**Experiment No :** 06

**Experiment name :** write a C programming for binary search

**Methodology :**

Binary search is like a list of numbers, from the list of numbers we will search for a number and how many cells that number is in. Suppose there are cells like one two three and we put some numbers there, then we will use the method to find out how many cells that number is in. That is we will search the list of numbers from the right and left side, this way we will find the number easily and less time.

**Flow-Chart :**

Fori=1;i<=n;i++

Read all elements

Search = bianry(a,n,x)

If (search == 0)

Print unsuccesfull search

Print succesfull search

Read value of n

**Code :**

#include <stdio.h>

int main()

{

int c, first, last, middle, n, search, array[100];

printf("Enter number of elements\n");

scanf("%d", &n);

printf("Enter %d integers\n", n);

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

scanf("%d", &array[c]);

printf("Enter value to find\n");

scanf("%d", &search);

first = 0;

last = n - 1;

middle = (first+last)/2;

while (first <= last) {

if (array[middle] < search)

first = middle + 1;

else if (array[middle] == search) {

printf("%d found at location %d.\n", search, middle+1);

break;

}

else

last = middle - 1;

middle = (first + last)/2;

}

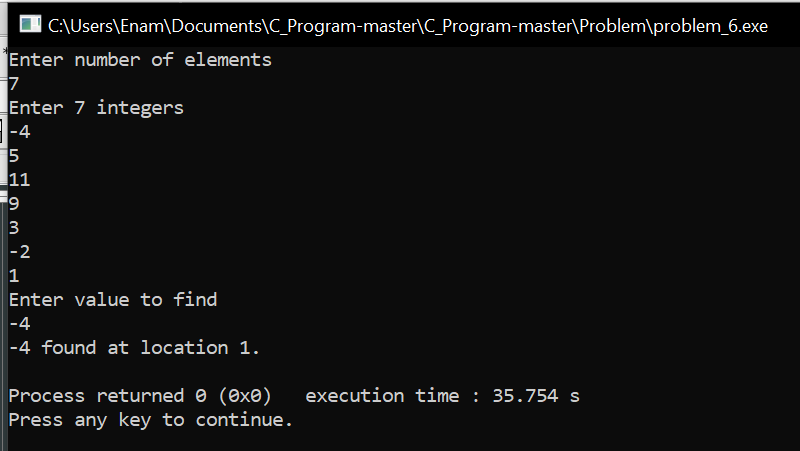
if (first > last)

printf("Not found! %d isn't present in the list.\n", search);

return 0;

}

**Output:**



**Result discussion :**

We mentioned in the methodology that binary search divides a list of numbers or a list of names into two from the middle. The method took less time to find our number and we were able to print the output