

EXPERIMENT NO.9

IMPLEMENTATION OF BINARY SEARCH

Aim: Write A Program to Implement Binary Search

Theory:

Binary Search, also known as **Half-Interval Search** or **Logarithmic search**, is a search algorithm that finds the position of a target value within a **Sorted Array**. It compares the target value to the **Middle Element** of the array; if they are unequal, the half in which the target cannot lie is **eliminated** and the search continues on the remaining half until it is successful.

Binary Search is a fast search algorithm with run-time complexity of **$O(\log n)$** . This search algorithm works on the principle of **Divide & Conquer**. For this algorithm to work properly the data collection should be in **sorted form**.

Algorithm:

1. Start
2. Get the Middle Element of the Array
3. if the Middle Element equals to the Searched Value, the algorithm stops
4. Otherwise, two cases are possible:
 - ✓ Searched Value is Less, than the Middle Element. In this case, go to the step 1 for the part of the Array, Before Middle Element.
 - ✓ Searched Value is Greater, than the Middle Element. In this case, go to the step 1 for the part of the Array, After Middle Element.

Program:

```
#include <stdio.h>

int main() {
    int a[100], n, key, low, mid, high, i, chk = 0;

    printf("How Many Elements:\n");
    scanf("%d", &n);

    printf("Enter the elements in sorted order:\n");
    for (i = 0; i < n; i++) {
        printf("Enter Element %d:\n", (i + 1));
        scanf("%d", &a[i]);
    }

    printf("Enter The Element To Be Searched:\n");
    scanf("%d", &key);

    low = 0;
    high = n - 1;

    while (low <= high) {
        mid = (low + high) / 2;
        if (a[mid] == key) {
            printf("Value Found At Position %d\n", (mid + 1));
            chk = 1;
            break;
        }

        if (a[mid] > key)
            high = mid - 1;
        else
            low = mid + 1;
    }

    if (chk == 0)
        printf("Value Not Found\n");

    return 0;
}
```

Output:

```
How Many Elements:
5
Enter the elements in sorted order:
Enter Element 1:
10
Enter Element 2:
20
Enter Element 3:
30
Enter Element 4:
40
Enter Element 5:
50
Enter The Element To Be Searched:
30
Value Found At Position 3
```

```
How Many Elements:
4
Enter the elements in sorted order:
Enter Element 1:
10
Enter Element 2:
20
Enter Element 3:
30
Enter Element 4:
40
Enter The Element To Be Searched:
55
Value Not Found
```

Conclusion:

Hence, we have successfully implemented Binary Search in C language.