Figure 17.1 A first sorting algorithm.

Suppose we have a list of integers in an array **list**:

i	list[i]	countSmaller(list[i], list)	sortedList[i]
0	7	2	5
1	5	0	6
2	6	1	7
3	9	4	8
4	8	3	9

There are two steps in this sorting algorithm:

1. For each item, count the number of smaller items, and
2. copy each item into the new sorted list, at the position given by **countSmaller**.

Combining these two steps gives our first sorting method:

```
// sort: sorts integers in "list" in increasing order
public static void sort (int[] list) {
    int[] sortedList = new int[list.length];
    // for each item ...
    for (int i = 0; i < list.length; i++) {
        // count the number of items smaller
        int whereInNewList = countSmaller(list[i], list);
        // copy to the new list at the position given by countSmaller
        sortedList[whereInNewList] = list[i];
    }
    // copy the new list back to the parameter "list"
    for (int i = 0; i < list.length; i++)
        list[i] = sortedList[i];
}
```

When does this method fail?

When is it hard to use?