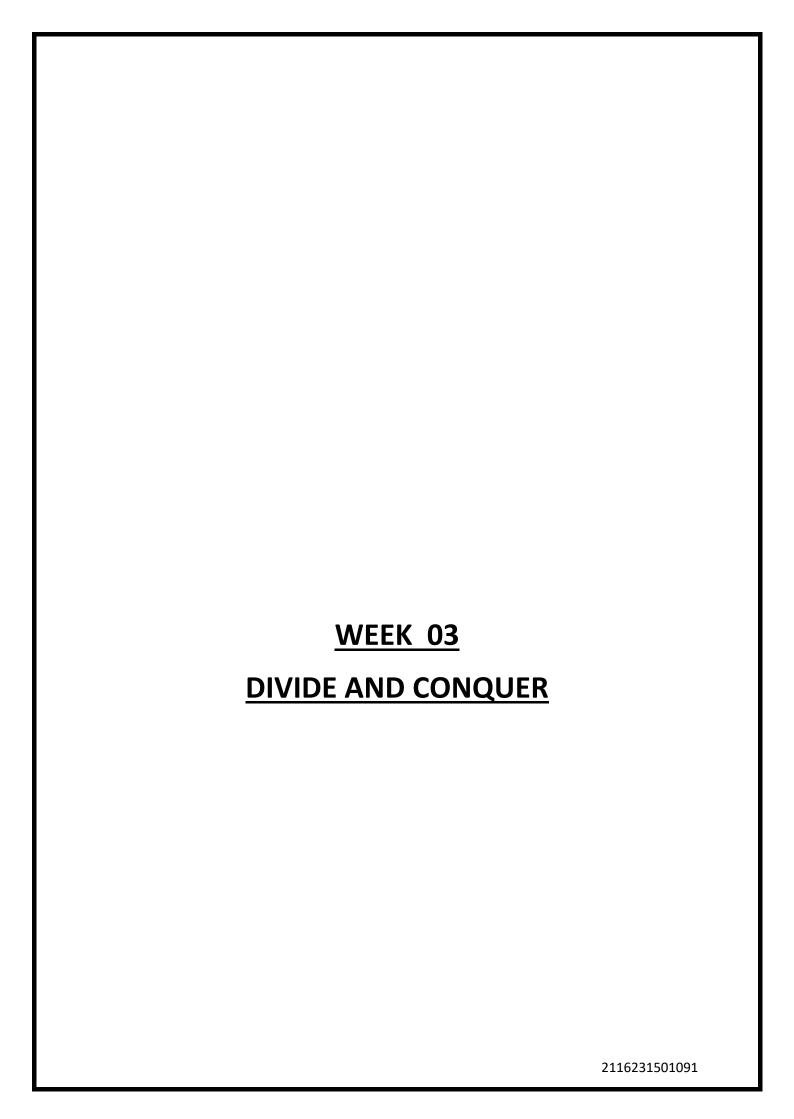
RAJALAKSHMI ENGINEERING COLLEGE RAJALAKSHMI NAGAR, THANDALAM – 602 105



CS23331 DESIGN AND ANALYSIS OF ALGORITHM LAB

Laboratory Observation Note Book

Name:
Year / Branch / Section :
Register No. : 231501091
Semester : 3 rd Semester
Academic Year: 2024-2025



1) Problem Statement

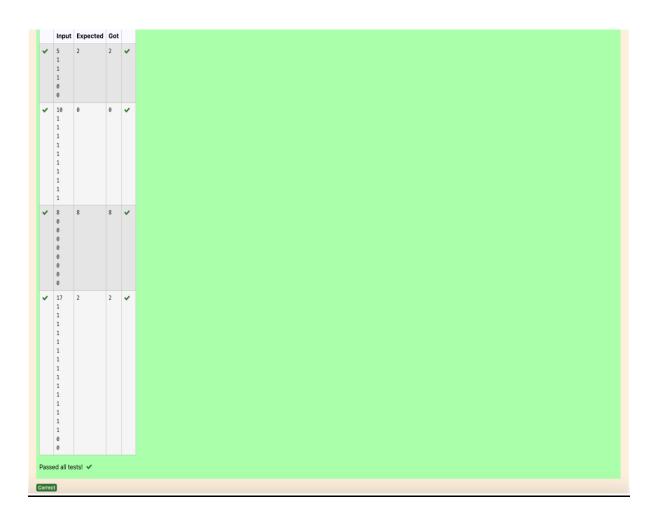
Given an array of 1s and 0s this has all 1s first followed by all 0s. Aim is to find the number of 0s. Write a program using Divide and Conquer to Count the number of zeroes in the given array. Input Format:

First Line Contains Integer m – Size of array
Next m lines Contains m numbers – Elements of an array
Output Format:

First Line Contains Integer – Number of zeroes present in the given array.

```
#include<stdio.h>
int conquer(int a[],int start,int end){
   int mid=(start+end)/2;
   if(start==end && a[start]==0){
     return 1;
   }
   if(start==end && a[start]!=0){
     return 0;
   }
   return(conquer(a,start,mid)+conquer(a,mid+1,end));
}
int main(){
   int n,i;
```

```
scanf("%d",&n);
int a[n];
for(i=0;i<n;i++){
    scanf("%d",&a[i]);
}
int start=0,end=n-1;
printf("%d",conquer(a,start,end));
}</pre>
```



2) Given an array nums of size n, return the majority element.

The majority element is the element that appears more than [n / 2] times. You may assume that the majority element always exists in the array.

Example 1:

Input: nums = [3,2,3]

Output: 3

Example 2:

Input: nums = [2,2,1,1,1,2,2]

Output: 2

Constraints:

- n == nums.length
- 1 <= n <= 5 * 10⁴
- -2³¹ <= nums[i] <= 2³¹ 1

For example:

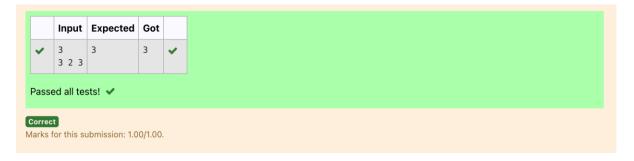
Input	Result
3 3 2 3	3
7 2211122	2

CODE:

#include<stdio.h>

int main(){

```
int n;
scanf("%d",&n);
int a[n];
for(int i=0;i<n;i++){
  scanf("%d",&a[i]);
}
for(int i=0;i<n;i++){
  int count=0;
  for(int j=0;j<n;j++){
    if(a[i]==a[j]){
       count++;
    }
  }
  if(count>n/2){
     printf("%d",a[i]);
     break;
  }
```



3) Problem Statement:

Given a sorted array and a value x, the floor of x is the largest element in array smaller than or equal to x. Write divide and conquer algorithm to find floor of x.

Input Format:

First Line Contains Integer n – Size of array

Next n lines Contains n numbers – Elements of an array

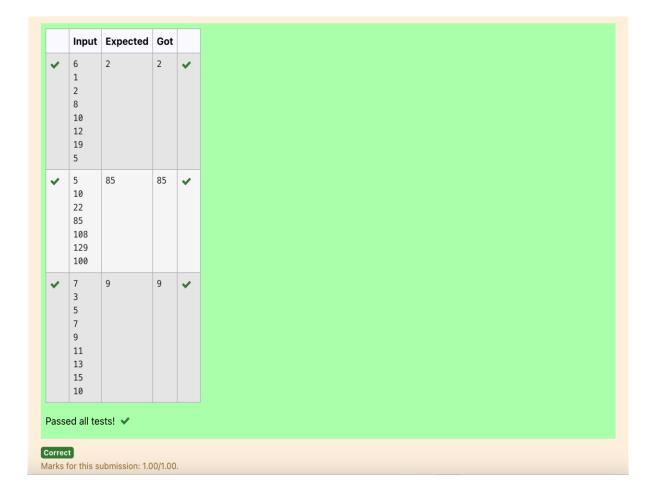
Last Line Contains Integer x – Value for x

Output Format:

First Line Contains Integer – Floor value for x

```
#include<stdio.h>
int main(){
   int n,x,flr,i;
   scanf("%d",&n);
   int a[n];
   for(i=0;i<n;i++)
       scanf("%d",&a[i]);
   scanf("%d",&x);
   int mid=n/2;
   if(x<a[mid])
   {
     flr=a[0];
     for(i=0;i<mid;i++)</pre>
```

```
{
       if(a[i]>=flr)
          if(a[i]<x)
            flr=a[i];
     }
  }
  else
  {
    flr=a[mid];
    for(i=mid;i<n;i++)
       if(a[i]>=flr)
          if(a[i]<x)
            flr=a[i];
     }
  }
  printf("%d",flr);
}
```



4) Problem Statement:

Given a sorted array of integers say arr[] and a number x. Write a recursive program using divide and conquer strategy to check if there exist two elements in the array whose sum = x. If there exist such two elements then return the numbers, otherwise print as "No".

Note: Write a Divide and Conquer Solution Input Format:

First Line Contains Integer n – Size of array

Next n lines Contains n numbers – Elements of an array

Last Line Contains Integer x – Sum Value

Output Format:

First Line Contains Integer – Element1
Second Line Contains Integer – Element2 (Element 1 and Elements
2 together sums to value "x").

```
for(j=i+1;j< n;j++){}
     if((a[i]+a[j])==x){
       q=a[i]+a[j];
       m=a[i];
       p=a[j];
  }
if(q==x) {
  printf("%d\n",m);
  printf("%d",p);
}
else
  printf("No");
```



5) Write a Program to Implement the Quick Sort Algorithm

Input Format:

The first line contains the no of elements in the list-n The next n lines contain the elements.

Output:

Sorted list of elements

For example:

Input	Result
5	12 34 67 78 98
67 34 12 98 78	

```
#include<stdio.h>
int main()
{
   int n,i,j,temp;
   scanf("%d",&n);
   int a[n];
   for(i=0;i<n;i++)
   {
      scanf("%d",&a[i]);
   }
}</pre>
```

```
int x;
  scanf("%d",&x);
  for(i=0;i<n;i++)
    for(j=i+1;j<n;j++)
    {
       if(a[i]>a[j])
       {
         temp=a[i];
         a[i]=a[j];
         a[j]=temp;
       }
    }
  }
  for(i=0;i<n;i++)
     printf("%d ",a[i]);
}
```

5 67 34 12 98 78 12 34 67 78 98 12 34 67 78 98 10 1 56 78 90 32 56 11 10 90 114 1 10 11 32 56 56 78 90 90 114 1 10 11 32 56 56 78 90 90 114 12 34 67 78 98 12 34 67 78 98 10 11 32 56 56 78 90 90 114 1 10 11 32 56 56 78 90 90 114 1 10 11 32 56 56 78 90 90 114 1 10 11 32 56 56 78 90 90 114 1 10 11 32 56 56 78 90 90 114 10 11 90 12 3 4 5 6 7 8 9 10 11 90		Input	Expected	Got	
1 56 78 90 32 56 11 10 90 114 12 9 8 7 6 5 4 3 2 1 10 11 90 1 2 3 4 5 6 7 8 9 10 11 90 1 2 3 4 5 6 7 8 9 10 11 90	~		12 34 67 78 98	12 34 67 78 98	~
9 8 7 6 5 4 3 2 1 10 11 90	*		1 10 11 32 56 56 78 90 90 114	1 10 11 32 56 56 78 90 90 114	~
Passed all tests! ✓	~		1 2 3 4 5 6 7 8 9 10 11 90	1 2 3 4 5 6 7 8 9 10 11 90	~