Q1: An array is a bitonic array if all integers from index 0 to index i are sorted in ascending order, and all subsequent integers from index i+1 to n-1 are sorted in descending order. Given a bitonic array of n distinct integers, write a C/C++ program to find the maximum integer in the array in O(log(n)) time.

Example:

Input: n = 6 , A[] ={1 2 4 8 7 6}

Output: 8

Code:

```c
#include<stdio.h>
void main(){
    int n;
    printf("enter no of elements you want to enter in array : ");
    scanf("%d",&n);
    int a[n];
for (int i = 0; i < n; i++)
{
    scanf("%d",&a[i]);

}
int maxi;
for (int i = 0; i < n; i++)
{
    if(a[i]<a[i+1]){
        maxi=i+1;
    }
    if(a[i]>a[i+1]){
        break;
    }

}
int mid;
printf("maximum index is %d\n",maxi);
int low=0,up=n;
while(low<up){
    mid=(low+up)/2;
    if(maxi==mid){break;}
    if(maxi>mid){low=mid-1;}
    if(maxi<mid){up=mid-1;}
}
printf("maximum element is %d",a[mid]);

}
```

Output:

```
PS C:\Users\Prath> cd "c:\Users\Prath\OneDrive\Desktop\ds lab\" ; if ($?) { gcc ds1.c -o ds1 } ; if ($?) { .\ds1 }
enter no of elements you want to enter in array : 6
1
2
4
8
7
6
maximum index is 3
maximum element is 8
PS C:\Users\Prath\OneDrive\Desktop\ds lab>
```

Q2: Let A[n] be an array of n distinct integers. If i < j and A[i] > A[j], then the pair (i, j) is called an inversion of A. Write a C/C++ program that determines the number of inversions in any permutation on n elements.

Example: A = {4, 1, 3, 2} output is 4

CODE

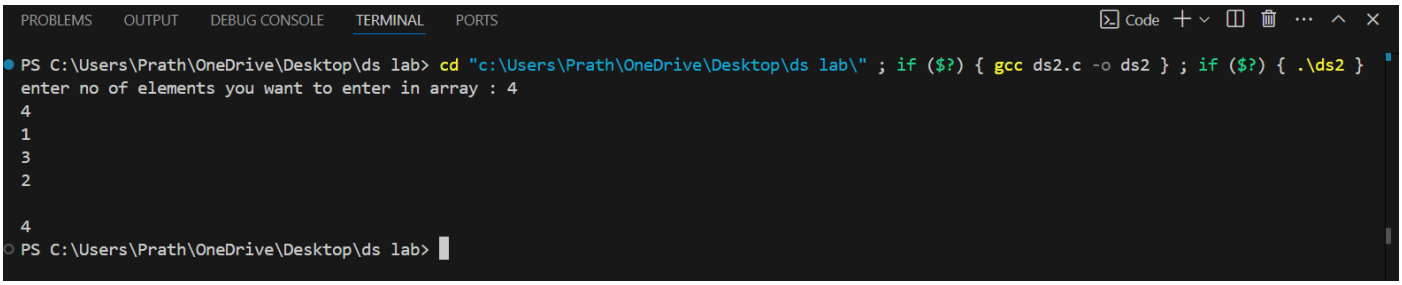
```c
#include<stdio.h>
int main(){
    int n,count=0;
    printf("enter no of elements you want to enter in array : ");
    scanf("%d",&n);
int a[n];
for (int i = 0; i <n ; i++)
{
    scanf("%d",&a[i]);
}

for (int i = 0; i <n; i++)
{
    for (int j=i; j<n; j++)
    {
    if(a[i]>a[j])
    {
        count++;

    }
    }

}
printf("\n%d",count);
return 0;
}
```

OUTPUT



Q3: Write a C program to manage the details of students using an array of structures. The program should:

1. Accept the number of students (n) from the user.
2. For each student, input the following details:
    - Roll number (integer)
    - Name (string)
    - Marks (floating-point value)
3. Store the details of all students in an array of structures.
4. Display the details of all students in a formatted way.

CODE

```c
#include<stdio.h>
struct student{
    int rollno;
    char sname[40];
    float marks;
};

int main(){
    int n;
    scanf("%d",&n);
struct student ai[n];
for (int i = 0; i <n; i++)
{
printf("\nenter roll no ");
scanf("%d",&ai[i].rollno);
printf("\nenter name");
scanf("%s",ai[i].sname);
printf("\nenter marks ");
scanf("%f",&ai[i].marks);


}

printf("\n student info ");
for (int i = 0; i < n; i++)
{
    printf("\n roll no=%d ",ai[i].rollno);
    printf("\nname = %s ",ai[i].sname);
    printf("\nmarks = %f ",ai[i].marks);

}

return 0;

}
```

OUTPUT

```
enter roll no  : 1

enter name : pratham

enter marks :  100

enter roll no  : 2

enter name : siddharth

enter marks :  99

enter roll no  : 3

enter name : siddh

enter marks :  98

enter roll no  : 4

enter name : harikesh

enter marks :  97

 student info
 roll no=1
name = pratham
marks = 100.000000
 roll no=2
name = siddharth
marks = 99.000000
 roll no=3
name = siddh
marks = 98.000000
 roll no=4
name = harikesh
marks = 97.000000
PS C:\Users\Prath\OneDrive\Desktop\ds lab>
```