Sorting Data

Selection sort

By the end of this video you will be able to…

- Explain the selection sort algorithm
- Write code to perform selection sort

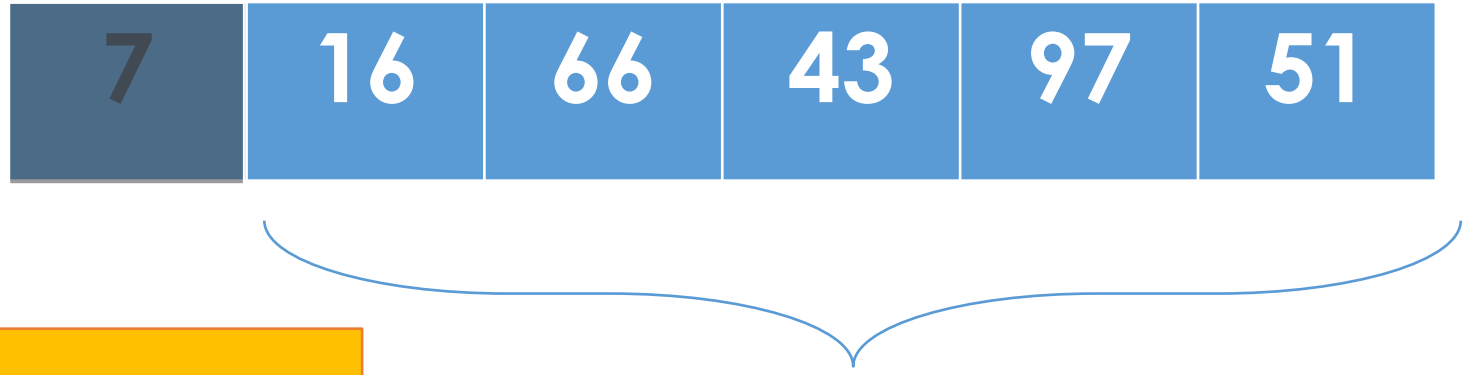
7	16	66	43	97	51

7	16	66	43	97	51

Smallest?


7
16
66
43
97
51

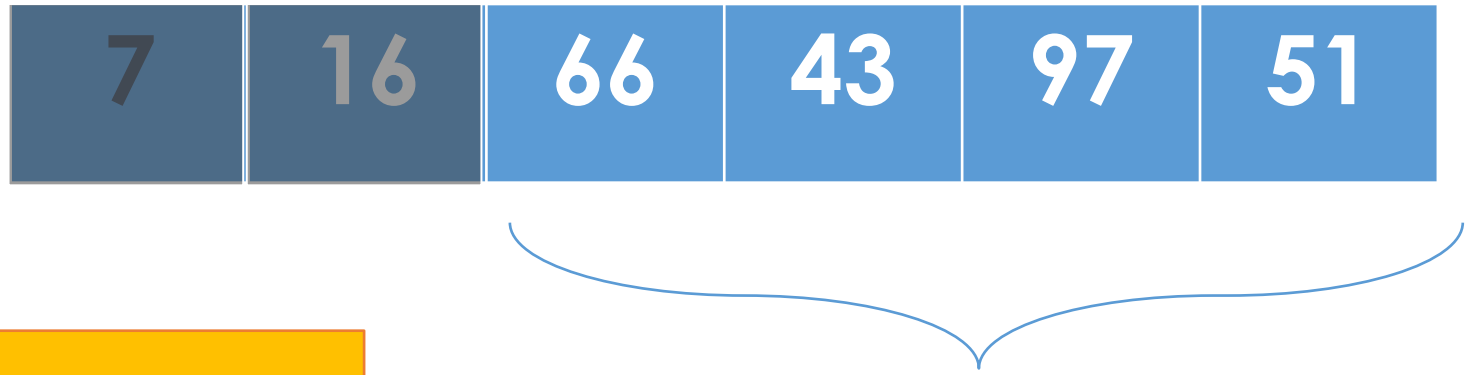

7
16
66
43
97
51



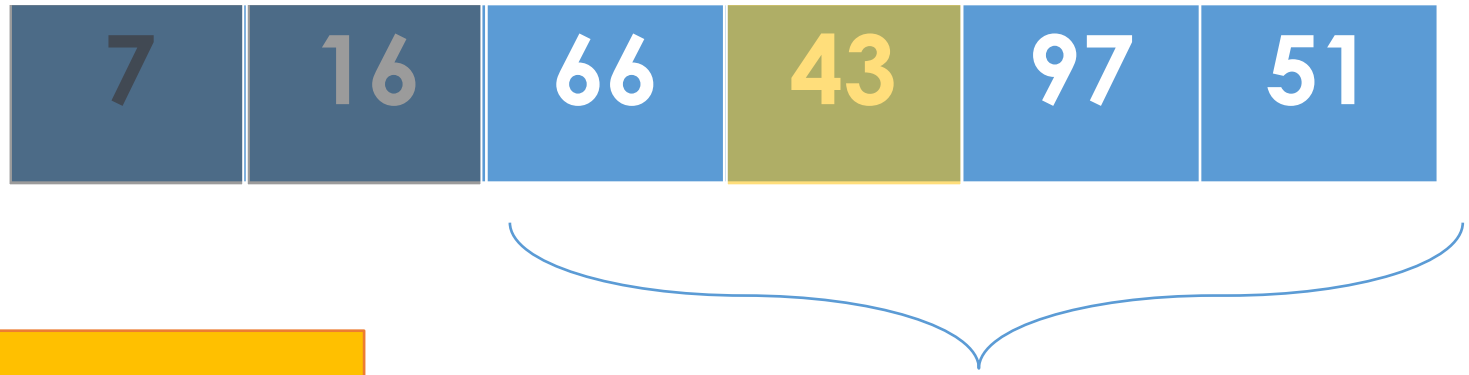
Smallest?


7
16
66
43
97
51


7
16
66
43
97
51

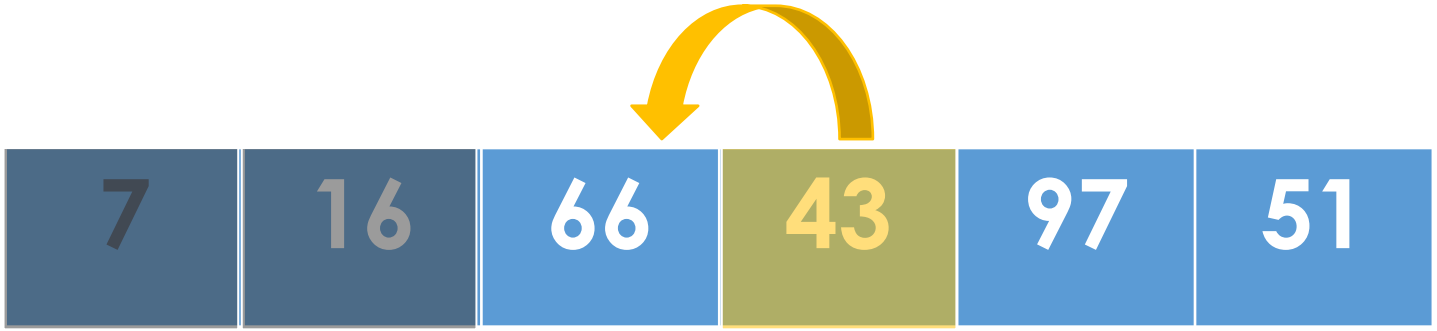


Smallest?

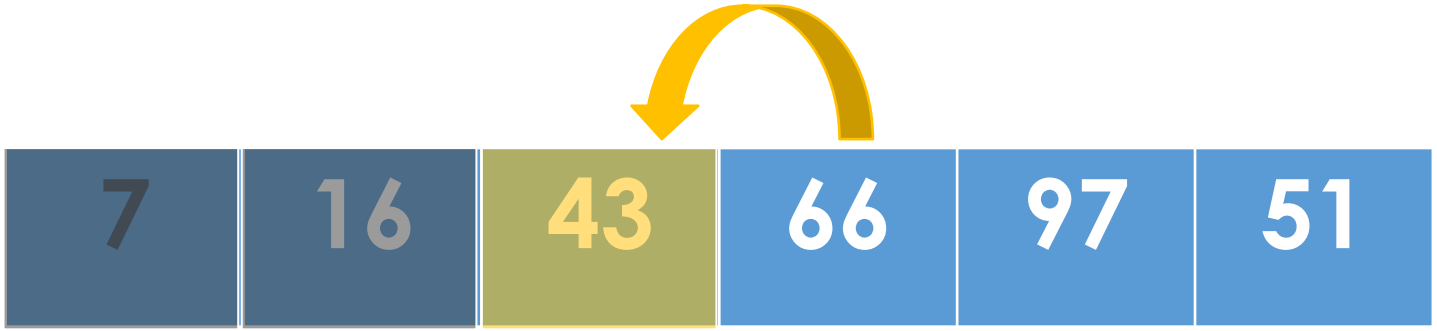


Smallest?

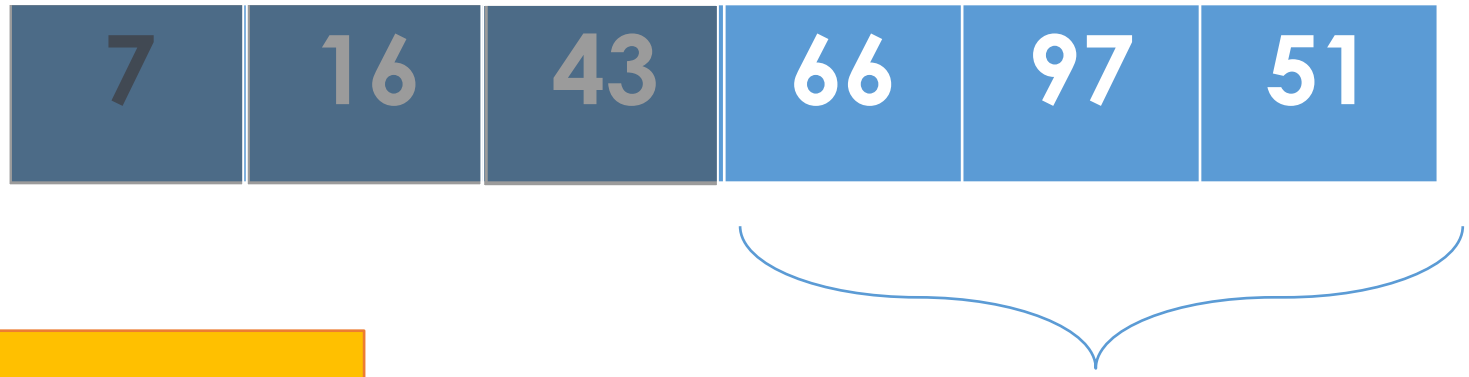
Move 43 to its correct position



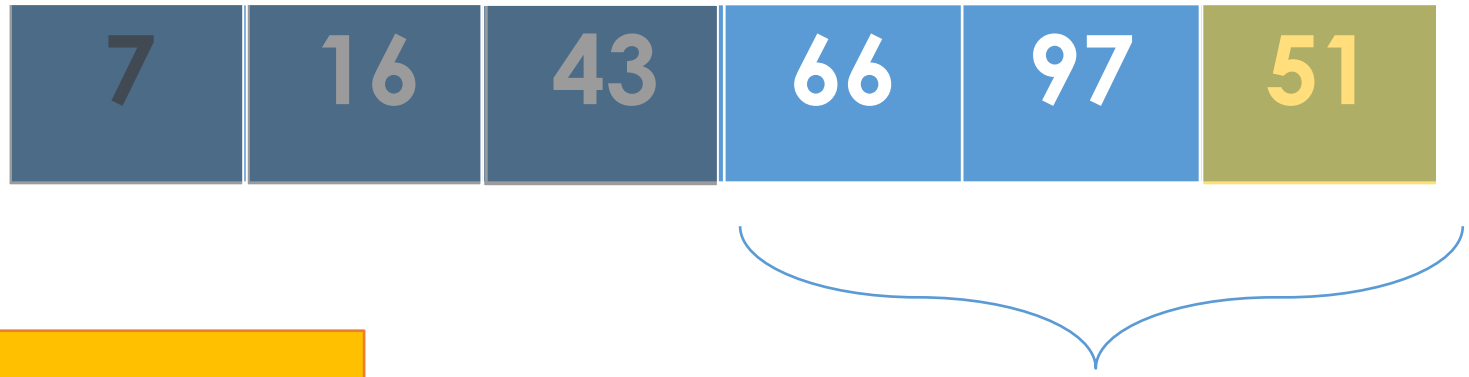
Move 43 to its correct position



Move 43 to its correct position

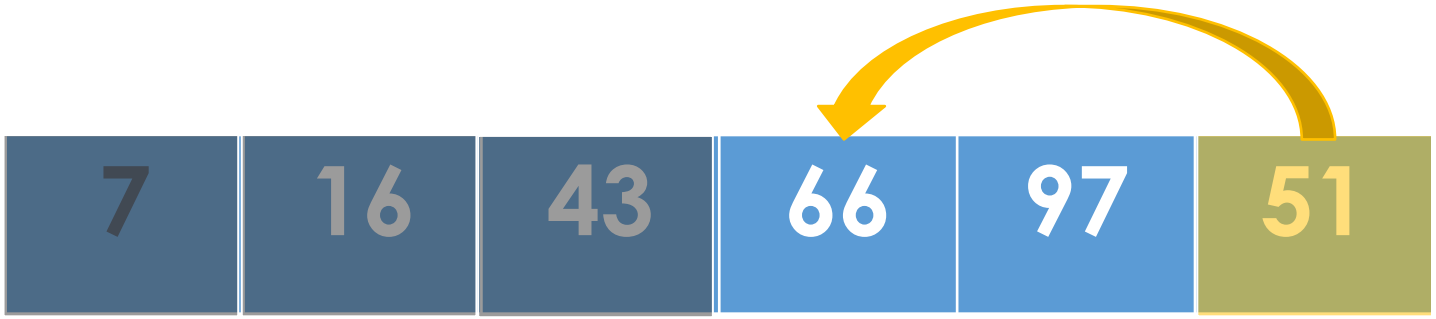


Smallest?

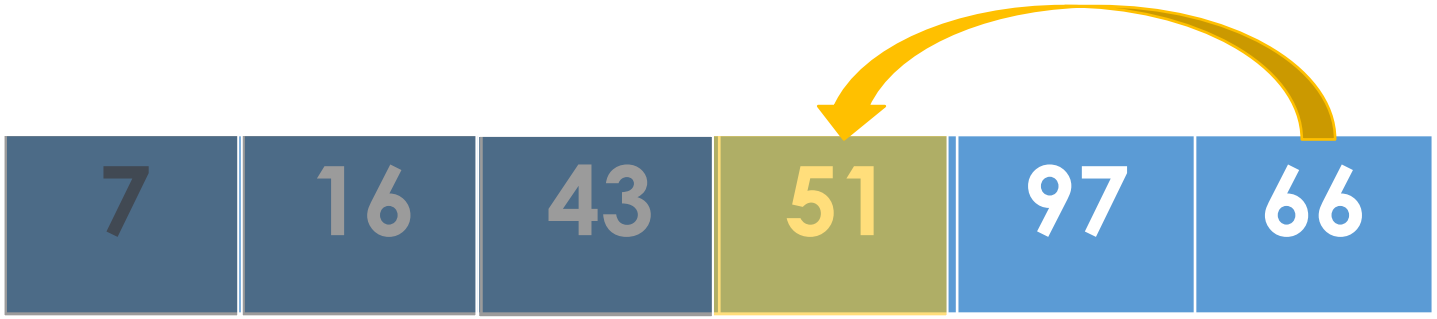


Smallest?

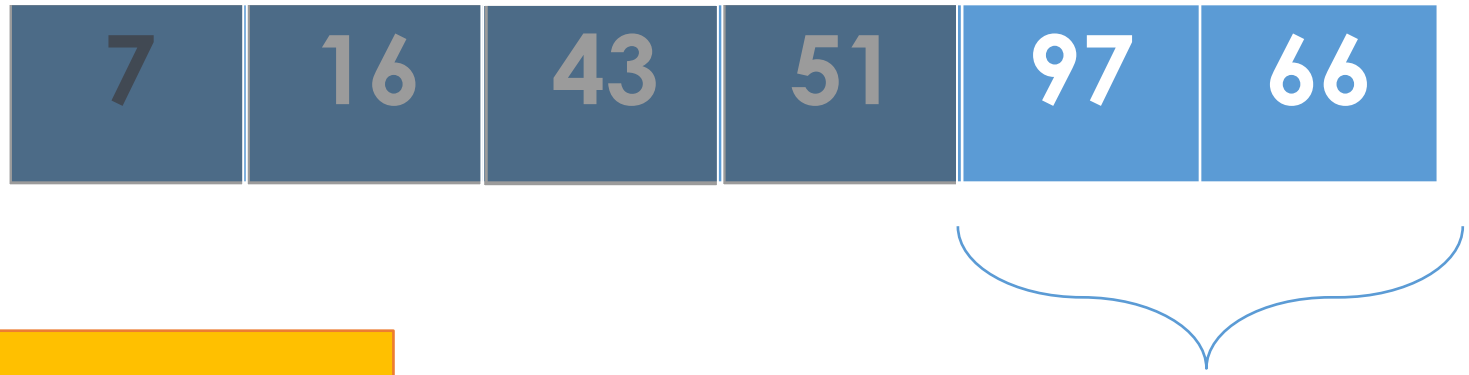
Move 51 to its correct position



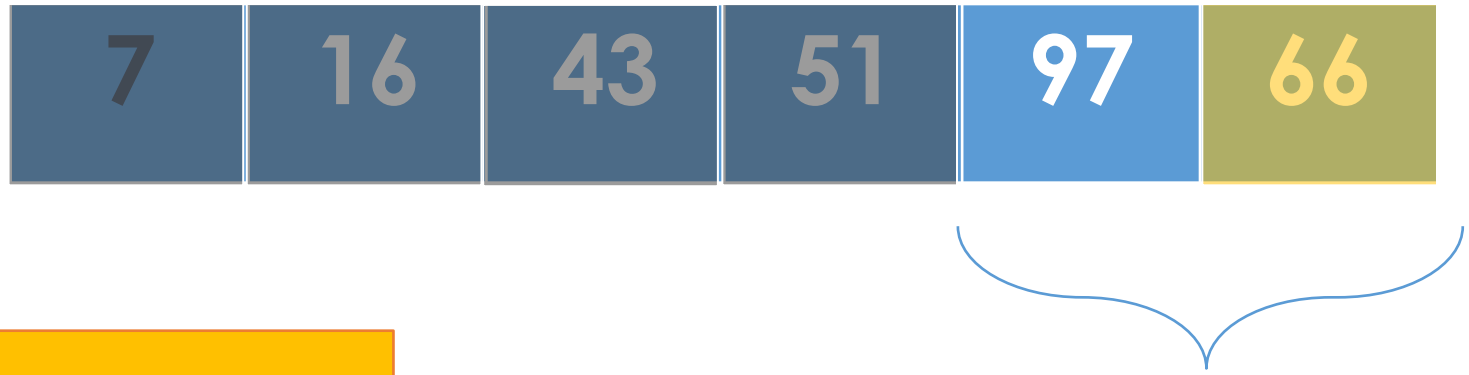
Move 51 to its correct position



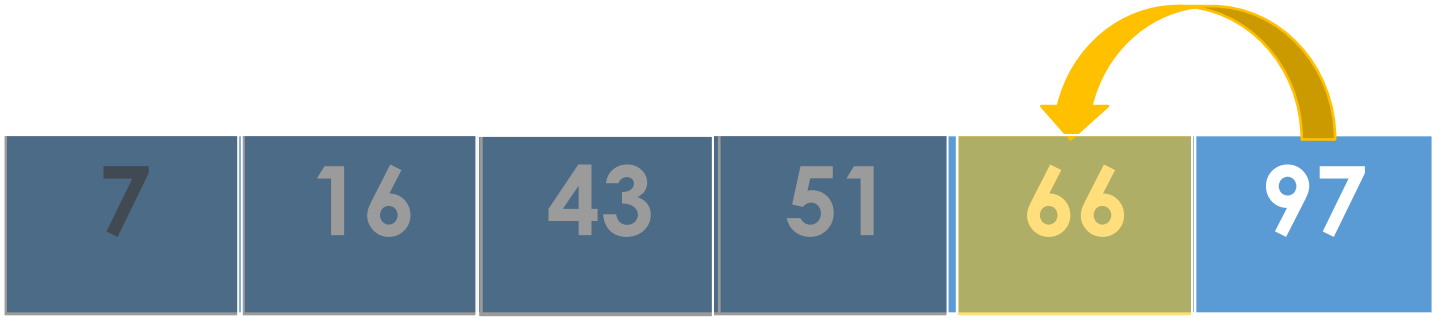
Move 51 to its correct position



Smallest?



Smallest?



Move 66 to its correct position

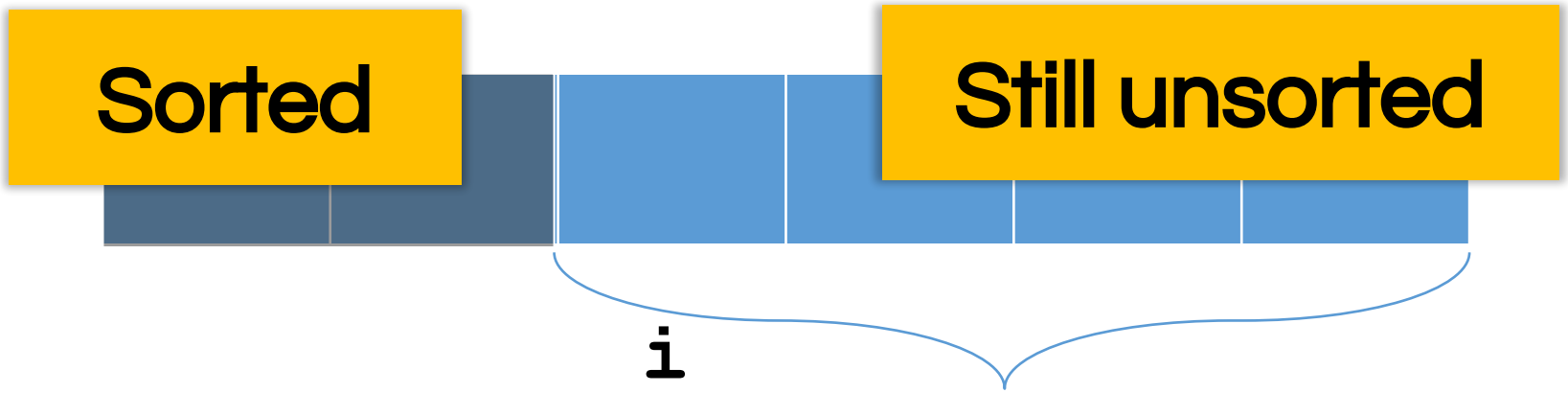
Last item must be in correct position!

- Find smallest element, swap it with element in location 0
- Find next smallest element swap it with element in location 1
- etc.

Selection Sort

Selection Sort: Basic Algorithm

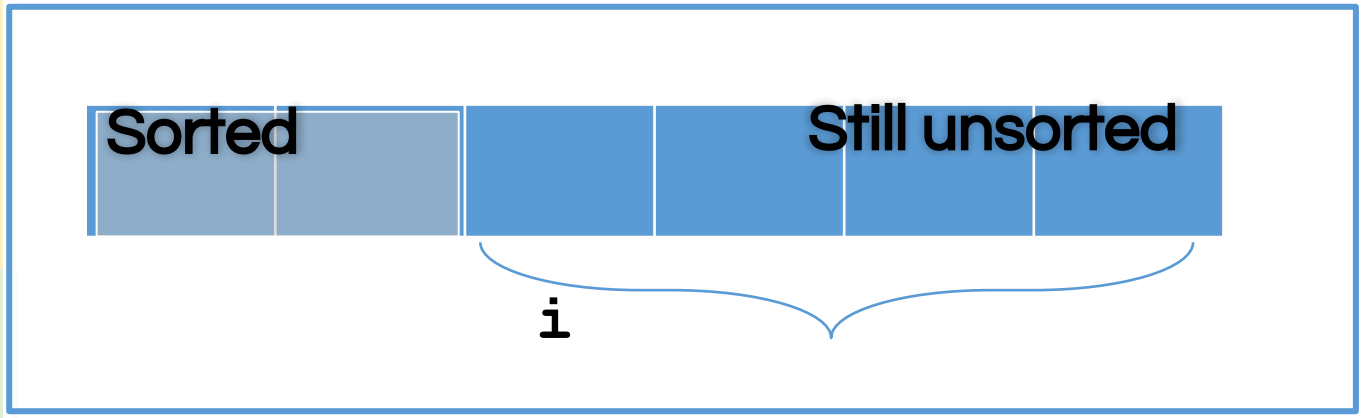
For each **position `i`** from `0` to **`length-2`**



Selection Sort: Basic Algorithm

For each **position `i`** from `0` to **`length-2`**

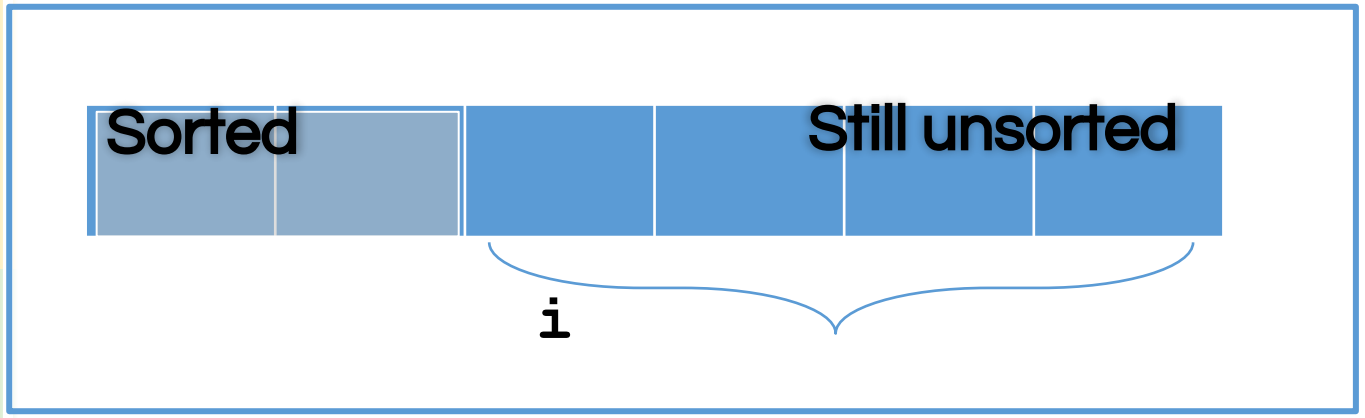
Find smallest element in "still unsorted"
Swap it to **position `i`**



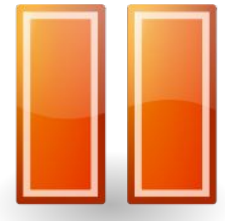
Selection Sort: Basic Algorithm

For each **position `i`** from `0` to **`length-2`**

Find smallest element in **positions `i` to `length-1`**
Swap it with element in **position `i`**

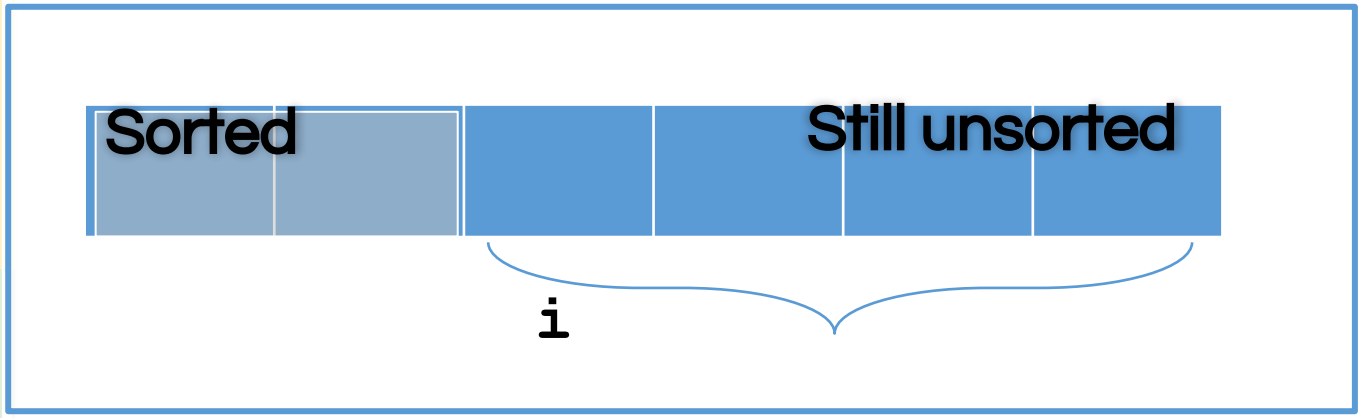


Selection Sort: Basic Algorithm



For each **position `i`** from `0` to **`length-2`**

Find smallest element in **positions `i` to `length-1`**
Swap it with element in **position `i`**



```
public static void selectionSort( int[] vals )     {



}
```

```
public static void selectionSort( int[] vals )     {

   int minI;

   for ( int i=0; i < vals.length-1 ; i++ ) {









    }
   }
}
```

```
public static void selectionSort( int[] vals )     {

   int minI;

   for ( int i=0; i < vals.length-1 ; i++ ) {
    minI = i ;

    for ( int j=0; j < vals.length; j++ ) {
      if ( vals[j] < vals[minI] ) {
       minI = j ;
      }
    }



   }
}
```

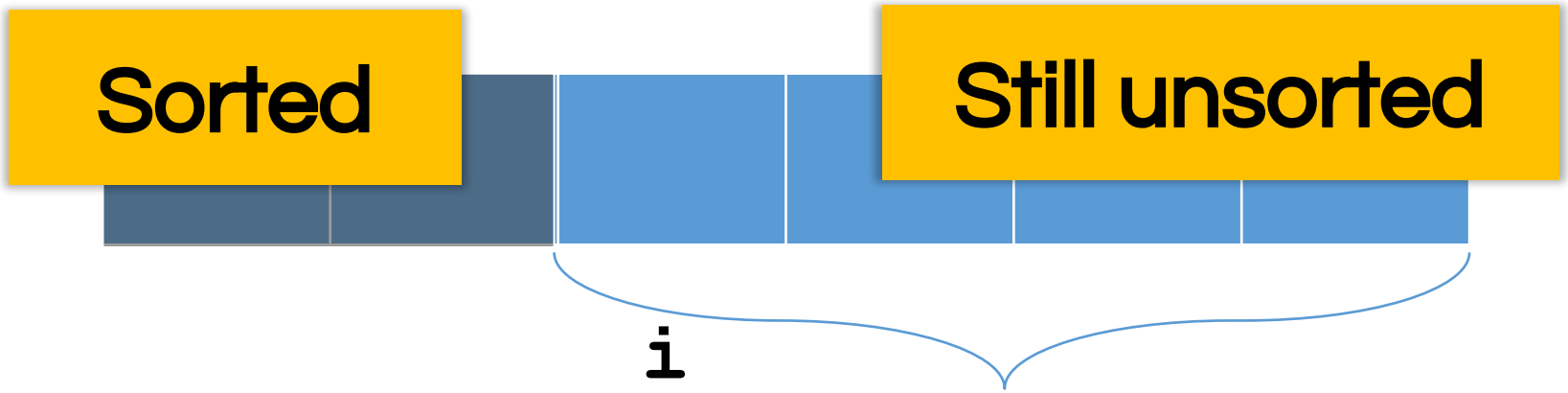
```
public static void selectionSort( int[] vals )     {

   int minI;

   for ( int i=0; i < vals.length-1 ; i++ ) {
    minI = i ;

    for ( int j=0; j < vals.length; j++ ) {
      if ( vals[j] < vals[minI] ) {
       minI = j ;
      }
    }

    if ( minI != i ) {
      swap ( minI , i );
    }
   }
}
```

Thought questions

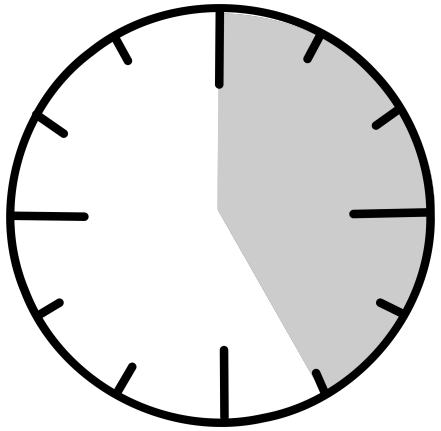
- How do we know this algorithm works?
- Are there other approaches?
- Can we do better?

Correctness

Correctness



Performance



Selection Sort: Basic Algorithm

For each **position `i`** from `0` to **`length-2`**

Find smallest element in **positions `i` to `length-1`**
Swap it with element in **position `i`**