1.Write a program to find biggest among three numbers using pointer

```c
#include <stdio.h>
int main()
{
    int num1, num2, num3;
    int *p1, *p2, *p3;

    //taking input from user
    printf("Enter First Number: ");
    scanf("%d",&num1);
    printf("Enter Second Number: ");
    scanf("%d",&num2);
    printf("Enter Third Number: ");
    scanf("%d",&num3);

    //assigning the address of input numbers to pointers
    p1 = &num1;
    p2 = &num2;
    p3 = &num3;
    if(*p1 > *p2)
    {
    if(*p1 > *p3)
    {
        printf("%d is the largest number", *p1);
    }
    else
    {
     printf("%d is the largest number", *p3);
    }
    }
    else
    {
    if(*p2 > *p3)
    {
         printf("%d is the largest number", *p2);
    }
    else
    {
         printf("%d is the largest number", *p3);
    }
    }
    return 0;
}
```

2. Write a program to find the sum of all the elements of an array using pointers.

```c
#include<stdio.h>
int main()
{
   int array[5];
   int i,sum=0;
   int *ptr;

   printf("\nEnter array elements (5 integer values):");
   for(i=0;i<5;i++)
      scanf("%d",&array[i]);

   /* array is equal to base address
    * array = &array[0] */
   ptr = array;

   for(i=0;i<5;i++)
   {
      //*ptr refers to the value at address
      sum = sum + *ptr;
      ptr++;
   }

   printf("\nThe sum is: %d",sum);
}
```

3. Write a program to swap value of two variables using pointer.

```c
#include <stdio.h>

// function to swap the two numbers
void swap(int *x,int *y)
{
    int t;
     t   = *x;
    *x   = *y;
    *y   =  t;
}

int main()
{
    int num1,num2;

    printf("Enter value of num1: ");
    scanf("%d",&num1);
    printf("Enter value of num2: ");
    scanf("%d",&num2);
```

```
    //displaying numbers before swapping
    printf("Before Swapping: num1 is: %d, num2 is: %d\n",num1,num2);

    //calling the user defined function swap()
    swap(&num1,&num2);

    //displaying numbers after swapping
    printf("After  Swapping: num1 is: %d, num2 is: %d\n",num1,num2);

    return 0;
}
```

4. Write a program to read a sentence and count the number of characters &words in that sentence.

```
#include<stdio.h>

int main()
{

 int cword=0, cchar=0, i;
 char str[50];

 printf("Please, Enter the string = ");
 gets(str);

 for(i=0; str[i]!=NULL; i++)
 {
   cchar++;
   if(str[i]==' ')
    {
      cwords++;
    }
 }

printf("\nThe Number of characters =  %d",cchar);
 printf("\nThe Number of words =  % d",cwords+1);

 return 0;
}
```

5. Write a program to read a sentence & delete all the white spaces. Replace all “.” by “:”

```
#include <string.h>

int main()
{
    char s[1000];
    int  i,k=0;
     printf("Enter  the string : ");
    gets(s);
```

```
    for(i=0;s[i];i++)
    {
        s[i]=s[i+k];


        if(s[i]==' '|| s[i]=='\t')
        {
          k++;
          i--;
        }

    }
    printf("string after removing all blank spaces:");

     printf("%s",s);


    return 0;
}
```

6. Write a program to copy one string to another string with and without using string handling function.

Without function

```
#include<stdio.h>

int main() {
   char s1[100], s2[100];
   int i;

   printf("\nEnter the string :");
   gets(s1);

   i = 0;
   while (s1[i] != '\0') {
      s2[i] = s1[i];
      i++;
   }

   s2[i] = '\0';
   printf("\nCopied String is %s ", s2);

   return (0);
}
```

1. Scan Entered String From **Left to Right , Character by Character**.

2. In Each Iteration **Copy One Character** To New String Variable.
3. As soon as Source or Original String Ends , Process of Coping Character Stops but **we still haven't Copied NULL Character** into new String so , **Append Null Character to New String.**

With library

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

void main()
{
    char str1[100],str2[50];

    printf("Enter string str1\n");
    gets(str1);

    strcpy(str2,str1);
    printf("Copied String(str2) is %s",str2);
}
```

7. Write a program to concatenate two strings.

```c
#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;
}
```

8. Write a program to compare two strings.

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

```
   char str1[20];  // declaration of char array
   char str2[20];  // declaration of char array
   int value; // declaration of integer variable
   printf("Enter the first string : ");
   scanf("%s",str1);
   printf("Enter the second string : ");
   scanf("%s",str2);
   // comparing both the strings using strcmp() function
   value=strcmp(str1,str2);
   if(value==0)
   printf("strings are same");
   else
   printf("strings are not same");
   return 0;
}
```

- We have declared two arrays of char type, i.e., str1 and str2. We take the user input as strings.
- We compare the strings by using the **strcmp()** function, i.e., **strcmp(str1,str2).** This function will compare both the strings str1 and str2. If the function returns 0 value means that both the strings are same, otherwise the strings are not equal.

9. Write a program to sort 5 string words stored in an array of pointers.

```
#include <stdio.h>
#include<string.h>
int main()
{
   char str1[20];  // declaration of char array
   char str2[20];  // declaration of char array
   int value; // declaration of integer variable
   printf("Enter the first string : ");
   scanf("%s",str1);
   printf("Enter the second string : ");
   scanf("%s",str2);
   // comparing both the strings using strcmp() function
   value=strcmp(str1,str2);
   if(value==0)
   printf("strings are same");
   else
   printf("strings are not same");
   return 0;
}
```