EXPERIMENT NO. 2

AIM: Write a program to implement Binary Search.

SCOPE: Searching is an operation or a technique that helps finds the place of a given element or value in the list. Any search is said to be successful or unsuccessful depending upon whether the element that is being searched is found or not.

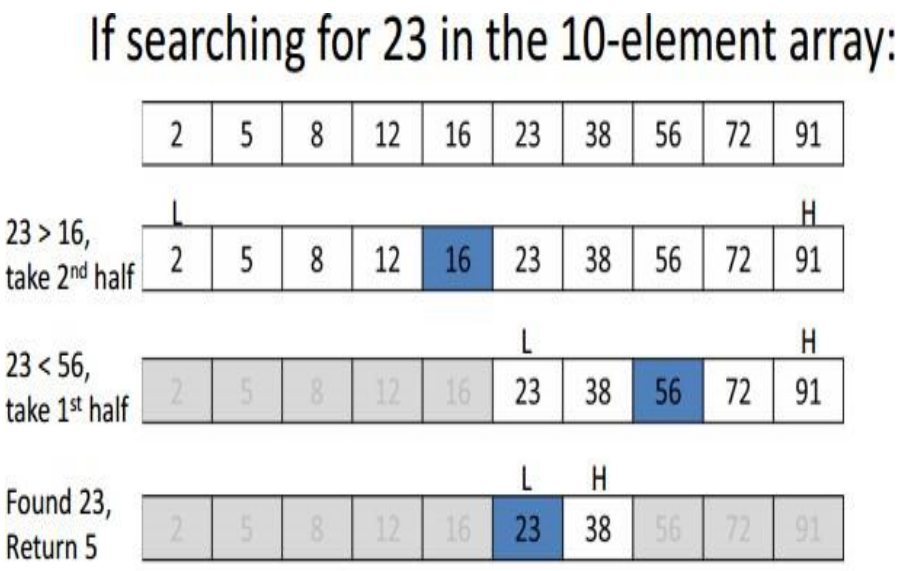
FACILITIES: Software Needed: Turbo C

THEORY:

Binary Search

Binary search works on sorted arrays. Binary search begins by comparing the middle element of the array with the target value. If the target value matches the middle element, its position in the array is returned. If the target value is less than or greater than the middle element, the search continues in the lower or upper half of the array, respectively, eliminating the other half from consideration.

Example:



IMPLEMENTATION:

```
//Program to implement Binary Search

#include <stdio.h>
int main ()
{
  int c, beg, end, middle, n, item, 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 item to find\n");
  scanf("%d", &item);
  beg = 0;
  end = n - 1;
  middle = (beg+end)/2;

  while (beg <= end) {
    if (array[middle] < item)
      beg = middle + 1;
    else if (array[middle] == item) {
      printf("%d found at location %d.\n", item, middle+1);
      break;
    }
    else
      end = middle - 1;

    middle = (beg + end)/2;
  }
  if (beg > end)
    printf("Not found! %d isn't present in the list.\n", item);

  return 0;
}
```

Output:

Enter number of elements
6
Enter 6 integers
22 33 55 66 77 88
Enter item to find
66
66 found at location 4.

RESULT: In this way we have Implemented Binary search with Turbo C and tested with examples.