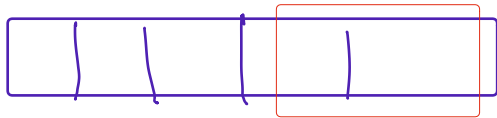
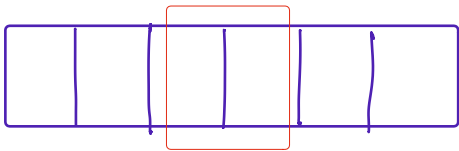
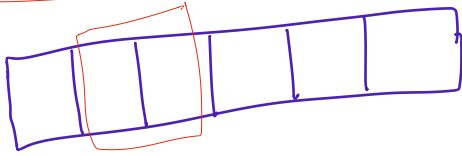
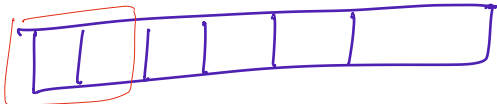


Bubble Sort

13	46	24	52	20	9
----	----	----	----	----	---

We have to do the adjacent swap in increasing order so after one round it will give $n-1$ as max value



Idea

Push the max

to the last

by adjacent

swap

13, 46, 24, 52, 20, 9

13, 46, 24, 52, 20, 9
↗ ↖

13, 24, 46, 52, 20, 9

13, 24, 46, 52, 20, 9
↗ ↖

13, 24, 46, 20, 52, 9

13, 24, 46, 20, 9, 52



one round of swapping

max num is at end

now repeated for

index 0 to $n-2$

13, 24, 46, 20, 9, 52

13, 24, 20, 46, 9, 52

13, 24, 20, 9,	46, 52
----------------	--------

13, 24, 20, 9, 46, 52

13, 20, 24, 9, 46, 52

13, 20, 9,	24, 46, 52
------------	------------

13, 20, 9, 24, 46, 52

13, 9, 20, 24, 46, 52

9, 13, 20, 24, 46, 52

0 — n-1

n-2

n-3

n-4

n-5

0 — 1

```

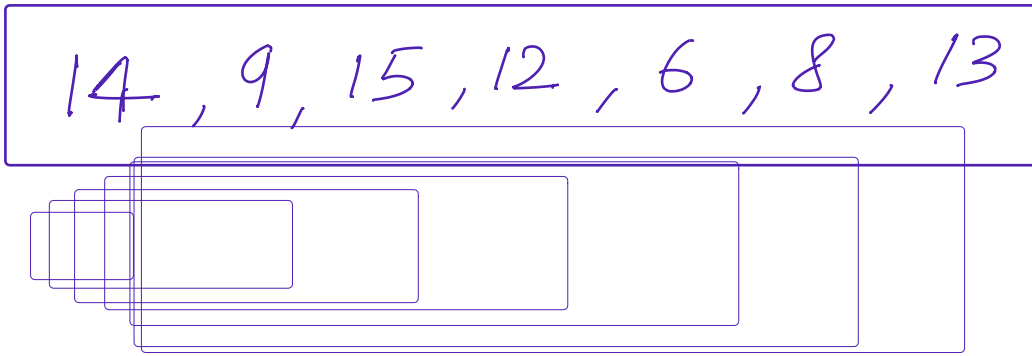
for (i = n-1; i >= 1; i--)
{
    didSwap = 0;
    for (j = 0; j < i-1; j++)
    {
        if (a[j] > a[j+1])
        {
            swap(a[j], a[j+1]);
            didSwap = 1;
        }
    }
    if (didSwap == 0) {
        break;
    }
}

```

$O(N^2)$ Worst
Average }

Best $\rightarrow O(N)$

Insertion Sort



Index {	0	1					
	0	1	2				
	0	1	2	3			
	0	1	2	3	4		
	0	1	2	3	4	5	
	0	1	2	3	4	5	6

Place it in correct order.

Idea

Take an element

4 place it in correct order.

14 9 15 12 6 8 13

9 14 15 12 6 8 13

9 14 15 12 6 8 13

9 12 14 15 6 8 13

6 9 12 14 15 8 13

6 8 9 12 14 15 13

6 8 9 12 13 14 15

```
for (i=0; i < n-1; i++)
```

```
{ j = i;
```

```
while (j > 0 && a[j-1] > a[j])
```

```
{
```

```
    swap(a[j-1], a[j])
```

```
    j--;
```

```
}
```

```
}
```

Worst $\rightarrow O(n^2)$

Average case $\rightarrow O(n^2)$

Best $\rightarrow O(n)$