");

	return 0;
}
```

Output: Enter the string:
Welcome to Sanfoundry C Class, Welcome to Java Programming, Welcome to C++ class
Welcome to Sanfoundry C Class, Java Programming, C++ class