Aim

To write a C program to search for a number using binary search.

Algorithm

1. Start program.
2. Input sorted array and target.
3. Initialize `low=0`, `high=n-1`.
4. While `low <= high`:
    - Find `mid = (low+high)/2`.
    - If `arr[mid]==key`, print found.
    - If `arr[mid]<key`, set `low = mid+1`.
    - Else `high = mid-1`.
5. If not found, print not found.

Code

```
#include <stdio.h>

int main() {
    int n, i, key, low, high, mid;
    printf("Enter size of sorted array: ");
    scanf("%d", &n);
    int arr[n];
    printf("Enter sorted elements:\n");
    for (i = 0; i < n; i++)
        scanf("%d", &arr[i]);
```

```
    printf("Enter element to search: ");
    scanf("%d", &key);

    low = 0;
    high = n-1;
    while (low <= high) {
        mid = (low + high) / 2;
        if (arr[mid] == key) {
            printf("Element found at position %d\n", mid+1);
            return 0;
        } else if (arr[mid] < key)
            low = mid + 1;
        else
            high = mid - 1;
    }
    printf("Element not found\n");
    return 0;
}
```

Sample Output

```
Enter size of array: 4
Enter elements:
1 2 3 4
Enter element to search: 2
Element found at position 2


=== Code Execution Successful ===
```

Result

Binary search successfully finds elements in sorted arrays.