1-Feb

01 February 2022 21:10

Searching :-

↳ Linear Search :-

[_ _ _ _ _ _ _]

0, 1, 2, 3 . . .

[15, 20, 5, 7, 12, 13, 19]

0 1 2 3 4 5 6

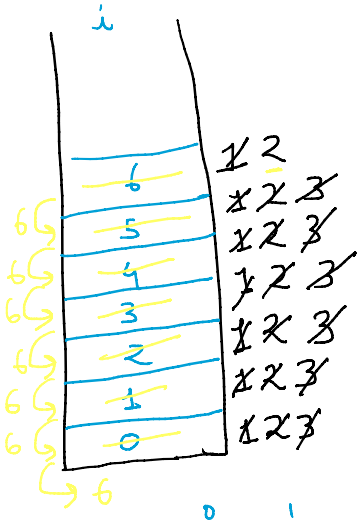
T.C. => O(n)

k = 18

```
Search(ars, K){
for(0 to ars.length-1){
    if(ars[i]==K){
        return i;
    }
}
return -1;
}
```

arr=[15, 20, 5, 7, 12, 13, 19] k=19, i=0

0 1 2 3 4 5 6

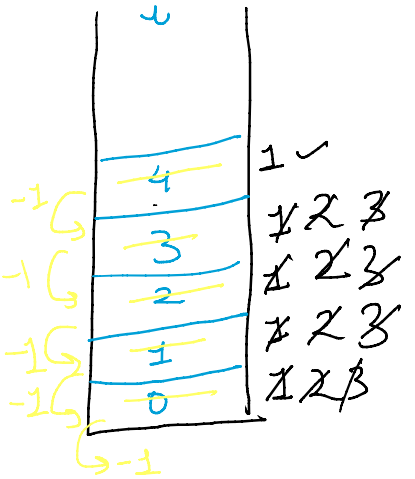



```
int linearSearchRec(arr,k,i){
1. if(i==ars.length) return -1;
2.     if(ars[i]==k){
           return i;
       }
3. return linearSearchRecursion(ars,k,i+1);
}
```

[2, 4, 1, 3] k=5

0 1 2 3

length = 4



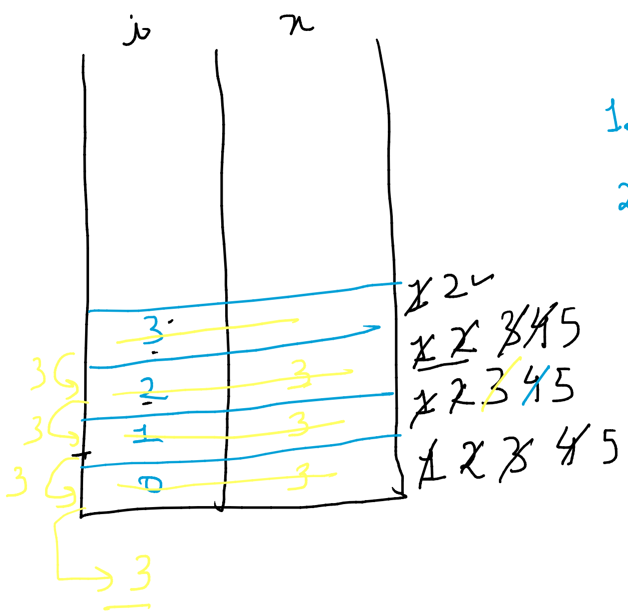

```
int linearSearchRec(arr,k,i){
1. if(i==ars.length) return -1;
2.     if(ars[i]==k){
           return i;
       }                    pre-order
3. return linearSearchRecursion(ars,k,i+1);
}                           Post-order
```

log(0 — n)

↳ -1

fos (0 — n)

[7, 8, 1, 2, 3]   K = 2, K = 5



```
int linearSearchRec(arr, k, i){
1.  if(i == arr.length) return -1;
2.      if (arr[i] == k){
            return i;
        }
3.  int n = linearSearchRecursion(arr, k, i+1);   // 2+1=3
4.  Sout ("I am recursion " + i);
5.  return n;
}
```

I'm recursion 2

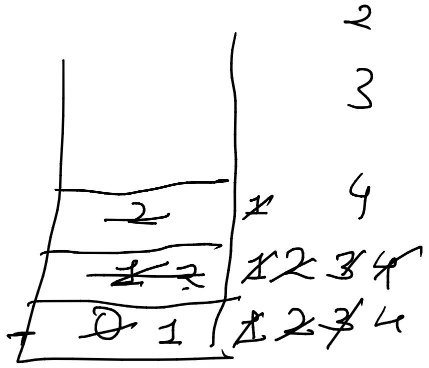
I'm recursion 1

I'm recursion 0

i++;

++i;

```
1  if (i == 2) return;
2      Sout ("pre" + i)
3      linearSRec( ++i )
4      Sout ("post" + i)
```



Pre 0

Pre 1

Post 2

Post 1

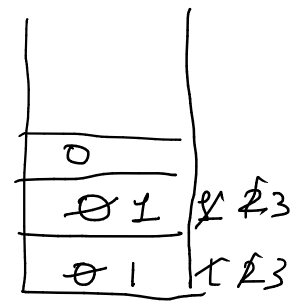
0

1

1

0

fos i++;



Pre 0

Pre 0

i=4

Sout (i++)