06. Write a Program to find an element using the binary search algorithm.

Theory: Binary search is 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)).

The binary search algorithm applied to our array DATA works as follows. During each stage of our algorithm, our search for ITEM is reduced to a segment of elements of Data:

DATA[BEG], DATA[BEG+1], DATA[BEG+2], ...

...., DATA[END].

The algorithm compares ITEM with ~~th~~ the middle element DATA[MID] of the segment.

MID = INT((BEG+END)/2)

If DATA[MID] = ITEM, then the search is successful and we set LOC = MID.
Otherwise a new segment of DATA is obtain as follows:

(a) If ITEM < DATA[MID], then ITEM can appear only in the left half of the segment:

DATA[BEG], DATA[BEG+1], ..., DATA[MID-1]

So we reset END = MID - 1

(b) If ITEM > DATA[MID] then ITEM can appear only in the right half of the segment:

DATA[MID+1], DATA[MID+2], ... DATA[END]

So we reset BEG = MID + 1

Here generally the ~~BGt~~ BEG = LB (Lower Bound), End = ~~&~~ UB (Upper Bound).

Example:

DATA: 1, 3, 5, 6, 7, 8, 10, 12

Suppose ITEM = 5. The search for ITEM in the array data.

1. ~~Ex~~ Here $BEG = 1$ and $END = 8$ Hence,

$MID = INT((1+8)/2) = 4$ So $DATA[MID] = 6$

2. Since $5 < 6$, END has its value changed by $END = MID - 1 = 4 - 1 = 3$. Hence.

$MID = INT((1+3)/2) = 2$ and $DATA[MID] = 3$

3. Since $5 > 3$, BEG has changed its value.

$BEG = MID + 1 = 2 + 1 = 3$

$MID = INT((3+3)/2) = 3$, So

$DATA[MID] = 5$

We have found ITEM in location

$LOC = MID = 5$

Algorithm:

(Binary search) BINARY(DATA,LB,UB,ITEM,LOC)

Here DATA is sorted array which lower bound LB, upper bound LB, searching item is item. Find location is LOC.

```
1. set BEG = LB, END = UB and MID = INT($(\frac{BEG+END}{2})$)

2. Repeat step 3 and 4 while BEG ≤ END
   and DATA[MID] ≠ ITEM.

3. If ITEM < DATA[MID], then
      set END = MID - 1
   else BEG = MID + 1

4. set MID = INt((BEG + END)/2)

5. If DATA[MID] = ITEM, then,
      set LOC = MID
   else
      set LOC = NULL

6. Exit.
```

Source code: C++

```
# include<iostream>
using namespace std;
int main()
{
    int DATA[10]={1,2,3,4,5,6,7,8,9,10};
    int N=10, LB=0, UB=N-1, LOC;
    int BEG=LB, END=UB, MID=int((BEG+EN
                                   2
    int ITEM=5;
    while(BEG<=END && DATA[MID]!=ITEM)
    {
        if (ITEM< DATA[MID])
            END=MID-1;
        else
            BEG=MID+1;

        MID=int((BEG+END)/2);
    }
   if (DATA[MID]=ITEM)
      LOC=MID;
   else
      LOC="NULL";
 cout<<"The searching element is:"<<
                              DATA[LOC]<<endl;
cout<<" The searching element location is:"
                      << loc << endl;
```

Input:

Data elemets are.

{ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10}

searching elemet / ITEM = 5

Output:

~~The searching~~

The searching element is : 5

The searching elements location is: 4