TW:- 6

Design, develop, code and run the program in any suitable language to

implement the binary search algorithm. Determine the basis paths and using them derive different test cases, execute these test cases and discuss the test results.

ALGORITHM

Step 1: Input value of ‘n’. Enter ‘n’ integer numbers in array int mid;

Step 2: Initialize low = 0, high = n -1

Step 3: until ( low <= high ) do

mid = (low + high) / 2

if ( a[mid] == key )

then do Step 5

else if ( a[mid] > key )

then do

high = mid - 1

else

low = mid + 1

Step 4: Print unsuccessful search do step 6.

Step 5: Print Successful search. Element found at position mid+1.

Step 6: Stop

#include<stdio.h>

#include<conio.h>

int main()

{

int a[20],n,low,high,mid,key,i,flag=0;

clrscr();

printf("Enter the value of n:\n");

scanf("%d",&n);

if(n>0)

{

printf("Enter %d elements in ASCENDING order\n",n);

for(i=0;i<n;i++)

{

scanf("%d",&a[i]);

}

printf("Enter the key element to be searched\n");

scanf("%d",&key);

low=0;

high=n-1;

while(low<=high)

{

mid=(low+high)/2;

if(a[mid]==key)

{

flag=1;

break;

}

else if(a[mid]<key)

{

low=mid+1;

}

else

{

high=mid-1;

}

}

if(flag==1)

printf("Successful search\n Element found at Location %d\n",mid+1);

else

printf("Key Element not found\n");

}

else

printf("Wrong input");

getch();

return 0;

Test case 1:

Enter the value of n:

8

Enter 8 elements in ASCENDING order

12 19 22 26 33 45 77 99

Enter the key element to be searched

33

Successful search

Element found at Location 5

Test case 2:

Enter the value of n:

6

Enter 6 elements in ASCENDING order

51 3 24 33 54 77

51 3 24 33 54 77

Enter the key element to be searched

2

Key Element not found