

Bubble Sort

0	1	2	?	9
5	9	3	2	1

Decreasing order.

Sort in increasing order

1	2	3	4	5
---	---	---	---	---

* swap the adjacent if needed. till we get all the array sorted.

1	0	1	2	?	9
	5	9	3	2	1

2	0	1	2	?	9
	9	5	3	2	1

3	0	1	2	?	9
	9	3	5	2	1

4	0	1	2	?	9
	4	3	2	5	1

5	0	1	2	?	9
	4	3	1	2	5

(1)

1	0	1	2	?	9
	3	4	2	1	5

(2)

1	0	1	2	?	9
	3	2	4	1	5

(3)

1	0	1	2	?	9
	3	2	1	4	5

(4)

1	0	1	2	?	9
	3	2	1	3	4

(5)

1	0	1	2	?	9
	1	2	3	4	5

4 3

1	0	1	2	?	9
	1	2	3	4	5

1	0	1	2	?	9
	1	2	3	4	5

1	0	1	2	?	9
	1	2	3	4	5

if ($arr[i] > arr[i+1]$) {
swap($arr[i]$, $arr[i+1]$);}

$i++$;

$i++$;

void bubbleSort(vector<int> arr)

```
for( int i=0; i<arr.size(); i++) {
```

```
    for( int j=0; j<arr.size(); j++) {
```

```
        if( arr[i] > arr[j+1]) {
```

```
            swap(arr[i], arr[j+1]);
```

```
}
```

```
else
```

```
    break;
```

```
}
```

```
}
```

```
2.
```

```
3.
```

j	i	5	4	3	2	1	?	?	?	?	?	?
0	1	5	4	3	2	1	?	?	?	?	?	?

4	1	2	3	5	?	?	?	?	?	?	?
1	2	3	4	5	?	?	?	?	?	?	?
1	2	3	4	5	?	?	?	?	?	?	?

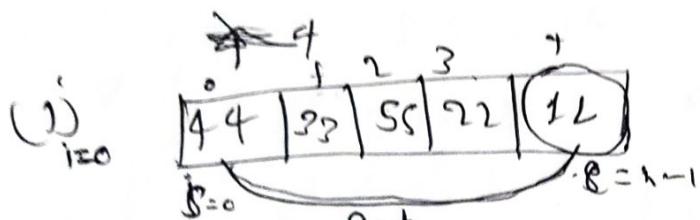
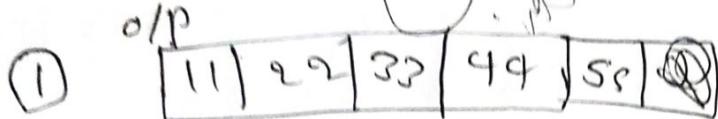
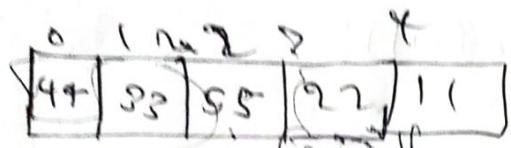
1	2	3	4	5	?	?	?	?	?	?	?
---	---	---	---	---	---	---	---	---	---	---	---

T.C $O(n^2)$

Space: $O(1)$

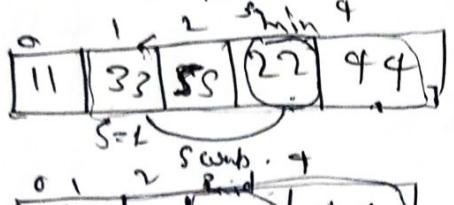
Selection sort

What if, I select the minimum element & put it
 @ right position



Find min element

i=1



for (int i=0; i<arr.size(); i++) {

int minValue = minElement(arr.begin() + i, arr.end());

arr[i] = minValue;

}

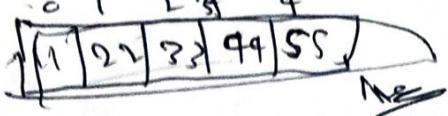
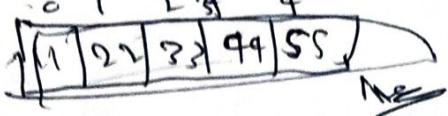
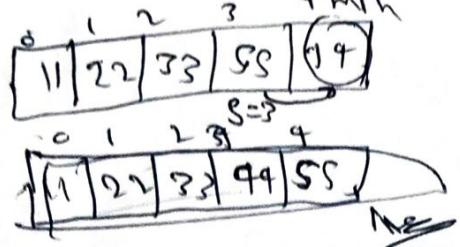
Step (1)

[0-4]

minValue = 11;

Step (2)

swap (arr[i], arr[minValue]);



→ for i th iterator, pick smallest element from $\text{arr}[i+1 \dots n]$
& swap it with i th element.

for($\text{int } i=0; i < \text{arr.size()}-1; i++$) {

 int minIndex = i ;

 for($\text{int } j=i+1; j < \text{arr.size(); } j++$) {

 if($\text{arr}[j] < \text{arr}[\text{minIndex}]$) {

 minIndex = j ;

 }

// Swap

 swap($\text{arr}[i], \text{arr}[\text{minIndex}]$);

}

for($\text{int } i=0; i < \text{arr.size(); } i++$) {

 cout < arr[i] << " ",

}

8

5

$i=0$

$\text{minIndex} = 0$

$j=1$,

① $i=0$,

$\text{mid} = j = 1$

$j = 2$,

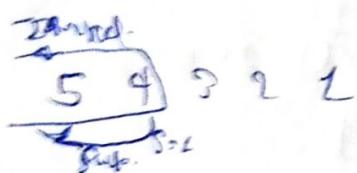
$\text{mid} = 2$

Insertion Sort

17/8/24

1st	5	4	3	2	1
-----	---	---	---	---	---

2nd	1	2	3	4	5
-----	---	---	---	---	---

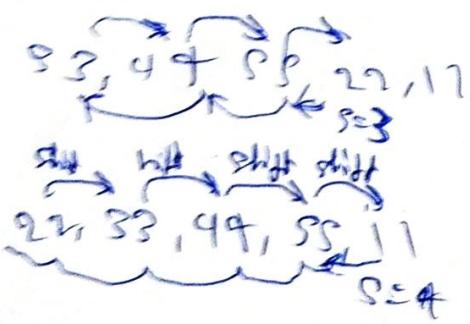


1	2	3	4	5
---	---	---	---	---

3rd	49	33	55	22	11
-----	----	----	----	----	----

4th	33	49	55	22	11
-----	----	----	----	----	----

No swap.



11	22	33	49	55	11
----	----	----	----	----	----

~~if (arr[i] > arr[i+1]) {~~

~~shift. ans = arr[i]~~

~~arr[i]~~

~~key = v[i]~~ : $i = j - 1$.

~~while (is = 0 & v[i] > key)~~

~~is~~

~~v[i+1] = v[i]~~

~~j--~~

~~j~~

~~v[i+1] = key~~.

$i = 1 \rightarrow n-n$

$i = j - 1 \rightarrow j = 0$,

void insertionSort(vector<int> arr) {

 int n = arr.size();

 for (int i = 1; i < n; i++) {

 int key = arr[i];

 int j = i - 1;

 key = ?.

 j = 0.

 while (j >= 0 && arr[j] > key) {

 arr[j + 1] = arr[j];

 arr[0] = 94.

 j = j - 1;

 }

 arr[j + 1] = key;

}

 for (int i = 0; i < n; i++) {

 cout << arr[i] << " ";

}

}

T.C Locality

0	1	2	3	4
94	33	00	22	11

j ↑

SC = $O(1)$.

Custom Comparator

C++ STL

for

sort(). sort(arr.begin(), arr.end());
arr

|↳ sort(arr, arr+n); The ascending

arr [10] 3 7 16 11 20 sort(arr.begin(), arr.end());

arr [3] 7 10 11 16 20 Inverting orders

decreasing order sort

bool mycomp(int &a, int &b) {

 return a > b; // decreasing order

}

int main () {

 sort (arr.begin(), arr.end(), mycomp);

$\left[\begin{matrix} 10, 44 \\ 10, 44 \end{matrix} \right], \left[\begin{matrix} 0, 85 \\ 0, 85 \end{matrix} \right], \left[\begin{matrix} 0, 72 \\ 0, 72 \end{matrix} \right],$
 $\left[\begin{matrix} 0, 10 \\ 0, 10 \end{matrix} \right], \left[\begin{matrix} 2, 33 \\ 2, 33 \end{matrix} \right] .$

1 sort w.r.t. Second element.

Initial

0 \rightarrow $\left[\begin{matrix} 1, 44 \\ 1, 44 \end{matrix} \right]$

1 \rightarrow $\left[\begin{matrix} 0, 85 \\ 0, 85 \end{matrix} \right]$

2 \rightarrow $\left[\begin{matrix} 0, 72 \\ 0, 72 \end{matrix} \right]$

3 \rightarrow $\left[\begin{matrix} 0, 10 \\ 0, 10 \end{matrix} \right]$

4 \rightarrow $\left[\begin{matrix} 2, 33 \\ 2, 33 \end{matrix} \right]$