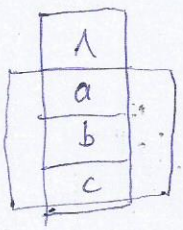
Palindrome

abcba
abcba
cabbc
cabac

if len(s) == 0
if len(s) == 1

Λ — Empty string.



x is a palindrome

aXa
bXb
} is also a palindrome

```
Boolean Ispalindrome (String S) {
    if (S.length() == 0 || S.length() == 1)
        return TRUE;
    if (S[0] != S[S.length-1])
        return FALSE;
    return Ispalindrome (S[1: S.length()-2])
```

} a [ ] k

} a [ ] a.

Draw the recursion tree for,

① abba ② abcba ③ abccdccaa ④ abca.

① IP(abba)
|
IP(bb)) TRUE
|
IP(Λ)) TRUE

② IP(abcba)
|
IP(bcb)) TRUE
|
IP(c)) TRUE

③ IP(abccdccaa)

↓

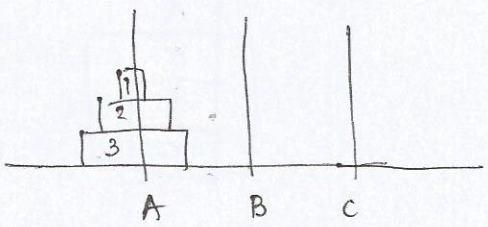
IP(bccdcca)) FALSE

④ IP(abca)

|

IP(bc)) FALSE

Towers of Hanoi



n disks from A to B :-

- n-1 disks from A to C ⟶ n-2 from A to B
- nth disk from A to B ⟶ (n-1)th from A to C
- n-1 disk from C to B ⟶ (n-2) from B to C.

Process terminates when n = 1 (when 1 disk only on tower)

Pseudo Code :-

```
Move Disks ( n , S , D , O ){
    if (n == 1)
        move 1 disk from S to D
    else {
        Move Disks (n-1, S, O, D)
        Move nth disk from S ⟶ D
        Move Disks (n-1, O, D, S)
    }
}
```

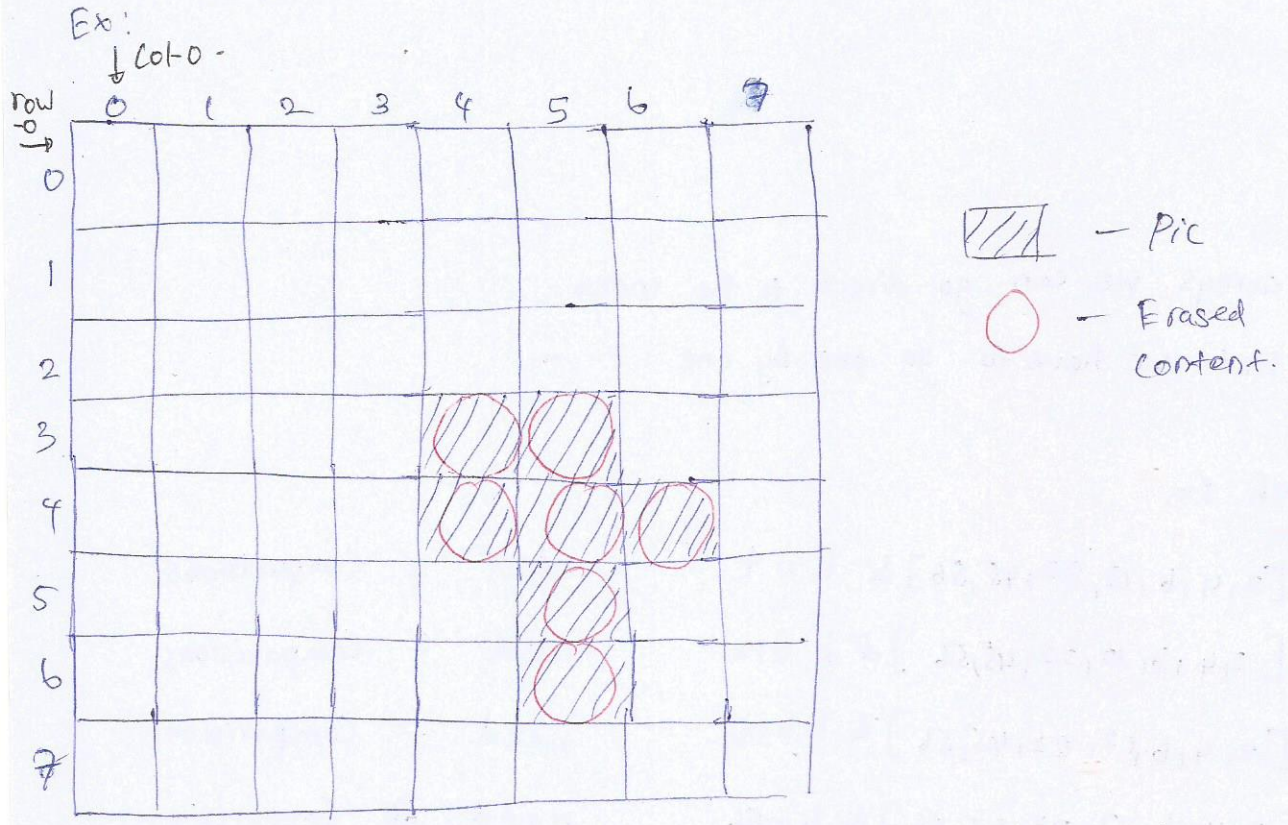
(n: no of disks; S: Source tower; D: Dest tower; O: other tower)

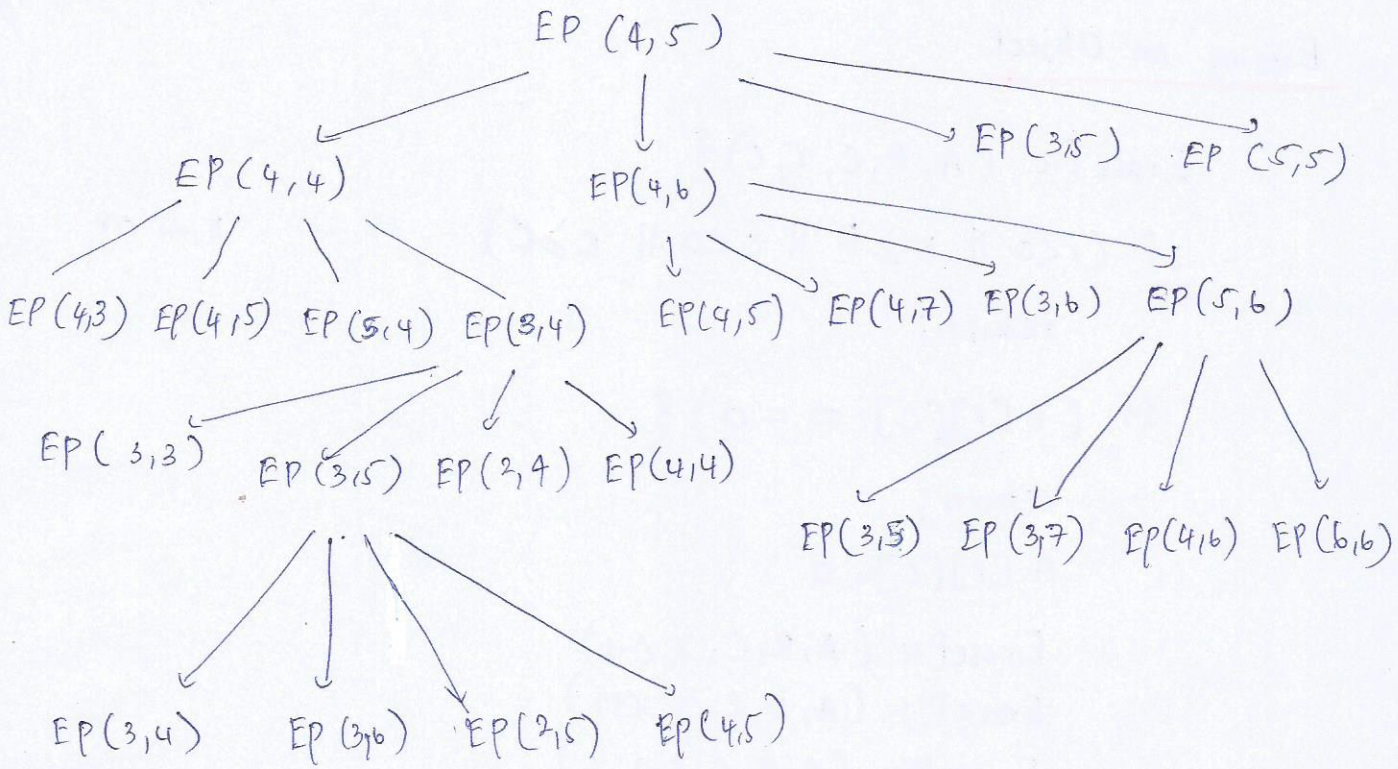
Here other tower becomes Source

Here Source tower becomes other.

Erasing an Object

```
ErasePic (A, R, C, r, c){
if (r<0 || r≥R || c<0 || c≥C)        || – or
    return;
if (A[r][c] == 0)
    return;
    A[r][c] = 0
    ErasePic (A, R, C, r, c-1)
    ErasePic (A, R, C, r, c+1)
    ErasePic (A, R, C, r-1, c)
    ErasePic (A, R, C, r+1, c)
}
```





ARRAYS

In arrays we can go direct to the index.
In linked list we have to go one by one.

Linear Search :-

(i) A = [2,4,6,12,22,45,56] & K = 4 need 2 Comparisions.

(ii) A = [2,4,6,12,22,45,56] & k = 12 need 4 Comparisions.

(iii) A = [2,4,6,12,22,45,56] & k = 56 need 7 Comparisions

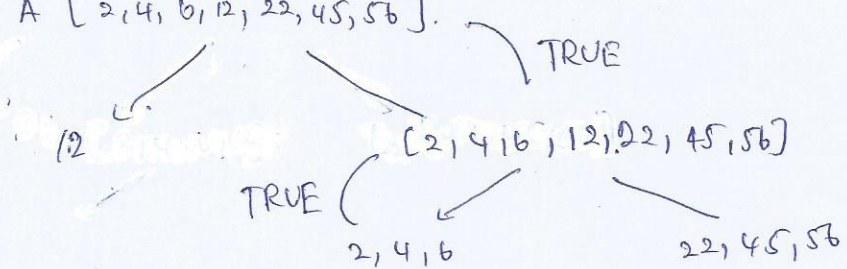
(iv) A = [2,4,6,12,22,45,56] & k = 66 need 7 Comparisions.

Best case = Something at the beginnig of the Array

Worst case = Something not in the array.

Recursion trees for above 2 scenarios.

Binary search (i)



Camparisions.

		Binary	Linear
(i)	k = 4 ⟶	2	2
(ii)	k = 12 ⟶	1	4
(iii)	k = 56 ⟶	$3 - \log_2 n$	$7 - n$
(iv)	k = 66 ⟶	$3 - \log_2 n$	$7 - n$

n – elements.