Insertion Sort

In this lesson, we'll learn how insertion sort works and see the implementation.

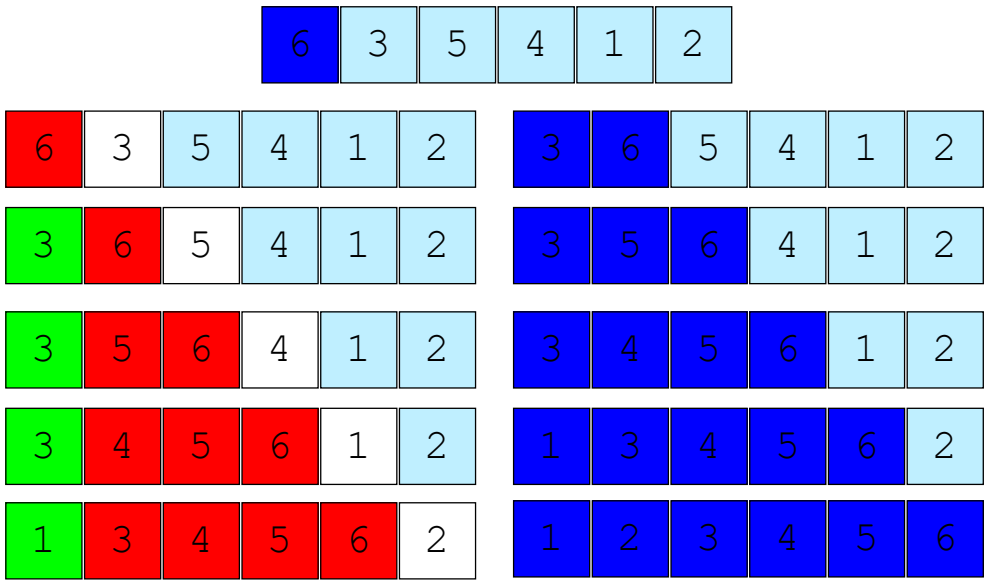
We'll cover the following

- Insertion sort

Insertion sort

Insertion sort maintains the sorted part of the array at the beginning of the array. For `ith` element, `arr[0..i-1]` is sorted; we then search for the position where `arr[i]` belongs, insert it at that position, and shift the affected elements to right.

Red-colored elements are to the right of where current elements (white) are supposed to be inserted, we move all red elements to the right by one place.



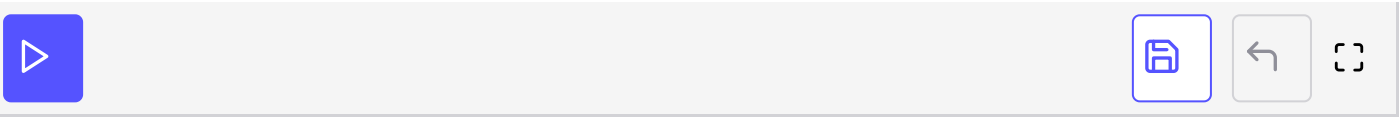

```
#include <bits/stdc++.h>
using namespace std;

int main() {
  int N = 6;
  int arr[N] = {6, 3, 5, 4, 1, 2};

  for (int i = 0; i < N; i++) {
    int j = i;
    while (j >= 1 && arr[j] < arr[j-1]) {
```

```
      swap(arr[j], arr[j-1]);
      j--;
    }
  }

  for (int i = 0; i < N; i++) cout << arr[i] << " "; cout << "\n";
  return 0;
}
```



In the next lesson, we'll start with some more efficient sorting algorithms.