Problem 4—Parallel Bubble Sort

Oppie has been trying to explain the concept of sorting to the Barbies. And, to be fair, Nuclear Physicist Barbie picked it up just fine, and Computer Programmer Barbie already knew about it, but most of the other Barbies weren't interested at all until Oppie mentioned Bubbles. Parallel Bubble Sort isn't a difficult sort to understand. If you have an array of ints, you swap the elements at the following pairs of positions if they are out of order: <0,1> <2,3> <4,5> <6,7> ... and so forth. (If there are an odd number of elements, the last one doesn't swap with anybody.) Then you swap the following pairs if they are out of order: <1,2> <3,4> <5,6> <7,8> ... and so forth. (Element 0 doesn't swap with anybody. If there an even number of elements, the last one doesn't swap with anybody.) You continue swapping back and forth between these two patterns until the list is sorted in ascending order. For example,

3 2 1
2 3 1 <0,1> were out of order. The 1 in position 2 didn't swap with anybody.
2 1 3 <1,2> were out of order. The 2 in position 1 didn't swap with anybody.
1 2 3 <0,1> were out of order. The 3 in position 2 didn't swap with anybody.
And now we are done since the list is sorted.

You are to emulate parallel bubble sort on a given array of ints.

<u>INPUT SPECIFICATION.</u> Each input case begins with an unsigned decimal integer *l* representing the length of the array, followed by **<EOLN>**. The following line consists of *l* unsigned positive decimal integers consisting of the initial contents of the array. There will be exactly one space between successive integers and the line will be terminated by **<EOLN>**. "0**<EOLN>**" follows the last case. It is not to be processed; it merely signifies the end of input.

<u>OUTPUT SPECIFICATION.</u> The output cases should be processed in the same order as their respective input cases. Each output case should begin with "Case *c*:**<EOLN>**" where *c* is the case number. There should follow a number of lines representing the array at each phase of the sort. The first line should be the original layout of the array. The last line should display the final sorted array. At no time should two successive lines be identical. Each line should be displayed as *l* integers with a space following each integer and the line terminated with **<EOLN>**. An extra **<EOLN>** should follow each case.

<u>SAMPLE INPUT.</u>

```
3<EOLN>
3·2·1<EOLN>
5<EOLN>
3·1·4·1·6<EOLN>
12<EOLN>
1·2·3·4·5·6·7·8·9·10·11·12<EOLN>
0<EOLN>
<EOF>
```

<u>SAMPLE OUTPUT.</u>

```
Case·1:<EOLN>
3·2·1·<EOLN>
2·3·1·<EOLN>
2·1·3·<EOLN>
1·2·3·<EOLN>
<EOLN>
Case·2:<EOLN>
3·1·4·1·6·<EOLN>
1·3·1·4·6·<EOLN>
1·1·3·4·6·<EOLN>
<EOLN>
Case·3:<EOLN>
1·2·3·4·5·6·7·8·9·10·11·12·<EOLN>
<EOLN>
<EOF>
```