1. Write a program to read Title, Author and Price of 5 books using array of structures.Display the records in ascending order of Price.

Program :

```
#include <stdio.h>
//#include <conio.h>
struct book
{
int price;
char title[80];
char author[80];
};
void accept(struct book list[80]);                              //func declare
void display(struct book list[80]);
void bsortAsc(struct book list[80]);
void main()
{
struct book data[20];
int n;
//clrscr();
accept(data);                                                   //func call
bsortAsc(data);
display(data);
//getch();
}
void accept(struct book list[5])                                // func initialize
{
int i;
for (i = 0; i <5; i++)
{
printf("\nEnter title : ");
scanf("%s", &list[i].title);
printf("Enter Author name: ");
scanf("%s",&list[i].author);
printf("Enter price : ");
scanf("%d", &list[i].price);
}
}
void display(struct book list[80])
{
int i;
printf("\n\nTitle\t\tAuthor\t\tprice\n");
printf("----------------------------------------\n");
for (i = 0; i<5; i++)
{
printf("%s\t\t%s\t\t%d\n", list[i].title, list[i].author, list[i].price);
}
}
void bsortAsc(struct book list[80])
{
int i, j;
struct book temp;
for (i = 0; i <5 ; i++)
{
for (j = 0; j < (5 -i); j++)
{
if (list[j].price >list[j + 1].price)
{
temp = list[j];
list[j] = list[j + 1];
list[j + 1] = temp;
}
}
}}
```

2. Implement a program to perform addition of two matrices.

Program:

```
#include <stdio.h>
int main()
{
   int row,col, i, j, first[10][10], second[10][10], sum[10][10];
   printf("Enter the number of rows and columns of matrix\n");
   scanf("%d %d", &row, &col);
   printf("Enter the elements of first matrix\n");
   for (i = 0; i < row; i++)
      for (j = 0; j < col; j++) scanf("%d", &first[i][j]);
   printf("Enter the elements of second matrix\n");
   for (i = 0; i < row; i++)
      for (j = 0; j < col; j++) scanf("%d", &second[i][j]);
   printf("Sum of entered matrices:-\n");
   for (i = 0; i <row; i++)
   {
      for (j = 0; j < col; j++)
      {
         sum[i][j] = first[i][j] + second[i][j];
         printf("%d\t", sum[i][j]);
      }
      printf("\n");
   }
   return 0;
}
```

3. Write a program to check whether a word is palindrome or not.

Program :

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
int main(){
   char str[20];
   int i, len, temp=0;
   int flag = 0;
   printf("Enter a string:");
   scanf("%s", str);
   len = strlen(str);
   for(i=0;i < len ;i++){
      if(str[i] != str[len-i-1]){
         temp = 1;
      break;
   }
 }

   if (temp==0) {
      printf("String is a palindrome");
   }
   else {
      printf("String is not a palindrome");
   }
   return 0;
}
```

4. What are bitwise and logical operators in C ?

6. Implement a program to find transpose of a matrix.

Program :

```
#include <stdio.h>
int main() {
  int a[10][10], transpose[10][10], r, c;
  printf("Enter rows and columns: ");
  scanf("%d %d", &r, &c);

  // asssigning elements to the matrix
  printf("\nEnter matrix elements:\n");
  for (int i = 0; i < r; ++i)
  for (int j = 0; j < c; ++j) {
    printf("Enter element a%d%d: ", i + 1, j + 1);
    scanf("%d", &a[i][j]);
  }

  // printing the matrix a[][]
  printf("\nEntered matrix: \n");
  for (int i = 0; i < r; ++i)
  for (int j = 0; j < c; ++j) {
    printf("%d  ", a[i][j]);
    if (j == c - 1)
    printf("\n");
  }

  // computing the transpose
  for (int i = 0; i < r; ++i)
  for (int j = 0; j < c; ++j) {
    transpose[j][i] = a[i][j];
  }

  // printing the transpose
  printf("\nTranspose of the matrix:\n");
  for (int i = 0; i < c; ++i)
  for (int j = 0; j < r; ++j) {
    printf("%d  ", transpose[i][j]);
    if (j == r - 1)
    printf("\n");
  }
  return 0;
}
```

11. Write a program to print Fibonacci series.

Program :

```
#include <stdio.h>
#include <conio.h>
int main() {
  int i, n;
  // initialize first and second terms
  int t1 = 0, t2 = 1;
  // initialize the next term (3rd term)
  int nextTerm = t1 + t2;
   clrscr();
  // get no. of terms from user
  printf("Enter the number of terms: ");
  scanf("%d", &n);
  // print the first two terms t1 and t2
  printf("Fibonacci Series: %d, %d, ", t1, t2);
  // print 3rd to nth terms
  for (i = 3; i <= n; ++i) {
    printf("%d ", nextTerm);
    t1 = t2;
    t2 = nextTerm;
    nextTerm = t1 + t2;
   }
  return 0;
  getch();
}
```

Output:

```
Enter the number of terms: 10
Fibonacci Series: 0 1 1 2 3 5 8 13 21 34
```

14. Write a C program to perform multiplication of two matrices.

Program :

```
#include<stdio.h>
#include<conio.h>
int main(){
 int a[10][10],b[10][10],mul[10][10],r,c,i,j,k;
 clrscr();
 printf("enter the number of row=");
 scanf("%d",&r);
 printf("enter the number of column=");
 scanf("%d",&c);
 printf("enter the first matrix element=\n");
 for(i=0;i<r;i++)
 {
  for(j=0;j<c;j++)
  {
    scanf("%d",&a[i][j]);
   }
 }
 printf("enter the second matrix element=\n");
 for(i=0;i<r;i++)
 {
  for(j=0;j<c;j++)
  {
    scanf("%d",&b[i][j]);
   }
 }
printf("multiply of the matrix=\n");
for(i=0;i<r;i++)
{
 for(j=0;j<c;j++)
 {
  mul[i][j]=0;
  for(k=0;k<c;k++)
  {
   mul[i][j]+=a[i][k]*b[k][j];
  }
 }
}
//for printing result
 for(i=0;i<r;i++)
 {
 for(j=0;j<c;j++)
 {
  printf("%d\t",mul[i][j]);
 }
  printf("\n");
 }
  return 0;
  getch();
}
```

Output:

```
enter the number of row=3
enter the number of column=3
enter the first matrix element=
1 1 1
2 2 2
3 3 3
enter the second matrix element=
1 1 1
2 2 2
3 3 3
multiply of the matrix=
```

```
6 6 6
12 12 12
18 18 18
```

16. Explain the term recursion. Write a program to find the power of x raised to n that is:x^n, using recursive function.

Ans.

A function that calls itself is known as a recursive function. And, this technique is known as recursion.The recursion continues until some condition is met to prevent it.

To prevent infinite recursion, if...else statement (or similar approach) can be used where one branch makes the recursive call, and other doesn't.

```
#include <stdio.h>
#include <conio.h>
int power(int n1, int n2);
int main() {
   int base, a, result;
   clrscr();
   printf("Enter base number: ");
   scanf("%d", &base);
   printf("Enter power number(positive integer): ");
   scanf("%d", &a);
   result = power(base, a);
   printf("%d^%d = %d", base, a, result);
   getch();
   return 0;
}
int power(int base, int a) {
   if (a != 0)
      return (base * power(base, a - 1));
   else
      return 1;
}
```

Output:

```
Enter base number: 3
Enter power number(positive integer): 4
3^4 = 81
```

18. Write a program to print the following

pattern.A

B

B

C

C

C

D D D D

Program:

```
#include<stdio.h>
void main()
{
   int i, j;
   clrscr();
   for(i=1;i<=5;i++)
   {
      for(j=1;j<=i;j++)
      {
         printf("%c ",'A'-1 + i);
      }
      printf("\n");
   }
   getch();
}
```

19. Write a program to find largest element of an 1D array.

```
#include <stdio.h>
int main(){
    int num;
    int arr[25];

    // Asking for input
    printf("Please enter total no. of elements[1 to 25]: ");
    scanf("%d", &num);
    for (int i = 0; i < num; ++i){
        printf("%d.Enter the number: ", i + 1);
        scanf("%d", &arr[i]);
    }

    for (int i = 0; i < num; ++i){
        if (arr[0] < arr[i]){
            arr[0] = arr[i];
        }
    }
    printf("Largest element of the array is: %d", arr[0]);
    return 0;
}
```

Output:

```
Please enter total no. of elements[1 to 25]: 5
1.Enter the number: 12
2.Enter the number: 17
3.Enter the number: 5
4.Enter the number: 44
5.Enter the number: 13
Largest element of the array is: 44
```

20.	Write a Program to calculate and display sum of all the elements of the matrix.

```
#include<stdio.h>
int main()
{
    int a[10][10],r,c,sum=0,i,j;
    printf("/*How Many Rows You Want To \nEnter in Matrix*/\nEnter Limit : ");
    scanf("%d",&r);
    printf("\n/*How Many Columns You Want To \nEnter in Matrix*/\nEnter
Limit : ");
    scanf("%d",&c);
    printf("\nEnter Elements for Matrix of Size %d*%d:\n\n",r,c);
    for(i=0;i<r;i++){
        for(j=0;j<c;j++)
        {
            scanf("%d",&a[i][j]);
        }
        printf("\n");
      }
    for(i=0;i<r;i++)
        for(j=0;j<c;j++)
            sum=sum+a[i][j];
    printf("\nSum of All Elements in Matrix = %d",sum);
    return 0;
}
```

21. Define a structure called player with data members as player name, team name,batting average. Store and display the information of at least 10 players.

```
#include<stdio.h>
#include<conio.h>
struct player
{
        char pname[20];
        char tname[20];
        float bavg;
};
int main()
{
        struct player s[10];
        int i,j,n=10;
        float p;
        clrscr();
        for(i=0;i<n;i++)
        {
                printf("\nEnter PName TName BAvg for player-%d = ",i+1);
                scanf("%s %s %f",s[i].pname,s[i].tname,&p);
                s[i].bavg=p;
        }
        for(i=0;i<n;i++)
        {
                printf("\n%s %s %.2f",s[i].pname,s[i].tname,s[i].bavg);
        }
        getch();
        return 0;
}
```

}

23. Write a program to display the following for the user specified number of lines.

```
     *
    **
   ***
  ****
 *****
******
```

Program:

```
#include <stdio.h>
#include <conio.h>
int main() {
  int i, space, rows, k;
  clrscr();
  printf("Enter the number of rows: ");
  scanf("%d", &rows);
      printf("\n");
  for (i = 1; i <= rows; i++ ) {
    for (space = 1; space <= rows - i+1; space++) {
      printf("  ");
    }
    for(k=1; k<=i;k++){
       printf("* ");
    }
    printf("\n");
  }
   getch();
  return 0;
```

27. Write a program in C to find the reverse of a given string without using inbuilt stringfunction.

Program:

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

void main()
{
   int i,n;
```

```
   char str[20];
   printf("Enter the String to get reversed: ");
   gets(str);
   n=strlen(str);
   printf("\nReversed string is ");
   for(i=n-1;i>=0;i--)
   {
     printf("%c",str[i]);
   }
}
```

Output:

```
Enter the String to get reversed: java
Reversed string is avaj
```

29. Write a program to store and display at least 10 records of the name, roll number andfees of a student using structure.

Program:

```
#include<stdio.h>
#include<conio.h>
struct student
{
        char name[20];
        int roll;
        int fees;
};
int main()
{
        struct student s[10];
        int i,j,n=10;
        clrscr();
        for(i=0;i<n;i++)
        {
                printf("\nEnter Name Roll No. Fees of student-%d = ",i+1);
                scanf("%s %d %d",s[i].name,s[i].roll,s[i].fees);
                }
        for(i=0;i<n;i++)
        {
                printf("\n%s %d %d",s[i].name,s[i].roll,s[i].fees);
        }
        getch();
        return 0;

}
```

31. Explain String function for the following operations with example.

- Copy string from source to destination.

strcpy() takes two strings as arguments and character by character (including \0) copies the content of string Src to string Dest, character by character.

```
#include <stdio.h>
#include <string.h>
#include <conio.h>
int main()
{
 char Src[15]= "DevOps";
 char Dest[15] = "";
 clrscr();
 strcpy(Dest, Src);   // calling strcpy function
 printf("After copying\n");
 printf("Source string: %s \n", Src);
 printf("Destination string: %s \n", Dest);
 getch();
 return 0;
}
```

Output

```
After copying
Source string: DevOps
Destination string: DevOps
```

- Merging of two strings.

The concatenation of strings is a process of combining two strings to form a single string. If there are two strings, then the second string is appended(added) at the end of the first string.

```
#include <stdio.h>
#include<string.h>
int main()
{
 char s1[20]; // declaration of char array
 char s2[20]; // declaration of char array
 printf("Enter the first string : ");
 scanf("%s", &s1);
 printf("\nEnter the second string :");
 scanf("%s",&s2);
 strcat(s1,s2);
 printf("\nThe concatenated string is : %s",s1);
   return 0;
 }
```

Output:

```
Enter the first string : Game
Enter the second string : Developer
The concatenated string is : GameDeveloper
```

33. Write a program to print the following pattern. (Note- Not only 4 lines, it should print N lines taken from the user.)

```
      A
    B   B
  C   C   C
D   D   D   D
```

Program:

```
#include<stdio.h>
#include<conio.h>
void main()
{
  int i, j,k,n;
  clrscr();
  printf("Enter any Number:");
  scanf("%d",&n);
  for(i=1;i<=n;i++)
  {
    for(k=n;k>=i;k--)
    printf(" ");
     for(j=1;j<=i;j++)
     {
        printf("%c ",'A'-1 + i);
     }
     printf("\n");
  }
  getch();
}
```

34. Write a C-program to create array of structures in order to store details of almost 100books. The book details are book name, book price, book page number and book author name.

```
Program :
#include <stdio.h>
//#include <conio.h>
struct book
{
int price;
char title[80];
char author[80];
};
void accept(struct book list[100]);                                    //func declare
void display(struct book list[100]);
void main()
{
struct book data[20];
```

```
int n;
//clrscr();
accept(data);                                                          //func call
display(data);
//getch();
}
void accept(struct book list[100])                                     // func initialize
{
int i;
for (i = 0; i <100; i++)
{
printf("\nEnter title : ");
scanf("%s", &list[i].title);
printf("Enter Author name: ");
scanf("%s",&list[i].author);
printf("Enter price : ");
scanf("%d", &list[i].price);
}
}
void display(struct book list[100])
{
int i;
printf("\n\nTitle\t\tAuthor\t\tprice\n");
printf("----------------------------------------\n");
for (i = 0; i<100; i++)
{
printf("%s\t\t%s\t\t%d\n", list[i].title, list[i].author, list[i].price);
}
}
```

35. Write a program that will accept two-dimensional square matrix and find the sum ofdiagonal elements. (Note- sum of diagonal elements should be calculated for both sides).

Program:

```
#include<stdio.h>
#include<conio.h>
int main()
{
        int i, j, rows, columns, a[10][10],principal=0,secondary=0 ;
```

```
        clrscr();
        printf("\n Please Enter Number of rows and columns  :  ");
        scanf("%d %d", &i, &j);
        printf("\n Please Enter the Matrix Elements \n");
        for(rows = 0; rows < i; rows++)
        {
                for(columns = 0;columns < j;columns++)
        {
                scanf("%d", &a[rows][columns]);
        }
        }
        for(rows = 0; rows < i; rows++)
        {
                for(columns = 0;columns < j;columns++)
        {
         // Condition for principal diagonal
         if (rows == columns)
            principal += a[rows][columns];

         // Condition for secondary diagonal
         if ((rows+columns) == (i - 1))
            secondary += a[rows][columns];
        }
        }
        printf("\n The Primary and secondary principal diagonal sum =  %d %d",
principal,secondary );

        printf("\n The Sum of All Diagonal Elements of a Matrix =  %d",
principal+secondary );
         getch();
        return 0;
}
```

Output:

```
Please Enter Number of rows and columns  :  2 2
 Please Enter the Matrix Elements 1 4 5 6
 The Primary and secondary principal diagonal sum =  7 9
 The Sum of All Diagonal Elements of a Matrix =  16
```

40. Write a C program to accept 10 integers from the user and arrange them in ascendingorder and display them.

Program:

```
#include<stdio.h>
#include<conio.h>
 void main()
 {
 int i,j,temp,a[10];
 clrscr();
 printf(“Enter 10 integer numbers: \n”);
 for(i=0;i<10;i++)
 scanf(“%d”,&a[i]);
 for (i=0;i<10;i++)
 {
  for(j=i+1;j<10;j++)
  {
  if(a[i]>a[j])
   {
    temp=a[j];
    a[j]=a[i];
    a[i]=temp;
    }
   }
 }
 printf(“\n\nThe 10 numbers sorted in ascending order are: \n”);
 for(i=0;i<10;i++)
 printf(“%d\t”,a[i]);
 getch();
 }
```

OUTPUT:

```
Enter 10 integer numbers:
2 9 7 4 3 6 8 1 5 10
The numbers 10 sorted in ascending order are:
1 2 3 4 5 6 7 8 9 10
```

38. Write a C program to find GCD of two numbers using recursion.

Program:

```
#include <stdio.h>
#include <conio.h>
int hcf(int n1, int n2);
int main() {
   int n1, n2;
   clrscr();
   printf("Enter two positive integers: ");
   scanf("%d %d", &n1, &n2);
   printf("G.C.D of %d and %d is %d.", n1, n2, hcf(n1, n2));
   return 0;
}
```

```
int hcf(int n1, int n2) {
   if (n2 != 0)
      return hcf(n2, n1 % n2);
   else
      return n1;
}
```

Run Code

Output

```
Enter two positive integers: 366
60
G.C.D of 366 and 60 is 6.
```