Maximum of minimum (`maxofmin`)

Given in input a vector V of N integers we want to find, for each size between 1 and N, the maximum of the minimum's of every contiguous subsequence in the vector.

Example

For $N = 6$ and $V[6] = [3, 1, 4, 6, 2, 9]$ we have this contiguous subsequences:

- Size 1: [**3**], [**1**], [**4**], [**6**], [**2**] and [**9**].

- Size 2: [3, **1**], [**1**, 4], [**4**, 6], [6, **2**] and [**2**, 9].

- Size 3: [3, **1**, 4], [**1**, 4, 6], [4, 6, **2**] and [6, **2**, 9].

- Size 4: [3, **1**, 4, 6], [**1**, 4, 6, 2] and [4, 6, **2**, 9].

- Size 5: [3, **1**, 4, 6, 2] and [**1**, 4, 6, 2, 9].

- Size 6: [3, **1**, 4, 6, 2, 9].

Where the minimum of each subsequence is bolded.

For each size then the maximum of the minimum's of every contiguous subsequence in the vector is: **9** (for size 1), **4** (for size 2), **2** (for size 3), **2** (for size 4), **1** (for size 5) and **1** (for size 6).

Implementation

You should submit a single file, with either a `.c`, `.cpp`, `.java` or `.py` extension.

Your program must read the input data from `stdin` and write the output data into `stdout`.

`stdin` consists of 2 lines:

- Line 1: The integer N, e.g. the size of the vector V.

- Line 2: N space-separated integers, e.g. the elements of V.

`stdout` consists of only one line:

- Line 1: N space-separated integers: the maximum of the minimum's of every contiguous subsequence in the array, for each size between 1 and N.

No additional output should be printed.

Constraints

- $3 \leq N \leq 100\,000$.
- $1 \leq V[i] \leq 1\,000\,000$ for each $0 \leq i < N$.

Scoring

Your program will be tested on several test cases grouped in subtask.

To achieve the score of a subtask, you need to correctly solve all of its test cases.

- **Subtask 1 [20 points]**: $N \leq 100$.
- **Subtask 2 [20 points]**: $N \leq 1\,000$.
- **Subtask 3 [20 points]**: $N \leq 10\,000$.
- **Subtask 4 [40 points]**: $N \leq 100\,000$.

Examples

stdin	stdout
6 3 1 4 6 2 9	9 4 2 2 1 1