**NAME - SUVIDHI V. PAREEK**

**SE(COMPS) / DIV - 3 / ROLL NO. 10**

**Experiment No. 10: Binary Search Method**

**Aim :** Implementation of Binary Search Method

**Objective:** 1) Understand how to implement Binary Search algorithm.

**Theory:**

The improvement to searching method to reduce the amount of work can be done using binary searching. Binary searching is more efficient than linear searching if an array to be searched is in sorted manner.

Here an key item to be searched is compared with the item at middle of array. If they are equal search is completed. If the middle element is greater than key item searching proceeds with left sub array. Similarly, if middle element is less than key item than searching proceeds with right sub array and so on till the element is found.

For large arrays, this method is superior to sequential searching.

**Algorithm**

Algorithm : FIND(arr, x, first, last)

if (first > last)then

return -1

End if

mid = (first + last) / 2

if (arr[mid] = x)

return mid

End if

if (arr[mid] < x)

return find(arr, x, mid+1, last)

End if

return find(arr, x, first, mid-1)

**Code:**

#include <stdio.h>

#include <stdlib.h>

int binarySearch(int arr[], int size, int element) {

int low, mid, high;

low = 0;

high = size - 1;

while (low <= high) {

mid = (low + high) / 2;

if (arr[mid] == element) {

return mid;

}

if (arr[mid] < element) {

low = mid + 1;

} else {

high = mid - 1;

}

}

return -1;

}

int main() {

int arr[] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};

int size = sizeof(arr) / sizeof(int);

int element = 10;

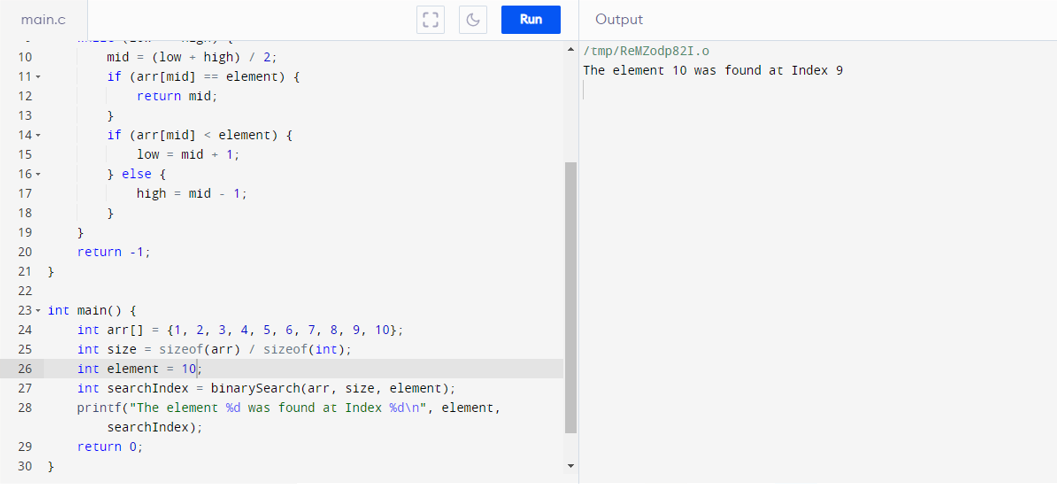
int searchIndex = binarySearch(arr, size, element);

printf("The element %d was found at Index %d\n", element, searchIndex);

return 0;

}

**Output:**



**Conclusion:**

Binary Search is defined as a searching algorithm used in a sorted array by repeatedly dividing the search interval in half. The idea of binary search is to use the information that the array is sorted and reduce the time complexity to O(log N).