Practical 3

AIM - WAP to implement a Binary Search.

* Algorithm

***** Time Complexity

$$T(n) = O(nlogn)$$

Space complexity

$$S(n) = O(1)$$

♦ Source code

```
#include <stdio.h>
int binarySearch(int Array[], int i , int j, int x){
  int mid = (i+j)/2;
  if(Array[mid]==x){
    return mid;
}
```

```
}
  else{
     if(Array[mid]>x){
       binarySearch(Array,i,mid-1,x);
     }else{
       binarySearch(Array,mid+1,j,x);
  }
}
int main() {
  int n;
  printf("Enter the size of Array\n");
  scanf("%d",&n);
  int Arr[n];
  for(int i=0; i< n; i++){
     if(i==0){
       printf("All Array element should in sorted mannner\n");
     printf("Enter the %d element for array\n",i);
    scanf("%d",&Arr[i]);
  }
  printf("YOUR ARRAY \n");
  for(int i=0; i< n; i++){
     printf("%d\t",Arr[i]);
  }
  int x;
  printf("\nEnter element to search \n");
  scanf("%d",&x);
  int y = binarySearch(Arr,0,n,x);
  printf("Position of element is %d",y+1);
  return 0;
}
```

Output

```
YOUR ARRAY

1 23 45
Enter element to search

23
Position of element is 2
```