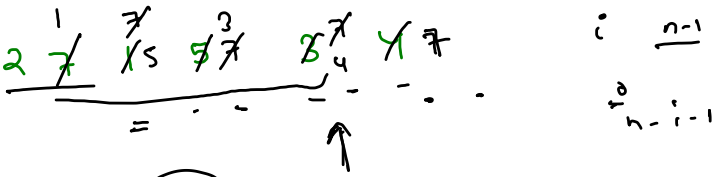
Bubble Sort.

ascending ↑ increasing.



i n-1

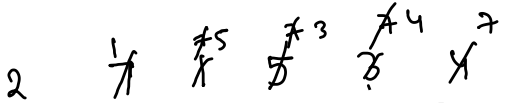
0
n-i-1

①

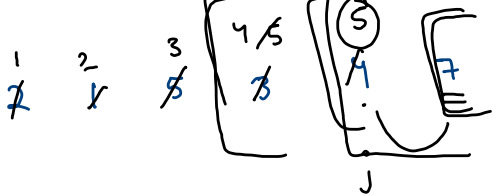
7 1 7

5 7 5

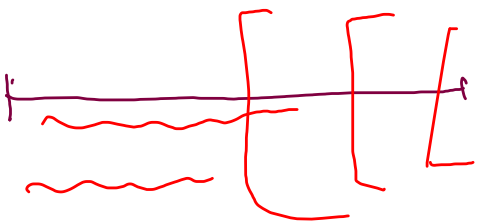
n=6



i=0 j = 0 1



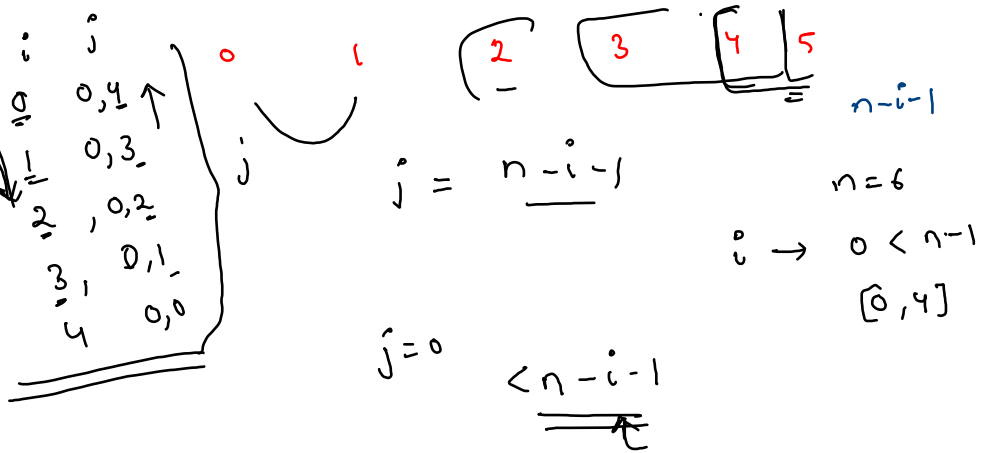
i=1 j ≥ 3



i=0

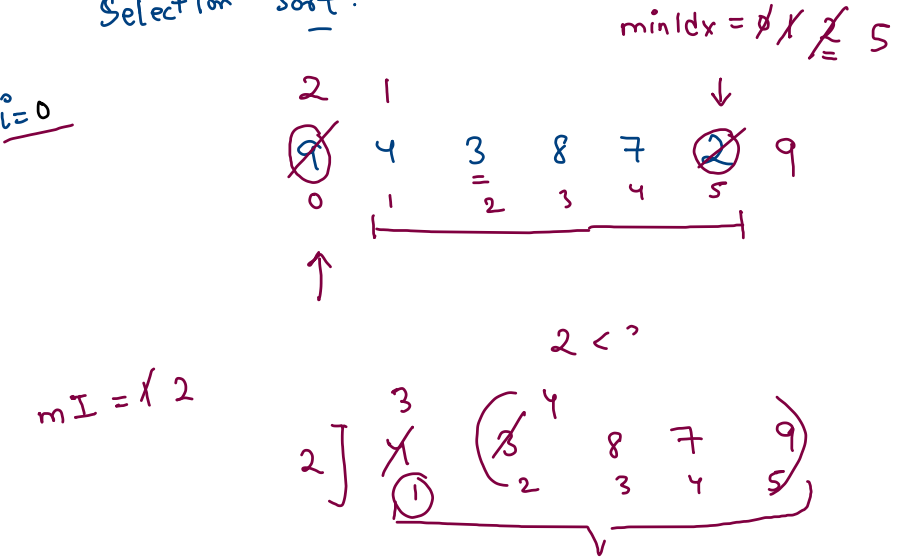
i=1

i=2




```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();
        }
        // n = 5

        // 9 4 3 8 7
        // 0 1 2 3 4
        //bubble sort
        for(int i = 0; i < n-1; i++){   //iterations
            for(int j = 0; j < n-i-1; j++){ //comparing
                if(A[j] > A[j+1]){

                    int temp = A[j];
                    A[j] = A[j+1];
                    A[j+1] = temp;

                }
            }
        }

        for(int i = 0; i < n; i++){
            System.out.print(A[i] + " ");
        }

    }
}
```

Selection Sort.




```java
import java.io.*;
import java.util.*;

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();
        }

        //selection

        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 temp = A[i];
            A[i] = A[minIdx];
            A[minIdx] = temp;
        }

        for(int i = 0; i < n; i++){
            System.out.print(A[i] + " ");
        }
    }
}
```

Inbuilt.

Algo

Wrapper Class.

→ value

↘ n. related.

Class.

↳ data member

related

member function } value.

↓

related

operation.