Problem 1: Bubble Sort

Your task is to implement the Bubble sort in Java, as covered in class, in a way that shows intermediate steps during sorting. As an input, your program must read several sequences of integers and on each sequence show how Bubble sort works by printing intermediate arrays after each swap.

There are some particular implementation requirements as follows:
• The Bubble sort must be implemented in a separate method with the following signature:

```
void bubble_sort(int[] arr)
```

• The `main` method should keep reading data and calling the `bubble_sort` method.
• The `bubble_sort` method will call a method named `print_array` with the following signature

```
void print_array(int[] arr)
```

which will print array after each iteration.
• In the introductory comment on your solution, you must include the Bubble sort algorithm in pseudocode. You must use the algorithm as discussed in the class, using a more efficient version where each pass is shorter since we know that some elements at the end are already sorted and maximal.

Input

The input consists of a series of lines with one sequence of integers per line. Each line, except the last, begins with a positive integer, n, denoting the length of the sequence, followed by n integers, separated by whitespace. The last line contains a single integer, 0, denoting the end of the input.

Output

For each sequence the program should print out the steps in Bubble sort in the format shown below.

Sample Input	*Sample Output*
3 3 2 1 3 1 2 3 0	Starting array: 3 2 1 Bubble sort: 2 1 3 1 2 3 1 2 3 Starting array: 1 2 3 Bubble sort: 1 2 3