Sorting ?

↳ arranging something in any particular order.

A → 2 1 4 5 3

asc → 1 2 3 4 5

desc → 5 4 3 2 1

odd...even → 1 5 3 2 4

Sorting.

↳ bubble sort
selection sort
insertion sort
} algo.

!o = Bubble Sort (asc → default)

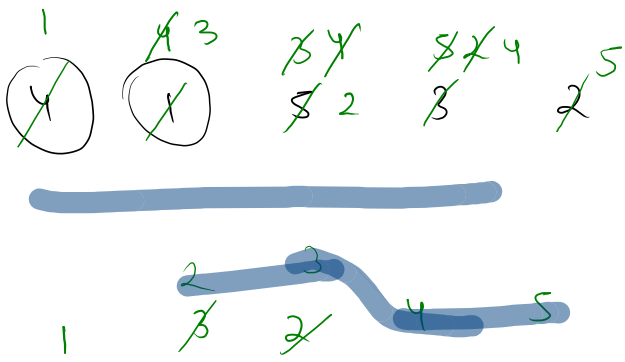
A → 4 1 3 5 2

⇓

sort → 1 2 3 4 5

Bubble Sort :-

(asc.)



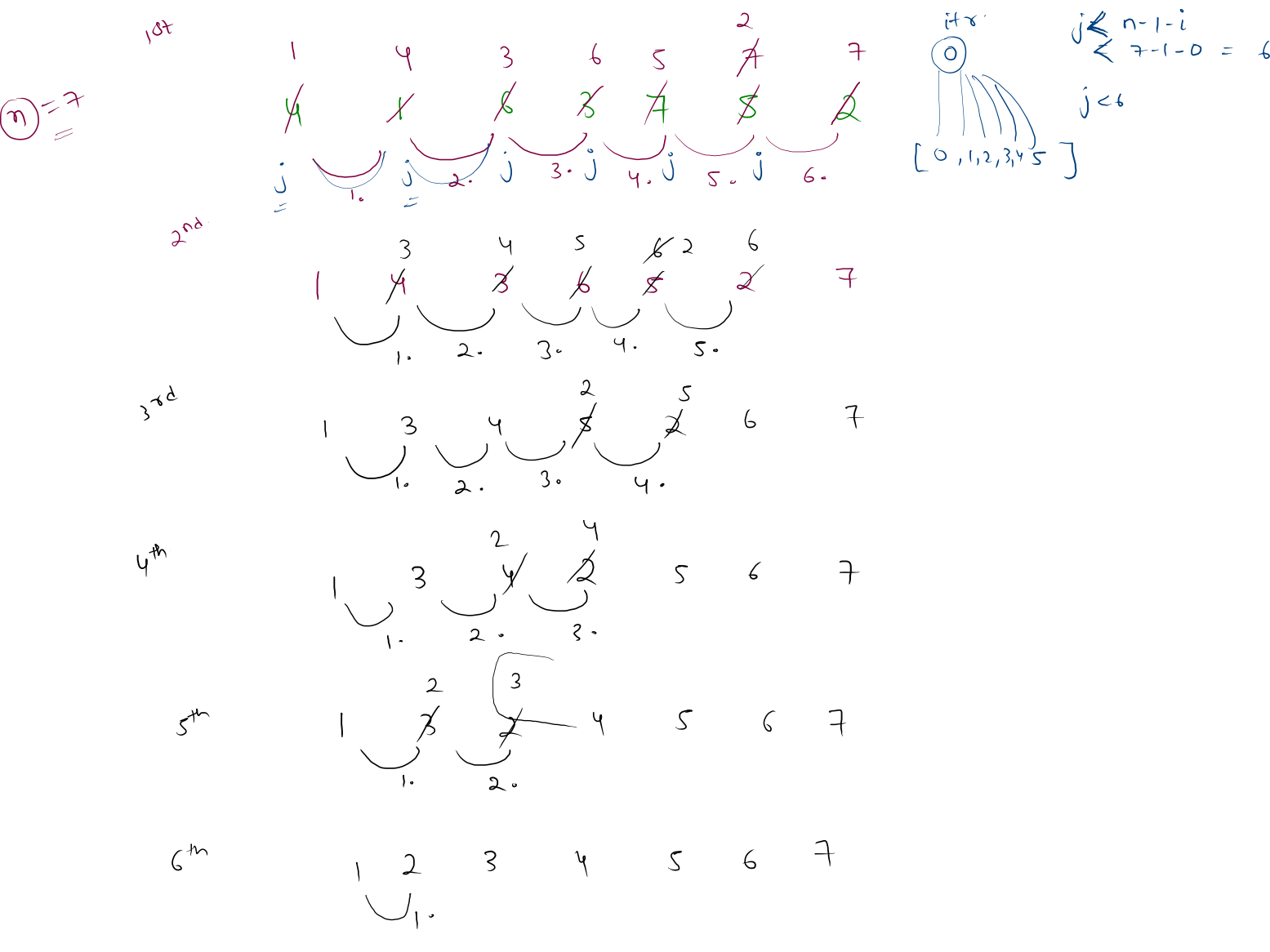
1 itr

biggest number is at correct pos.

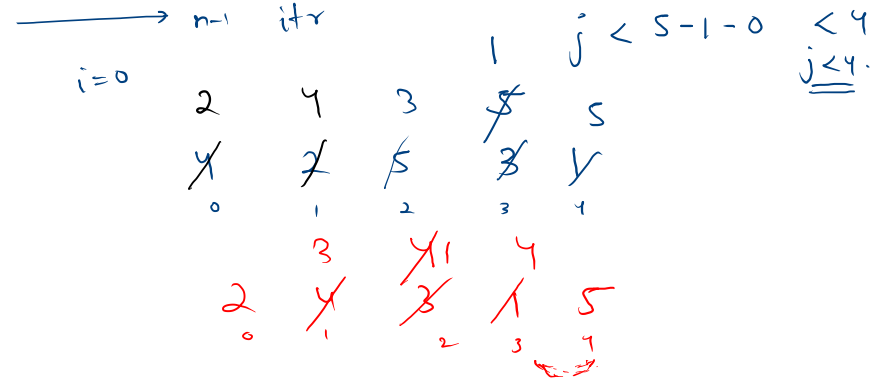
in each itr biggest ele is at correct position

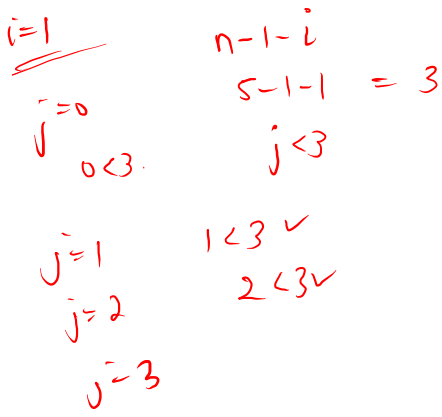
n → n-1 itr

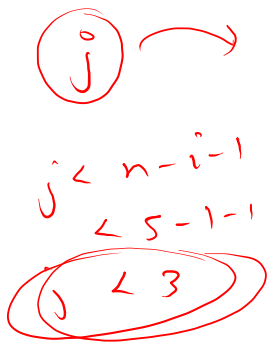
last


1st
n = 7
1 4 3 6 5 7 2
j 1. j 2. j 3. j 4. j 5. j 6.
itr
0
[0,1,2,3,4,5]
j < n-1-i
< 7-1-0 = 6
j < 6
2nd
1 3 4 5 6 2 7
1. 2. 3. 4. 5.
3rd
1 3 4 2 5 6 7
1. 2. 3. 4.
4th
1 3 2 4 5 6 7
1. 2. 3.
5th
1 2 3 4 5 6 7
1. 2.
6th
1 2 3 4 5 6 7
1.

```java
//bubble sort
for(int i = 0; i < n-1; i++){
    for(int j = 0; j < n-1-i; j++){
        if(A[j] > A[j+1]){
            int tmp = A[j];
            A[j] = A[j+1];
            A[j+1] = tmp;
        }
    }
}
```







```java
public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int n = scn.nextInt();
        int [] A = new int[n];
        for(int i = 0; i < n; i++){
            A[i] = scn.nextInt();
        }
        //bubble sort
        for(int i = 0; i < n-1; i++){
            for(int j = 0; j < n-1-i; j++){
                if(A[j] > A[j+1]){
                    int tmp = A[j];
                    A[j] = A[j+1];
                    A[j+1] = tmp;
                }
            }
        }
        //print
        for(int i = 0; i < n; i++){
            System.out.print(A[i] + " ");
        }
    }
}
```

SC O(1)

TC.

1st → n

2nd → n-1

3rd → n-2

⋮

nth → 1

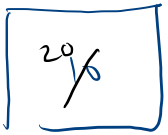
$1+2+3+\cdots+n-2+n-1+n$

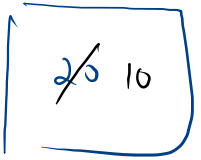
$= \frac{n(n+1)}{2}$

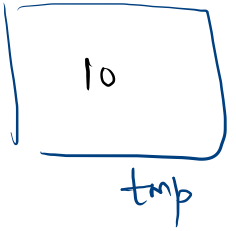
$= \frac{1}{2}n^2 + n$

$O(n^2)$

Swapping







tmp = a

a = b

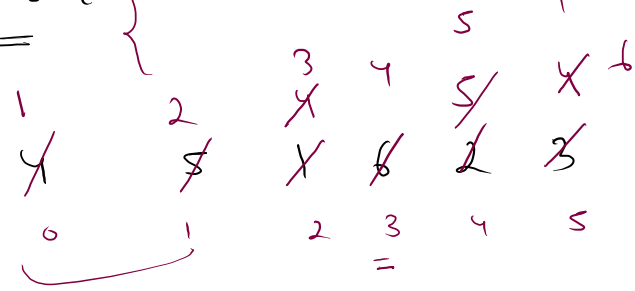
b = tmp

tmp = a

a = b

b = tmp

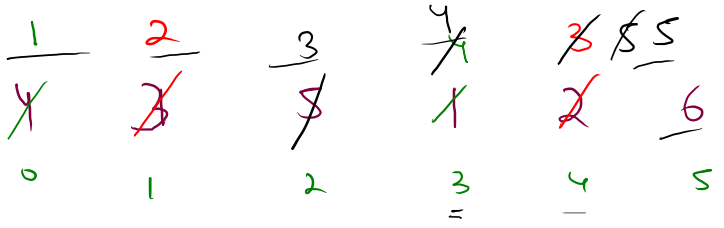
Selection Sort: { in each itr place smallest at it's correct positio



Logic: 0th itr:

i = 0
i = 1
i = 2
i = 3
i = 4

minIdx = 2
minIdx = 4
minIdx = 5
minIdx = 5
minIdx = 4



n = 6.

5 itr.

i = 0

i = 1

i = 2

i = 3

i = 4

minIdx = 3

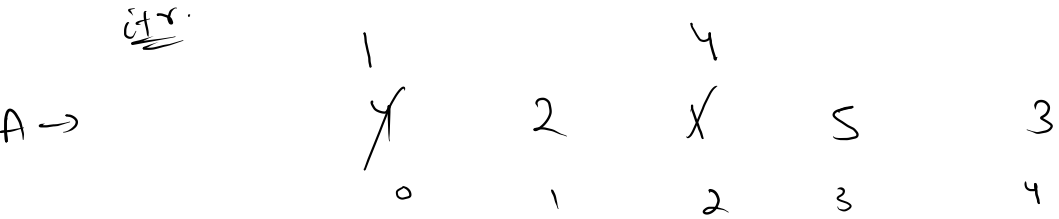
minIdx = 4

minIdx = 4

minIdx = 3

minIdx = 4

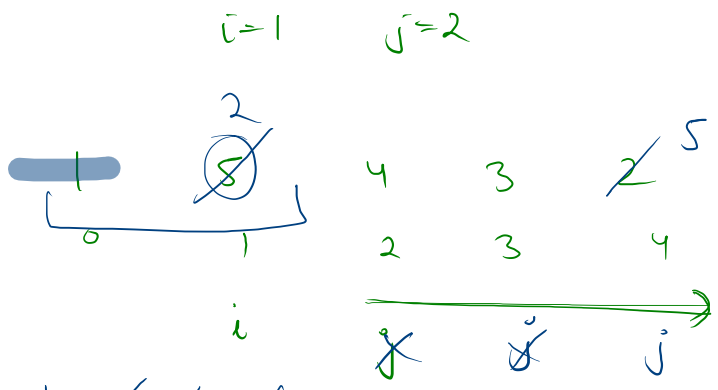
itr



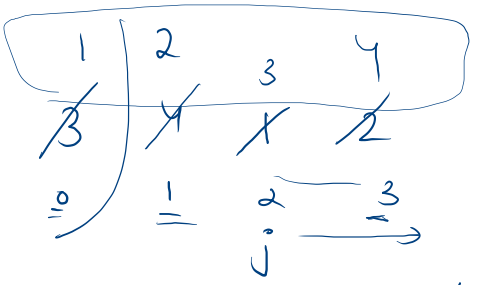
i = 0 → correct value at i^{th} idx

itr → 0^{th} idx ⇒ minIdx = 2

```java
//logic: selection sort
for(int i = 0; i < n-1; i++){
    int minIdx = i;
    for(int j = i+1; j < n; j++){
        if(A[j] < A[minIdx]){
            minIdx = j;
        }
    }
    int tmp = A[i];
    A[i] = A[minIdx];
    A[minIdx] = tmp;
}
```



```
13      //logic: selection sort
14      for(int i = 0; i < n-1; i++){
15          int minIdx = i;
16          for(int j = i+1; j < n; j++){
17              if(A[j] < A[minIdx]){
18                  minIdx = j;
19              }
20          }
21          int tmp = A[i];
22          A[i] = A[minIdx];
23          A[minIdx] = tmp;
24      }
```



i = 0

minIdx = 2

A[3] < A[2]

2 < 1

i = 1

```java
    public static void printMin(int [] A){
        int min = A[0];
        for(int i = 1; i < A.length; i++){
            if(A[i] < min){
                min = A[i];
            }
        }
        System.out.println(min);
    }
```

```java
    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int n = scn.nextInt();
        int [] A = new int[n];
        for(int i = 0; i < n; i++){
            A[i] = scn.nextInt();
        }
        //logic: selection sort
        for(int i = 0; i < n-1; i++){
            int minIdx = i;
            for(int j = i+1; j < n; j++){
                if(A[j] < A[minIdx]){
                    minIdx = j;
                }
            }
            int tmp = A[i];
            A[i] = A[minIdx];
            A[minIdx] = tmp;
        }
        //print
        for(int i = 0; i < n; i++){
            System.out.print(A[i] + " ");
        }
    }
}
```