One Dimensional Array:

- Sort the 'n' elements of an array in Descending order

```
#include<stdio.h>

int main(){
   int arr[5],i,j,temp=0;
   int *ptr;

   ptr = arr;

   for(i=0;i<5;i++){
      printf("Enter a number :: ");
      scanf("%d",ptr+i);
   }

   for(i=0;i<5;i++){
      for(j=0;j<5;j++){
         if(*(ptr + i) > *(ptr + j)){
            temp = *(ptr + j);
            *(ptr+j) = *(ptr + i);
            *(ptr+i) = temp;
         }
      }
   }
   printf("Elements in Descending order :: ");
```

```
    for(i=0;i<5;i++){
        printf("%d ",*(ptr + i));
    }
}
```

Output:

```
Enter a number :: 5
Enter a number :: 9
Enter a number :: 8
Enter a number :: 3
Enter a number :: 2
Elements in Descending order :: 9 8 5 3 2
```

- Find the second largest and smallest element in an array

```
#include<stdio.h>

int main(){
   int arr[5],i,j,temp=0;
   int *ptr;

   ptr = arr;

   for(i=0;i<5;i++){
      printf("Enter a number :: ");
      scanf("%d",ptr+i);
   }

   for(i=0;i<5;i++){
      for(j=0;j<5;j++){
        if(*(ptr + i) > *(ptr + j)){
           temp = *(ptr + j);
           *(ptr+j) = *(ptr + i);
           *(ptr+i) = temp;
        }
      }
   }
  printf("Second largest number :: %d\n",*(ptr+1));
  printf("Smallest number :: %d",*(ptr+4));
```

```
}
```

Output:

```
Enter a number :: 9
Enter a number :: 8
Enter a number :: 3
Enter a number :: 2
Enter a number :: 1
Second largest number :: 8
Smallest number :: 1
```

Two Dimensional Array:

- Print the leading diagonal, upper triangular and lower triangular elements of [mxm] array.

```
#include<stdio.h>

int main(){
int arr[3][3],i,j,s=0,k=0;
int (*ptr)[3];

ptr = arr;

for(i=0;i<3;i++){
   for(j=0;j<3;j++){
      printf("Enter a number :: ");
     scanf("%d",(*(ptr+i)+j));
   }
}
printf("printing Diagonal...\n\n");
for(i=0;i<3;i++){
   j = i ;
   k=0;
      while(k<s){
         printf(" ");
         k++;
      }
```

```
        printf("%d ",*(*(ptr+i)+j));

     s++;
     printf("\n");
    }

printf("\n\nprinting lower triangle...\n\n");
for(i=0;i<3;i++){
    for(j=0;j<=i;j++){
        printf("%d ",*(*(ptr+i)+j));
    }
    printf("\n");
}
s=0;
printf("\n\nprinting upper triangle...\n\n");
for(i=0;i<3;i++){
    k=0;
    for(j=i;j<3;j++){

        while(k<s){
            printf(" ");
            k++;
        }
        printf("%d ",*(*(ptr+i)+j));
    }
    s+=2;
```

```
    printf("\n");
}

}
```

Output:

```
Enter a number :: 1
 Enter a number :: 2
 Enter a number :: 3
 Enter a number :: 4
 Enter a number :: 5
 Enter a number :: 6
 Enter a number :: 7
 Enter a number :: 8
 Enter a number :: 9
 printing Diagonal...

1
 5
  9


printing lower triangle...

1
4 5
7 8 9
```

printing upper triangle...

1 2 3

  5 6

    9

- Find the maximum & minimum element in each row and each coloumn of mxm array

```
#include<stdio.h>

int main(){
int arr[3][3],i,j,k,l,temp=0;
int (*ptr)[3];

ptr = arr;

for(i=0;i<3;i++){
   for(j=0;j<3;j++){
       printf("Enter a number :: ");
      scanf("%d",(*(ptr+i)+j));
   }
}

for(i=0;i<3;i++){
   for(j=0;j<3;j++){
     for(k=0;k<3;k++){
        for(l=0;l<3;l++){

           if(*(*(ptr+i)+j) < *(*(ptr+k)+l)){

              temp = *(*(ptr+k)+l);
              *(*(ptr+k)+l) = *(*(ptr+i)+j);
```

```
                *(*(ptr+i)+j) = temp;
            }
        }
    }
  }
}
printf("Sorted matrix...\n\n");
for(i=0;i<3;i++){
   for(j=0;j<3;j++){
      printf("%d ",*(*(ptr+i)+j));
   }
   printf("\n");
}

printf("\n in first row...\n");
printf("smallest number :: %d\nlargest number ::
%d",*(*(ptr+0)+0),*(*(ptr+0)+2));

printf("\n in second row...\n");
printf("smallest number :: %d\nlargest number ::
%d",*(*(ptr+1)+0),*(*(ptr+1)+2));

printf("\n in third row...\n");
printf("smallest number :: %d\nlargest number ::
%d",*(*(ptr+2)+0),*(*(ptr+2)+2));

}
```

Output:

```
Enter a number :: 3
Enter a number :: 5
Enter a number :: 6
Enter a number :: 2
Enter a number :: 1
Enter a number :: 4
Enter a number :: 7
Enter a number :: 8
Enter a number :: 9
Sorted matrix...

1 2 3
4 5 6
7 8 9

 in first row...
smallest number :: 1
largest number :: 3
 in second row...
smallest number :: 4
largest number :: 6
 in third row...
smallest number :: 7
largest number :: 9
```

- Perform matrix multiplication between two mxm array.

```
#include<stdio.h>

int main(){
int arr[3][3],arr1[3][3],mul[3][3],i,j,k,l,temp=0;
int (*ptr)[3],(*ptr1)[3],(*res)[3];

ptr = arr;
ptr1 = arr1;
res = mul;
printf("Enter elements in first array... \n");
for(i=0;i<3;i++){
   for(j=0;j<3;j++){
       printf("Enter a number :: ");
      scanf("%d",(*(ptr+i)+j));
   }
}

printf("Enter elements in second array... \n");
for(i=0;i<3;i++){
   for(j=0;j<3;j++){
       printf("Enter a number :: ");
      scanf("%d",(*(ptr1+i)+j));
   }
}
```

```
for(i=0;i<3;i++){
    for(j=0;j<3;j++){
        *(*(res+i)+j) = 0;
      for(k=0;k<3;k++){
         *(*(res+i)+j) += *(*(ptr+i)+k) * *(*(ptr+k)+j);
          }
      }
    }

printf("Matrix multiplication...\n\n");
for(i=0;i<3;i++){
    for(j=0;j<3;j++){
        printf("%d ",*(*(res+i)+j));
    }
    printf("\n");
}



 return 0;
}
```

Output:

```
Enter elements in first array...
Enter a number :: 1
Enter a number :: 2
Enter a number :: 3
Enter a number :: 4
Enter a number :: 5
Enter a number :: 6
Enter a number :: 7
Enter a number :: 8
Enter a number :: 9
Enter elements in second array...
Enter a number :: 1
Enter a number :: 2
Enter a number :: 3
Enter a number :: 4
Enter a number :: 5
Enter a number :: 6
Enter a number :: 7
Enter a number :: 8
Enter a number :: 9
Matrix multiplication...
30 36 42
66 81 96
102 126 150
```

- **String Manipulations using Pointers**
- Write a C Program to convert
- Lower case to Upper case

```
#include<stdio.h>

int main(){
int i=0;
char *ptr,ch[15];

ptr = ch;

printf("Enter a string in lower case :: ");
gets(ptr);

printf("String in upper case :: ");
while(*(ptr+i)!= '\0'){
   printf("%c",*(ptr+i)-32);
   i++;
}
 return 0;
}
```

Output:

Enter a string in lower case :: shreetik
String in upper case :: SHREETIK

- Upper case to lower case

```
#include<stdio.h>

int main(){
int i=0;
char *ptr,ch[15];

ptr = ch;

printf("Enter a string in Upper case :: ");
gets(ptr);

printf("String in lower case :: ");
while(*(ptr+i)!= '\0'){
   printf("%c",*(ptr+i)+32);
   i++;
}
 return 0;
}
```

Output:

```
Enter a string in Upper case :: SHREETIK
String in lower case :: shreetik
```

- Toggle case

```
#include<stdio.h>

int main(){
int i=0;
char *ptr,ch[15];

ptr = ch;

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

printf("String in Toggle case :: ");
while(*(ptr+i)!= '\0'){
   if(*(ptr+i)>=65 && *(ptr+i)<=90){
      printf("%c",*(ptr+i)+32);
   }
   else{
      printf("%c",*(ptr+i)-32);
   }
   i++;
}
 return 0;
}
```

Output:

Enter a string :: ShRee

String in Toggle case :: sHrEE

- Write a C program to read a string and prints if it is a palindrome or not.

```
#include<stdio.h>

int main(){
int c=0,i=0,p=0;
char *ptr,ch[15];

ptr = ch;

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

while(*(ptr+i) != '\0'){
 c++;
 i++;
}
c = c-1;//deleting NULL value index

i=0;
while(i<=c){
  if(*(ptr+i) == *(ptr+c-i)){
      p++;
  }
  i++;
}
```

```
c = c+1; //adding NULL value index

if(p == c){
   printf("its a palindrome string");
}
else{
   printf("its not a palindrome string");
}
 return 0;
}
```

Output:

```
Enter a string :: cts
its not a palindrome string

Enter a string :: abcdcba
its a palindrome string
```

- **Functions using Pointers**

- Check Prime and Armstrong Number by making function

```
#include<stdio.h>
void checkPrime();
void checkArmstrong();

int main(){
   int num;

   printf("enter a number :: ");
   scanf("%d",&num);
   checkPrime(&num);
  checkArmstrong(&num);

 return 0;
}

void checkPrime(int *ptr){
    int i=1,c=0;
    while(i<=*ptr){
     if(*ptr % i == 0){
        c++;
     }
     i++;
```

```
    }
    if(c == 2){
        printf("%d is a prime number\n\n",*ptr);
    }
    else{
        printf("%d is not a prime number\n\n",*ptr);
    }
}

void checkArmstrong(int *ptr){
     int copy,num,d=0,sum=0;
     num = *ptr;
     copy = *ptr;

    while(num > 0){
        d = num % 10;
        sum += d * d * d;
        num /= 10;
    }
    if(sum == copy)
    printf("%d is a armstrong number\n\n",copy);
    else
    printf("%d is not a armstrong number\n\n",copy);
}
```

Output:

11 is a prime number

11 is not a armstrong number

153 is not a prime number

153 is a armstrong number

- Reverse a sentence using String Functions

```
#include<stdio.h>
void reverse();

int main(){
  char str[30];
  printf("Enter a sentence :: ");
  gets(str);
  printf("Reverse Sentence :: ");
  reverse(str);

 return 0;
}

void reverse(char *ptr){
    int i=0,c=0;
    while(*(ptr+i) != '\0'){
       c++;
        i++;
    }
    i=0;
    c = c-1;

    while(c>=i){
       printf("%c",*(ptr+c));
       c--;
```

```
    }

}
```

Output:

```
Enter a sentence :: shree tik
Reverse Sentence :: kit eerhs
```

- Calculate the power of a number using recursion

```
#include<stdio.h>
int res=1,s;
int power(int *n,int *p){
 if(*p == 0){
    return 1;
 }
 else{
   s = *p-1;
    return(*n * power(n,&s));
 }

 }

int main(){
int num,pow,mul;

printf("Enter number :: ");
scanf("%d",&num);
printf("Enter power :: ");
scanf("%d",&pow);
 mul = power(&num,&pow);
 printf("%d ^ %d = %d",num,pow,mul);
   return 0;
}
```

Output:

Enter number :: 5

Enter power :: 3

5 ^ 3 = 125

- **Structures using Pointers**

- Store Information(name, roll and marks) of a Student Using Structure

```
#include<stdio.h>

typedef struct student{
   char name[15];
   int roll;
  int mark;

}stu;


int main(){
   stu s1;
   stu *ptr;
   ptr = &s1;

   printf("Enter student name :: ");
   gets(ptr->name);
   printf("enter roll number :: ");
   scanf("%d",&ptr->roll);
   printf("enter mark of the student :: ");
   scanf("%d",&ptr->mark);
```

```
    printf("\n\nStudent information\n");
    printf("--------------------\n\n");

    printf("Student name :: %s\n",ptr->name);
    printf("Student Roll no :: %d\n",ptr->roll);
    printf("Student Mark :: %d\n",ptr->mark);

    return 0;
}
```

Output:

```
Enter student name :: Shreetik Sahani
enter roll number :: 55
enter mark of the student :: 45


Student information
--------------------

Student name :: Shreetik Sahani
Student Roll no :: 55
Student Mark :: 45
```