

C++ Binary Search

Lecture-24

Raghav Garg



What and

```
Search Space
Why? arr = {1,2,4,5,9,15,18,213
                                        7 obs
          target = 18
          bool flog = false; //target not present
         for (int i=0; izn; i++)
                                     Mean
                                      A ROTCIA
             if (arr[i] = = target)E
```

flog = true; braak,

T. C = 0(n)

if (flog = = true) cout < 2 "Present"; else cout << "Not Present";

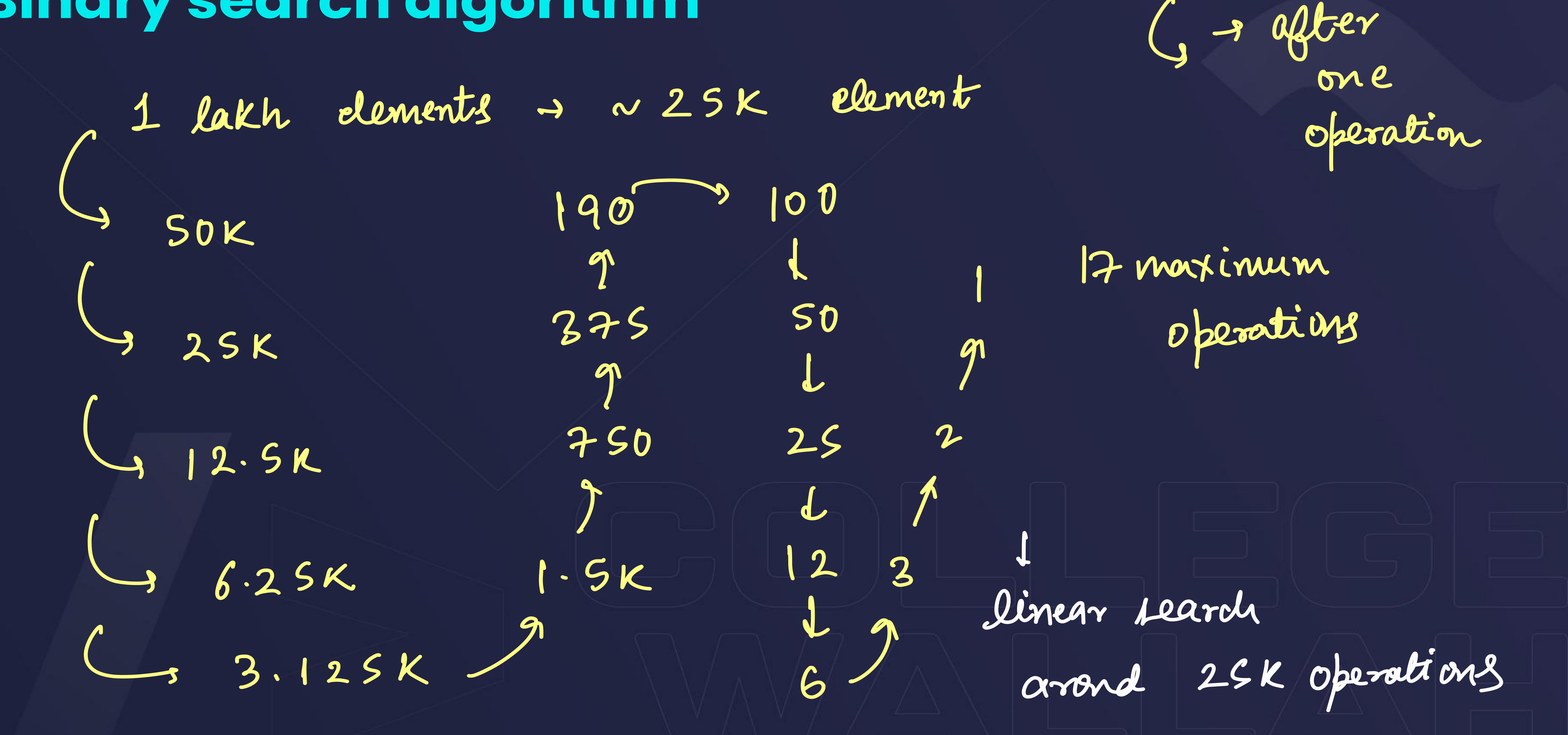
Binary search algorithm - Works only if array is

Sorted (ascending or descending)

```
0 1 2 3 4 5 6 7
  arr = [1,2,4,5,9,15,18,21,243
target = 15
              int mid = Lothi;
int 10 = 0;
              if(arr[mid] == target) return true;
int hi = n-1;
            if (arr [mid] < target) lo = mid+1;
           if (arr[mid] > target) ni = mid-1;
```



Binary search algorithm



Ques: Binary Search

Leetcode 704

int
$$lo = 0$$
;
int $hi = n-1$;
while $(lo <= hi)$?
int $mid = lo + (hi-lo)/2$;

$$\frac{lo + hi}{2} = \frac{lo + lo + hi - lo}{2} = \frac{lo + lo}{2} + \frac{hi - lo}{2}$$

$$= \frac{lo + (ni - lo)}{2}$$



Time complexity analysis

$$n \rightarrow \frac{n}{2^1} \rightarrow \frac{n}{2^2} \rightarrow \frac{n}{2^3} \rightarrow \cdots \rightarrow \frac{n}{2^x}$$

T. C.
$$\rightarrow O(x) = O(\log n)$$



*Lower bound

Ques: Given a sorted integer array and an integer 'x', find the lower bound of x.

```
arr = [1,2,4,5,9,15,18,21,243
MAC
      while (lo <=ni) {
       mid = lo + (hi-lo)/2;
       if (arr[mid] == x) cout << arr[mid-1];
else if (arr[mid] < x) lo = mid+1
       else if (arr [mid] > n) ni = mid-1 cout << arrshi];
```



lower_bound(arr.begin(), arr.end(), target)

rector



*Upper bound Ques: Given a sorted integer array and an integer 'x', find the upper bound of x.

```
Copy Paste

Bas yahan pe aafko

Cout < 2 aro [lo];
```







Ques: Given a sorted array of n elements and a target 'x'. Find the first occurrence of 'x' in the array. If 'x' does not exist return -1.

```
0 1 2 3 4 5 6 7 8 9 10 11 12
arr = { 1, 2, 2, 3, 3, 3, 3, 3, 4, 4, 5, 8, 93;
      while (lo <= ni) \( \) mid
         int mid = lo+(hi-lo)/2;
                              arr smid-1] == 3 - shi=mid-1 ~
d -> yes this is the first occurence
        if (arr[mid] = = x)
        if (arr [mid] > x) hi = mid-1;
      if(arr[mid] < x) lo = mid+1;
```



0(20gm)

Ques: Given a sorted array of non-negative distinct integers, find the smallest missing non-negative element in it.



Ques: Given a sorted array of non-negative distinct integers, find the smallest missing non-negative element in it.

```
2<sup>nd</sup> approach: Binary Search O(logn)
                 0 1 2 3 4 5 6 7
      arr = {0,1,2,3,4,8,9,12}
                              10 mid
                     nuid = lo + (ni-lo)/2;
                    if (arr [i] == i) lo = mid + 1;
                   if (arr(i))=i) cont <ii, break of
```

B SKILLS

```
0 1 2 3 4 5 6 7
  arr = {0,1,3,4,6,7,8,9,113
            if (arr[mid] ==mid) lo = mid+1;
             elle 5
ans = X
                ans = mud;
               ni = mid - 1;
```

Ques: Sqrt(x)

x = 36int lo = %; % ; % 6

int Mi = 36; % 7

Leetcode 69

while (lo <= hi) {

int mid =

if (mid * mid == x) return mid;

if (mid * mid > x) hi = mid - 1;

if (mid * mid < n) lo = mid + 1;

Leetcode 69

Ques : Sqrt(x)

$$x = 20 \rightarrow ans = 4$$



THANKYOU