$O(n^2)$ Sorting

Time Complexity of Sorting

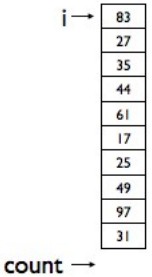
- a really bad, yet "correct" sorting algorithm:
  - shuffle the array randomly
  - check to see if it is in order
  - if yes, done
  - if not, repeat
- the probability of never generating the correct order is zero
- the time complexity is O(n!)

$O(n^2)$ Sorting

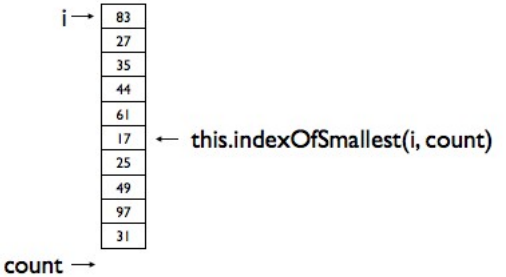
- $O(n^2)$ sorting algorithms take O(n) steps to place each of n elements into its correct position
- insertion sort
- bubble sort
- others

```
public class Array {
    private Comparable[] val;
    private int count;

    public Array (int max) {
        this.val = new Comparable[max];
        this.count = 0;
    }

    public void swap (int i, int j) {
        Comparable temp = this.val[i];
        this.val[i] = this.val[j];
        this.val[j] = temp;
    }
.
.
.
}
```

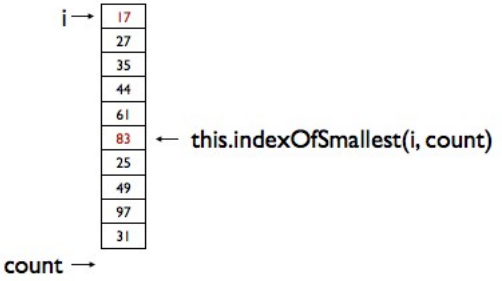
Insertion Sort



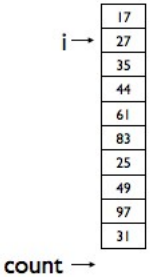
Insertion Sort



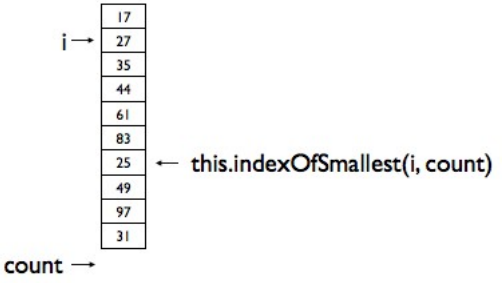
Insertion Sort



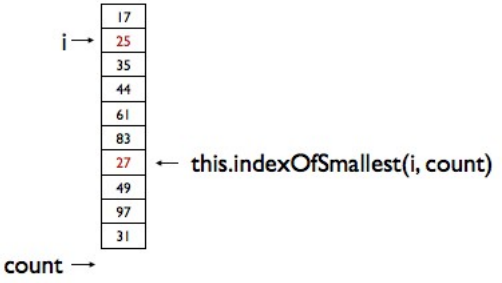
Insertion Sort



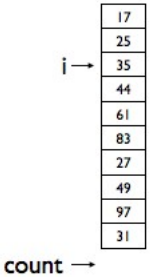
Insertion Sort



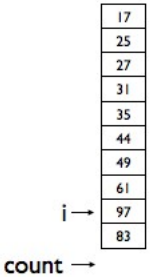
Insertion Sort



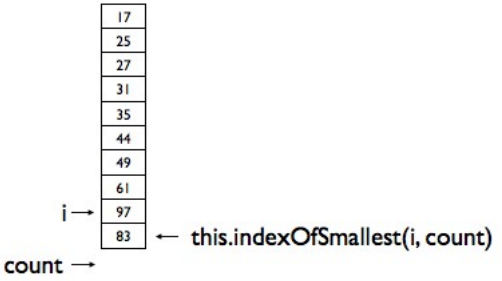
Insertion Sort



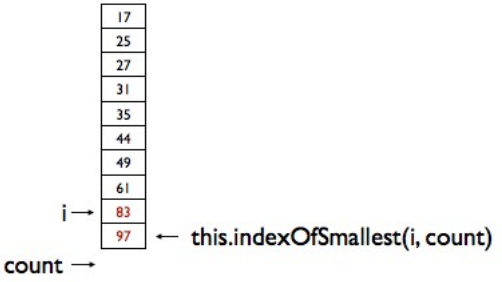
Insertion Sort



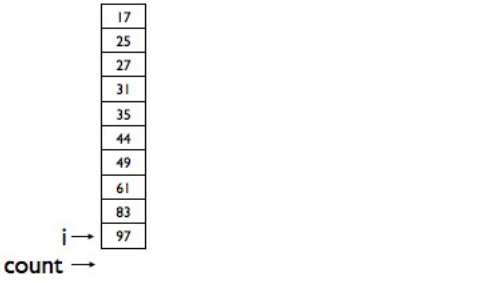
Insertion Sort



Insertion Sort

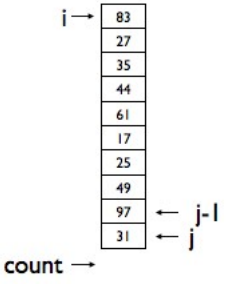


Insertion Sort

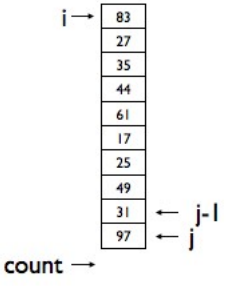


```
public class Array {
.
.
.
    public void insertionSort () {
        for (int i = 0; i < this.count - 1; i++)
            this.swap(i,this.indexOfSmallest(i, this.count));
    }

    private int indexOfSmallest (int top, int out) {
        int small = top;
        for (int i = top + 1; i < out; i++)
            if (this.val[small].compareTo(this.val[i]) > 0)
                small = i;
        return small;
    }
.
.
.
}
```

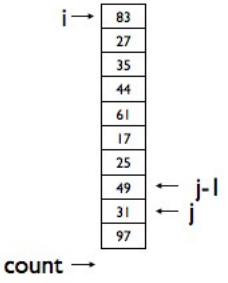
Bubble Sort



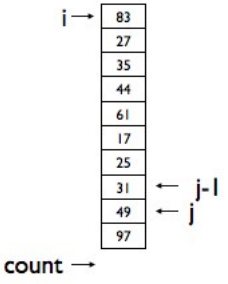
Bubble Sort



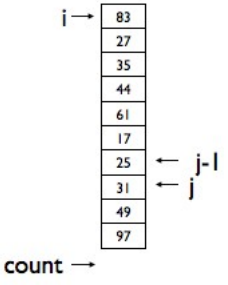
Bubble Sort



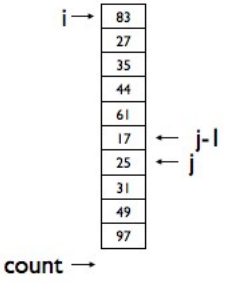
Bubble Sort



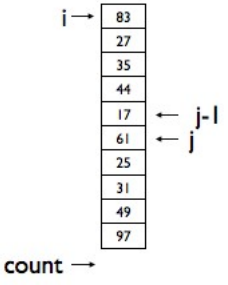
Bubble Sort



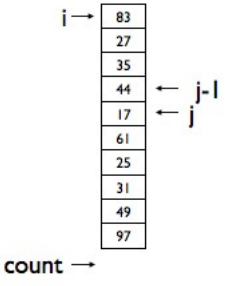
Bubble Sort



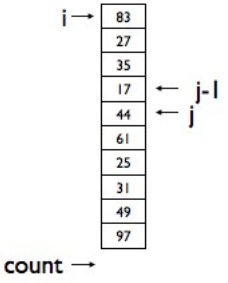
Bubble Sort



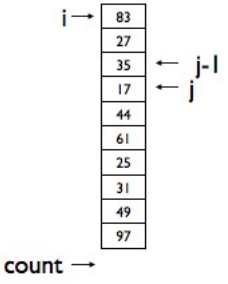
Bubble Sort



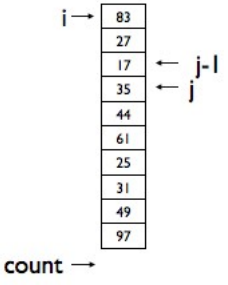
Bubble Sort



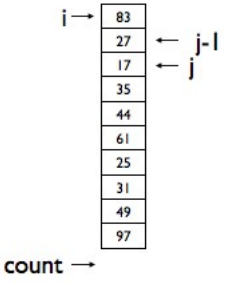
Bubble Sort



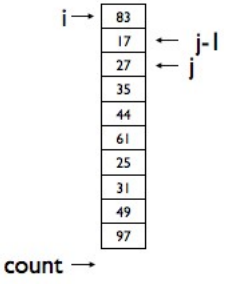
Bubble Sort



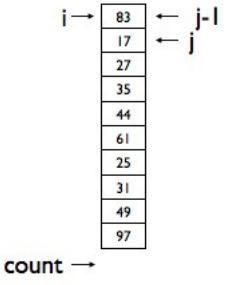
Bubble Sort



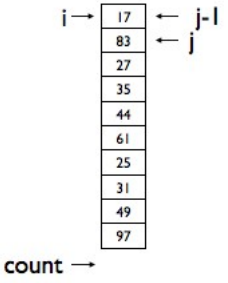
Bubble Sort



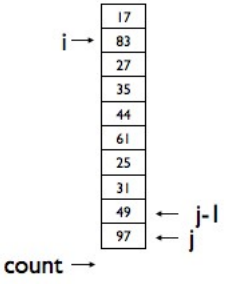
Bubble Sort



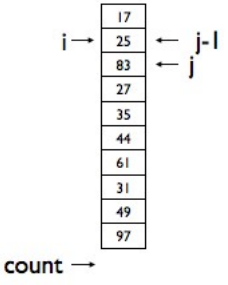
Bubble Sort



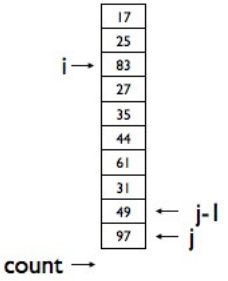
Bubble Sort



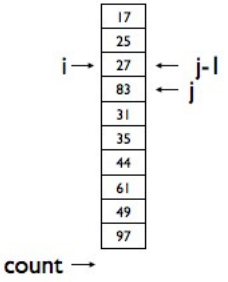
Bubble Sort



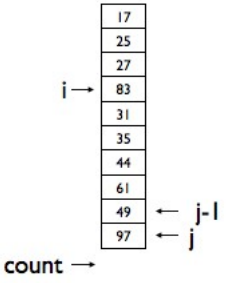
Bubble Sort



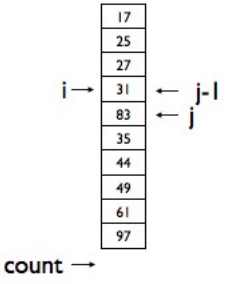
Bubble Sort



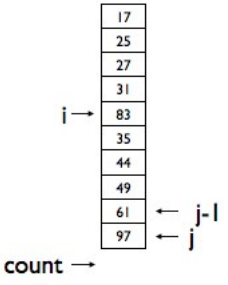
Bubble Sort



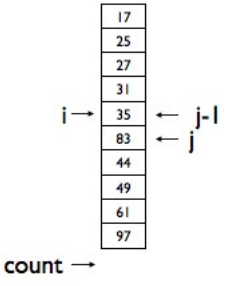
Bubble Sort



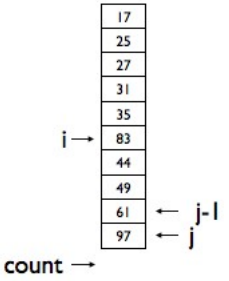
Bubble Sort



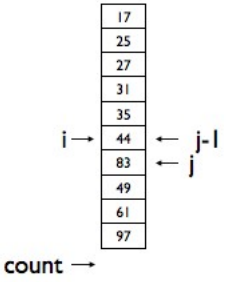
Bubble Sort



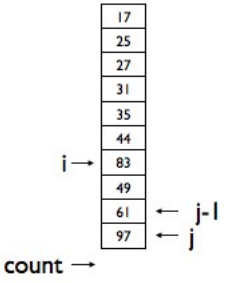
Bubble Sort



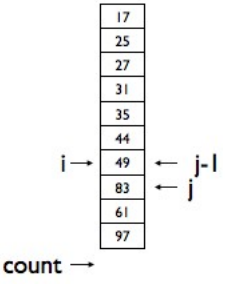
Bubble Sort



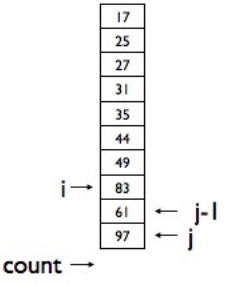
Bubble Sort



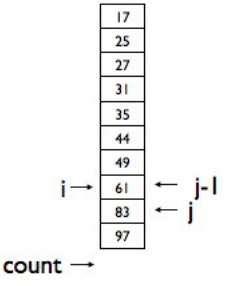
Bubble Sort



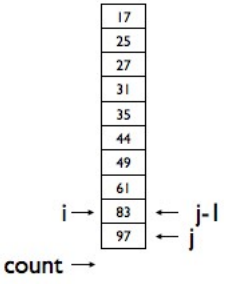
Bubble Sort



Bubble Sort



Bubble Sort




```
public class Array {
.
.
.
    public void bubbleSort () {
      for (int i = 0; i < this.count; i++)
        for (int j = this.count - 1; j > i; j--)
          if (this.val[j].compareTo(this.val[j-1]) < 0)
            this.swap(j,j-1);
    }
.
.
.
}
```