Recursion Concepts & Qns

Video 8

"मैं, DSA की शपथ लेता हूँ कि मैं जो पढ़ाऊंगा बहोत अच्छे से पढ़ाऊंगा।"

Facebook
Instagram → codestorywithMIK

(Twitter) → CSwithMIK

codestorywithMIK →

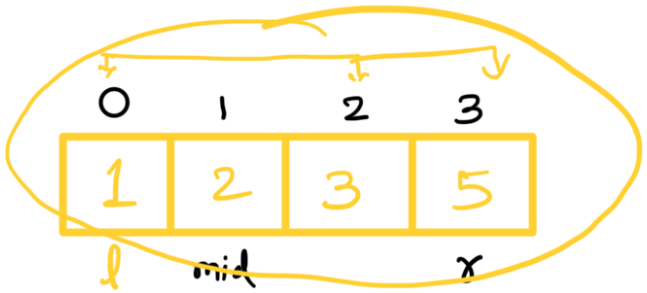
Motivation
(भाषण)

"Coding is not just about building things, but about continuously learning and improving"

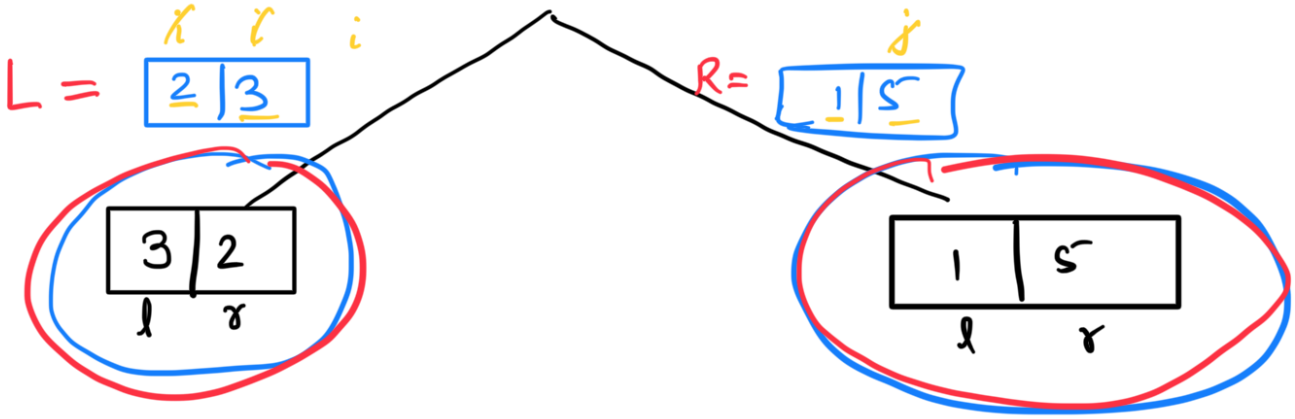
=> Keep learning & upskilling.

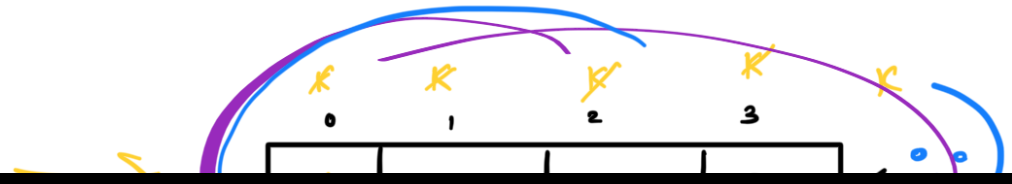
#codestorywithMIK ...

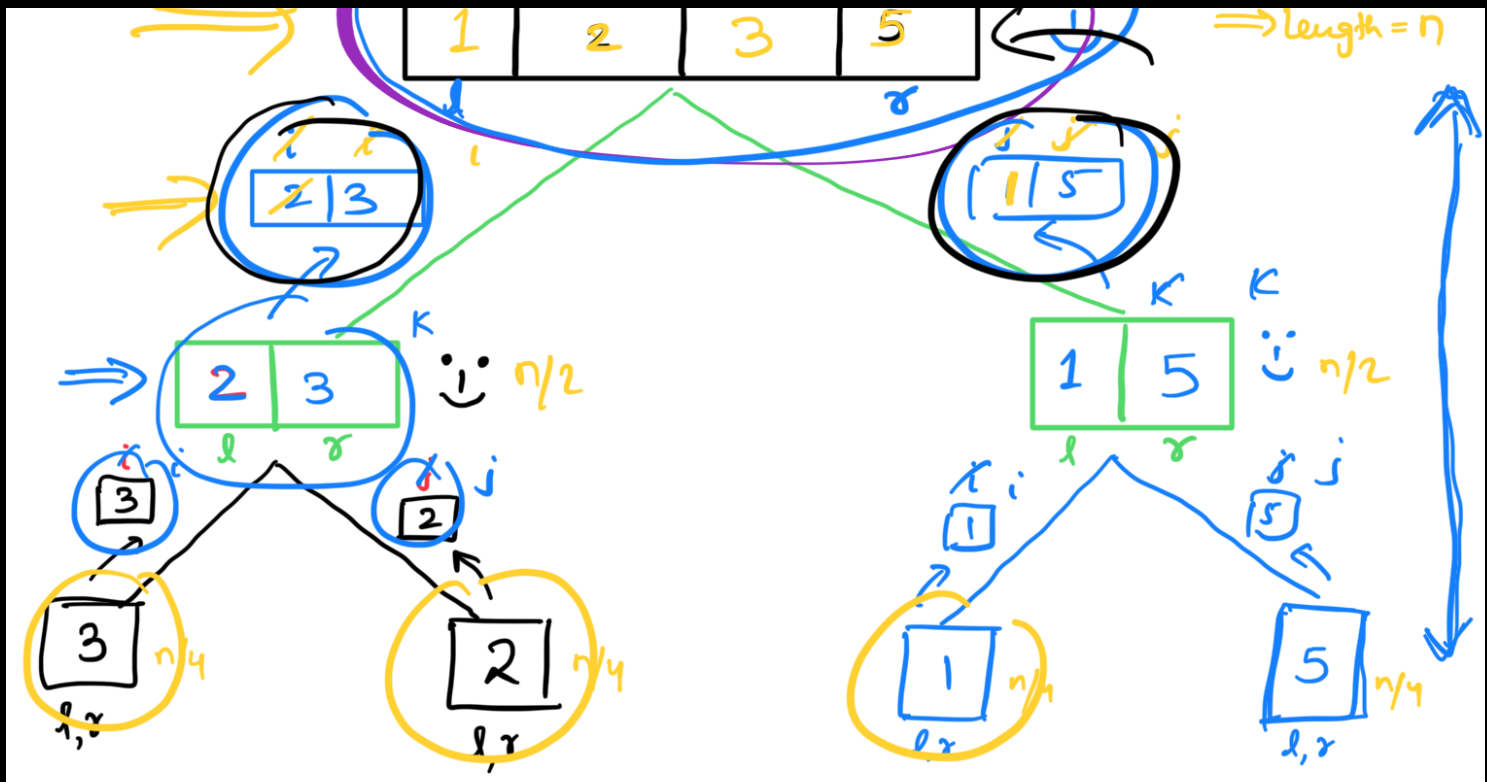
Merge Sort



merge.







$n \longrightarrow n/2 \longrightarrow n/4 \ldots 1$

$n \rightarrow n/2 \rightarrow n/4 \ldots\ldots 1 \Rightarrow \log(n)$

depth $= n * \log(n)$

$T.C = O(n * \log(n))$

merge

$n \rightarrow n/2 \rightarrow n/4 \ldots 1$

$S.C = \log(n) \rightarrow$ recursion stack of System.

Lets write Story :-

Let's write Story:

(•) array , n → Solve(array, 0, n-1) ;

(l ↙ at 0, r ↙ at n-1)

(•) mid = l + (r-l)/2 ;

↳ Solve(array, l, mid); //L

↳ Solve(array, mid+1, r); //R

↳ merge करदूँगा

(•) if length is 1 → if (l == r)
return ;

Story To Code:-

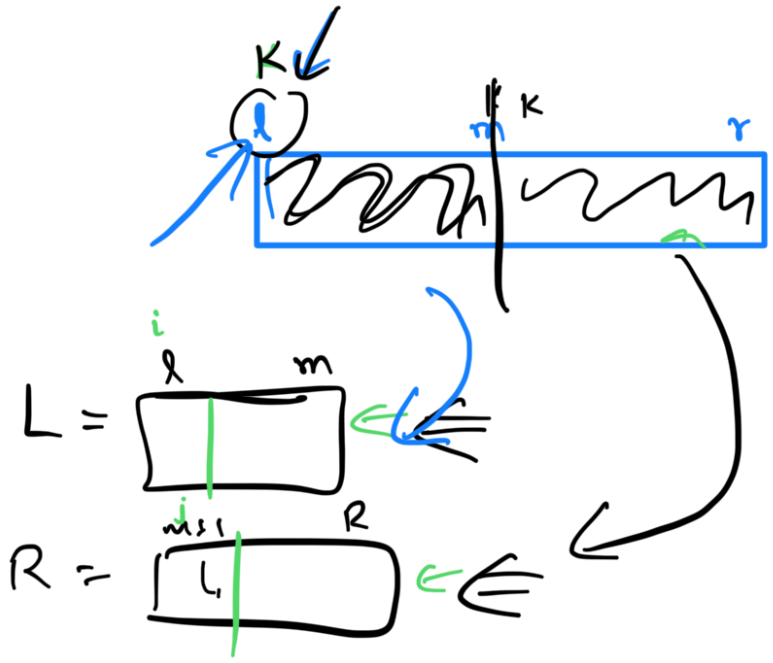
```
void mergeSort (arr, l, r) {
    if (l == r) {  // only one element.
        return;
    }
```

```
mid = l + (r-l)/2

mergesort (arr, l, m);      // Trust
mergesort (arr, m+1, r);    // Trust

merge (arr, l, m, r);

}
```



merge (arr, l, m, r) {

$L \longrightarrow m-l+1 \longrightarrow n_1$

$R \longrightarrow r-m \longrightarrow n_2$

$\lceil n_1 \rceil$

$= O(n)$.

int K = l;

```
for ( i = 0;  i<n1  ; i++) {
        L [i] = arr [K];
        K++;
}

for ( i=0;   i<n2 ;   i++) {
        R[i] = arr [K];
        K++;
}
```

// merge them in arr;

int i = 0; → L

int j = 0 → R

K = l ; → arr

```
while (i<n1  && j<n2) {
    if (L[i] <= R[j]) {
        arr[K] = L[i];
        i++;
    } else {
```

```
            arr[k] = R[j];
            j++;
        }

        k++;
    }

    while (i < n1) {
        arr[k] = L[i];
        i++;
        k++;
    }

    while (j < n2) {
        arr[k] = R[j];
        k++;
        j++;
    }

}
```