

# INTRO

19 July 2023

12:29

## # Selection Sort

- Select the minimum and swap it.
- after each iteration minimum element is shifted to the left.

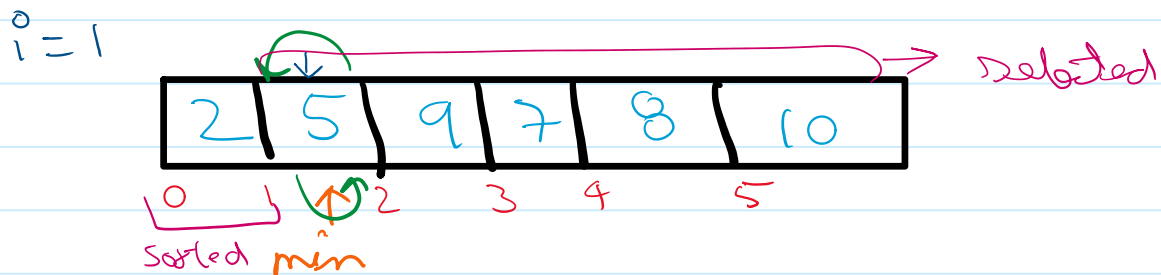
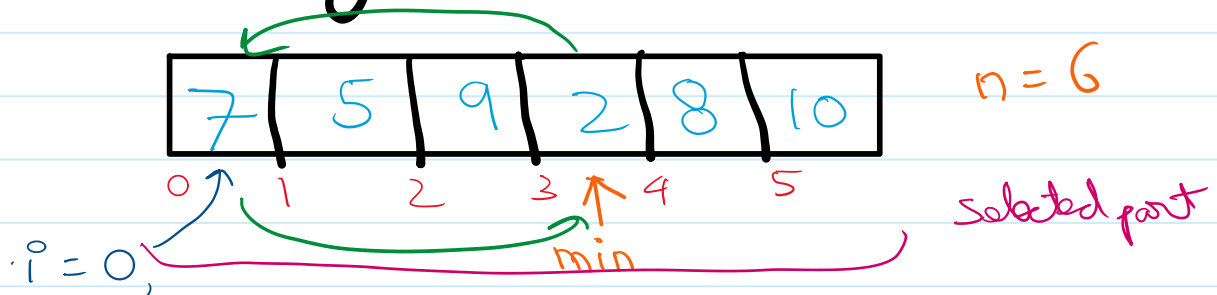
## # Example:

25	2	100	50	39	5	0	90
0	1	2	3	4	5	6	7

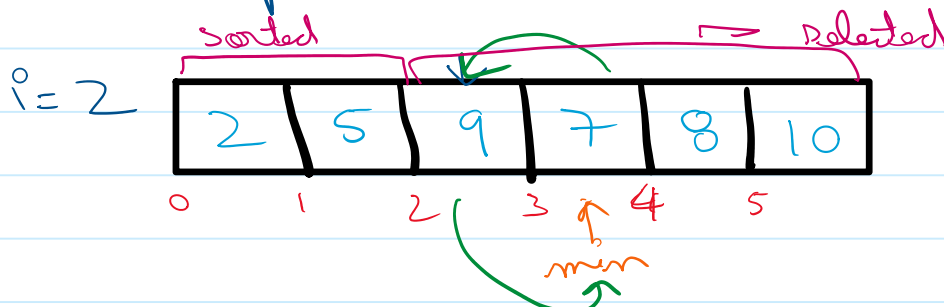
## # Algorithm:

- Select the range, from  $0 \Rightarrow n-1$ , containing all the elements, (using a for loop)
- in each iteration, select the minimum element from range of the unsorted array.
- Swap the minimum element with the first element of the selected range.
- after each iteration you will find the array is sorted upto the selected range.

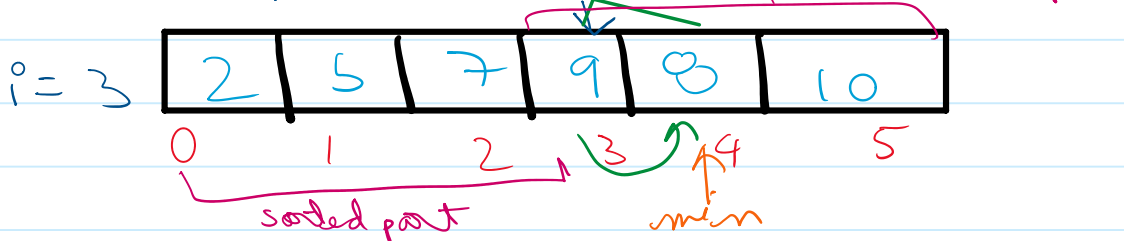
# # Dry Run:



$\rightarrow$  swap 5 with 2

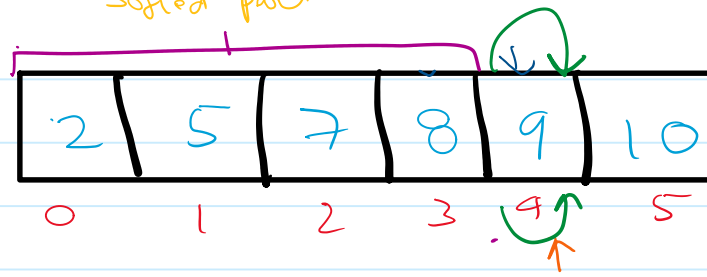


$\rightarrow$  swap 7 with 9



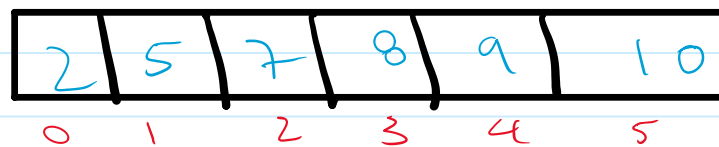
$\rightarrow$  swap 8 with 9

19 July 2023 13:02

$$i = 4$$


main related part

→ swap 9 with 9



Note: only one element is left, and one element is always sorted.

→ Stop the iteration

## # Pseudo Code:

```
for (i = 0; i <= n - 2) {
    mini = i
    for (j = i; j <= n - 1) {
        if (arr[j] < arr[mini])
            mini = j;
    }
    swap(arr[mini], arr[i]);
}
```

# CODE:

19 July 2023 13:17

## C++

→ function with, input array and array size:

```
void selectionSort(vector<int> &arr, int n)
{
    // outer loop
    for (int i = 0; i <= n - 2; i++)
    {
        int minIndex = i;
        // inner loop to find minimum element
        for (int j = i; j <= n - 1; j++)
        {
            if (arr[j] < arr[minIndex])
                minIndex = j;
        }
        // swap minimum with starting index of selected array
        int temp = arr[minIndex];
        arr[minIndex] = arr[i];
        arr[i] = temp;
    }
}
```