



sorting

100 points

Source code: `sorting.c`, `sorting.cpp`, `sorting.pas`

Input files: `sorting.in`

Output files: `sorting.out`

Time limit: **0.3 s**

Memory limit: **64 MB**

Little P has just learned the shell-sort sorting algorithm. He was given some code that sorts an array of N integers in ascending order. Let A be the array to be sorted.

Pascal	C/C++
<pre> 1 gap := X; 2 repeat 3 ok := 1; 4 for i := 1 to N - gap do 5 if A[i] > A[i+gap] then 6 begin temp:=a[i]; 7 A[i]:=A[i+gap]; 8 A[i+gap] := temp; 9 ok := 0 10 end; 11 if gap div 2>1 then gap:=gap div 2 else gap:=1 12 until ok=1; </pre>	<pre> 1 gap = X; 2 do 3 { ok = 1; 4 for (i = 1; i<= N - gap; i++) 5 if (A[i] > A[i+gap]) 6 { temp = A[i]; 7 A[i] = A[i+gap]; 8 A[i+gap] = temp; 9 ok = 0; 10 } 11 if (gap/2 > 1) gap=gap/2; else gap=1; 12 } while (ok == 0); </pre>

where $i, N, X, gap, temp, ok$ are integers (int for C/C++, longint for Pascal).

While typing this code, little P **forgot** to copy line 11.

Task

You are given the array to be sorted, A. A has N distinct elements, all between 1 and N.

You are asked to find all the values X for which the algorithm (without line 11) sorts A. We call these X values to be valid.

Input

The input file `sorting.in` has 2 lines. The first line has one integer, N. The next line describes A: N integers separated by one space.



Output

The output file `sorting.out` should have the number of valid values X on the first line. The second line should have all the valid values X , separated by one space. They should be sorted in ascending order.

Restrictions and remarks

- $1 < N < 500000$
- $1 \leq X \leq N-1$

Example

sorting.in	sorting.out	Explanations
6 4 2 6 1 5 3	2 1 3	N is 6 and A is: 4, 2, 6, 1, 5, 3. Valid values for X are: <ul style="list-style-type: none">• $X = 1$, we swap the numbers on the following positions (1,2), (3,4), (4,5), (5,6), (2,3), (4,5), (1,2), (3,4);• $X = 3$, we swap the numbers on the following positions (1,4), (3,6).