1 | 2 | 3 | 4 | 5

2 → Selection Sort

what If, I select minimum & place at right position

44 | 33 | 55 | 22 | 11 → find Min place at 0 then i++

(1) -> 11 | 33 | 55 | 22 | 11

i → index

(2) -> 11 | 22 | 55 | 33 | 44

(3) -> 11 | 22 | 33 | 55 | 44

(4) -> 11 | 22 | 33 | 44 | 55

☆ -> For ith iteration, pick smaller element from i to (n-1) index & swap it with ith index

23/9/23

Date: 2152

i -> 0 to n-1
[0, n-1)

0	1	2	3	4
44	33	55	22	44

I, i = 0, step 1 [0,4] -> find smallest index

min index = 4

Step 2 -> swap (v[i], v[min index])

i++; -> loop ke learn

II, i = 1, step 1

0	1	2	3	4
11	33	55	22	44

[1,4] (min Index) mI = 3

swap(v[i], v[mi])

III, i = 3, step 1 ->

0	1	2	3	4
11	22	55	33	44

[2,4] mI = 3

swap(v[i], v[mI])

IV -> [3,4] ->

0	1	2	3	4
11	22	33	55	44

mI = 4

swap(v[i], v[mI])

=> 11 | 22 | 33 | 44 | 55

N = 5

I i = 0 -> 4
II i = 1 -> 3
III i = 2 -> 2
IV i = 3 -> 1

= n-i-1 -> Comparisons

After 1st

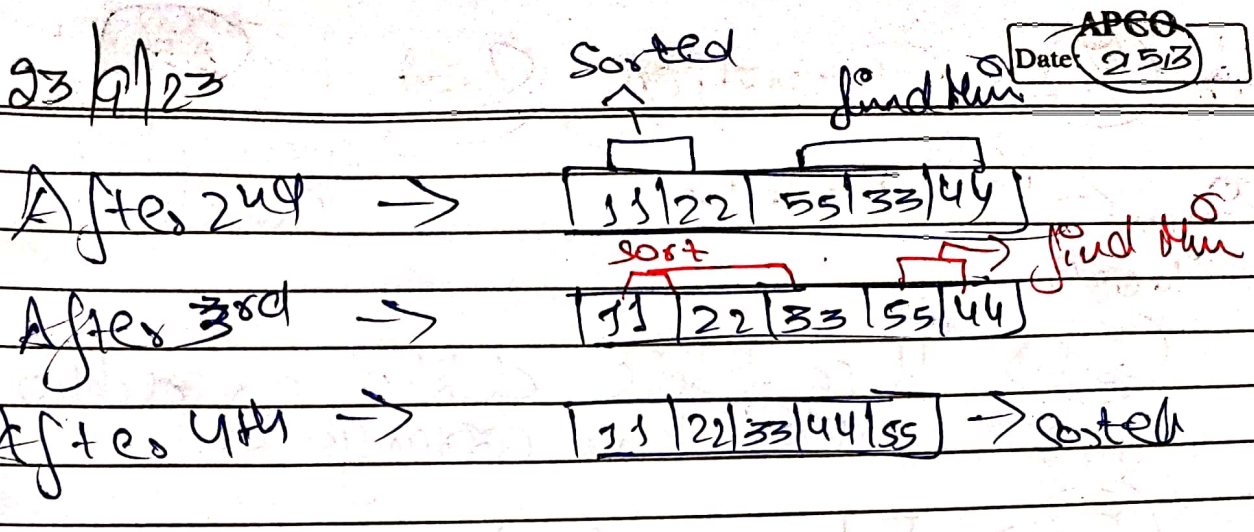
11	33	55	22	44

sort

Final Min

23/9/23

APCO
Date 25/3



Codes →

```
for(i=0; i<n-1; i++){          // n -> size of Array
    int minIndex=i;
    for(j=i+1; j<n; j++){
        if(v[j]<v[minIndex]){
            minIndex=j;
        }
    }
    swap(v[i], v[minIndex]);
}
```

TC → $O(n^2)$

Space → $O(1)$