

Bubble sort

Simeon Monov

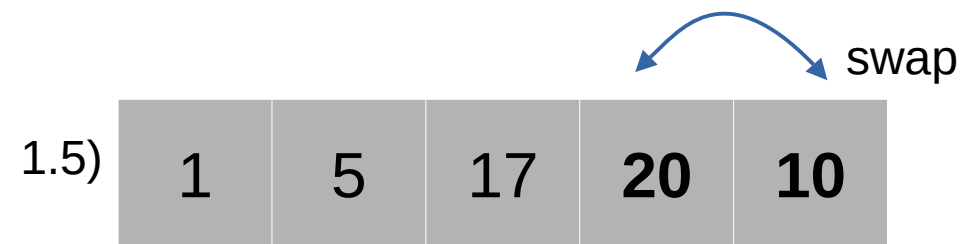
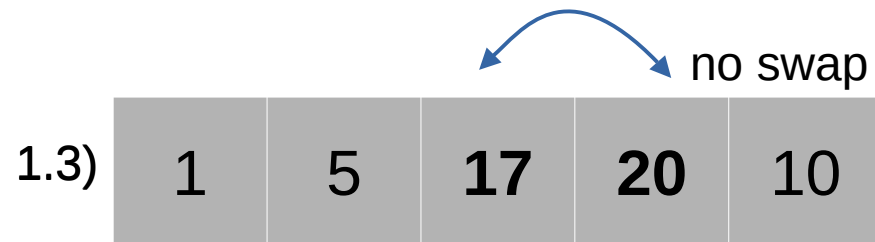
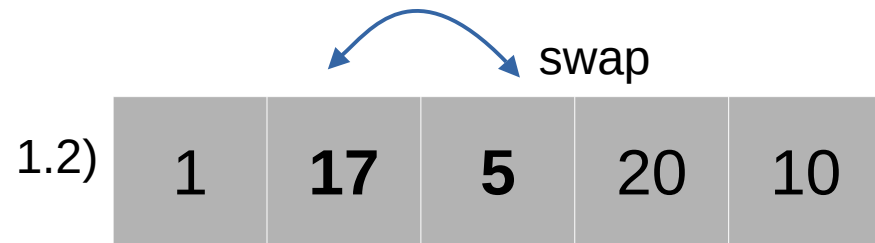
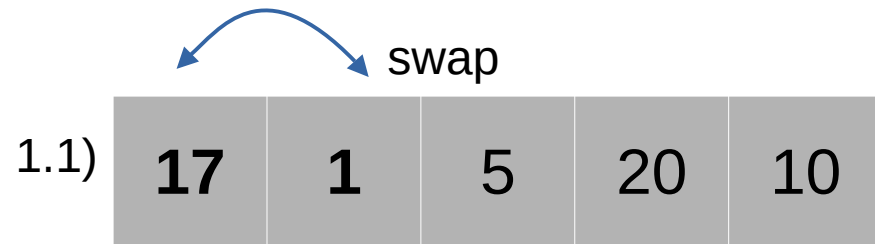
Bubble sort

17	1	5	20	10
----	---	---	----	----

- Traverse the array $n-1$ times or until sorted;
- On each iteration check each pair of numbers and swap them in a way, that higher numbers stay to the right;
- The result will be on each iteration we will have the highest number go to the most right in the sub-array.

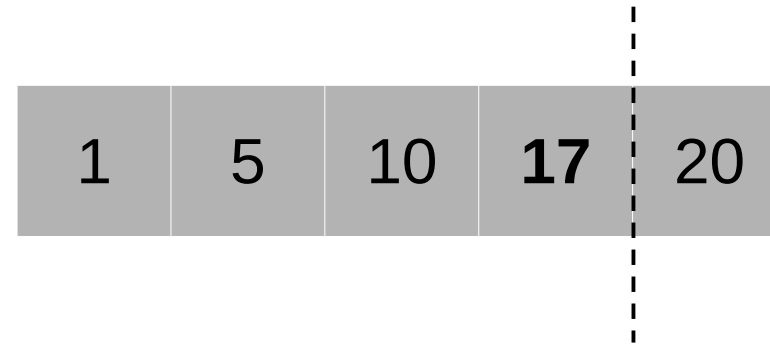
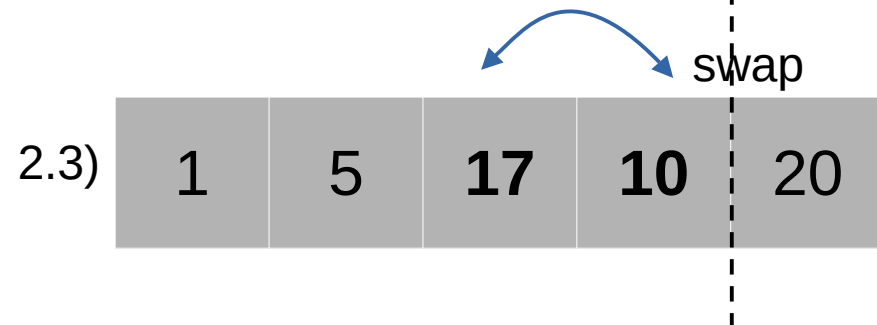
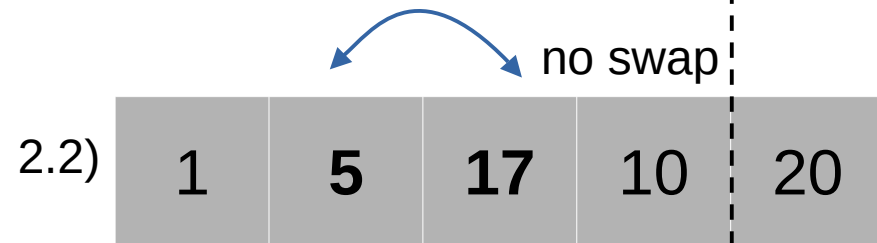
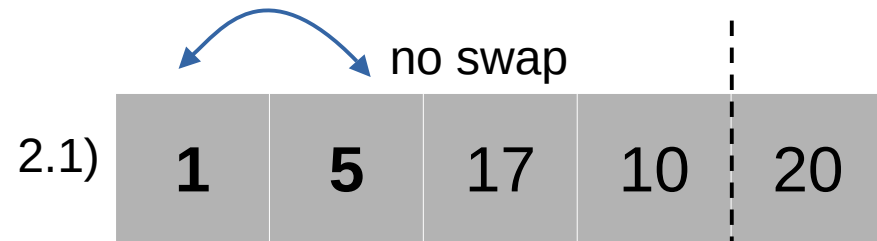
Bubble sort

Check which is higher and swap if needed



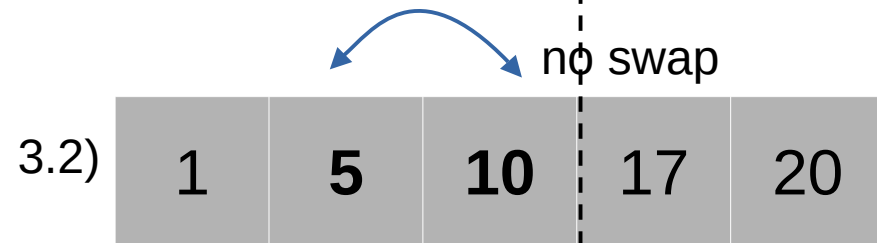
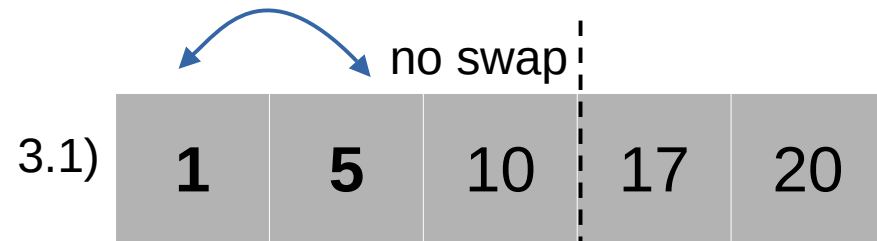
Bubble sort

Check which is higher and swap if needed



Bubble sort

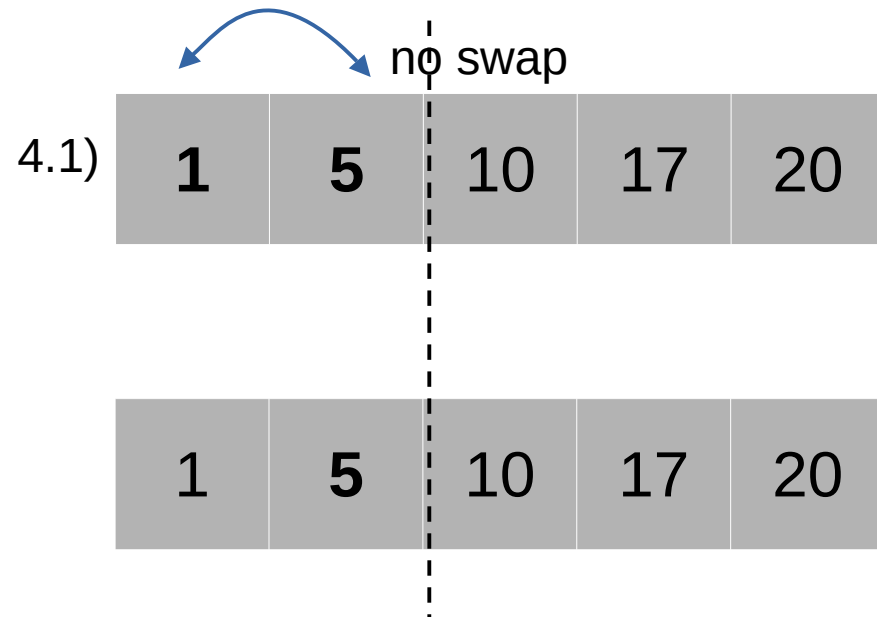
Check which is higher and swap if needed



- We can end the algorithm here, because there were no swaps in this iteration. This means the array is already sorted and no need to continue.

Bubble sort

Check which is higher and swap if needed



Bubble sort

Final result :

1	5	10	17	20
---	---	----	----	----