

Destroy the Village (barbarian)

A village is being raided by a barbarian, who, despite being on his own, shouldn't be underestimated. The village consists of N houses, numbered from 0 to $N - 1$ and arranged in a straight line perpendicular to the shore, with house i being D_i meters away from the shore. The barbarian's strategy is simple: he will raze the house he is in, then move to the closest house that hasn't been razed yet, until all the treasures in the village have been hoarded. If there is more than one house at the minimum distance, the barbarian will move to the one closest to the shore first.



Figure 1: The mighty barbarian.

The villagers have a plan though, which is to hide in the last house that the barbarian will raid to face him when he's most tired. Trying to predict the barbarian's moves is giving them a headache, moreover, the raid may start from any house. In order to help them, you have to find, for each possible starting house i , which house H_i would be destroyed last by the barbarian.

Among the attachments of this task you may find a template file `barbarian.*` with a sample incomplete implementation.

Input

The input file consists of:

- a line containing integer N .
- a line containing the N integers D_0, \dots, D_{N-1} .

Output






The output file must contain a single line consisting of the N integers H_0, \dots, H_{N-1} .

Constraints

- $2 \leq N \leq 10^6$.
- $0 \leq D_i \leq 10^{12}$ for each $i = 0 \dots N - 1$.
- $D_i > D_{i-1}$ for each $1 \leq i \leq N - 1$.

Scoring

Your program will be tested against several test cases grouped in subtasks. In order to obtain the score of a subtask, your program needs to correctly solve all of its test cases.

- **Subtask 1** (0 points) Examples.

- **Subtask 2** (12 points) $N \leq 100$, $D_i \leq 10^9$, for each $0 \leq i \leq N - 1$.

- **Subtask 3** (28 points) $N \leq 1000$.

- **Subtask 4** (35 points) $N \leq 100\,000$.

- **Subtask 5** (25 points) No additional limitations.


Examples

input	output
5 1 4 6 7 10	4 0 4 4 0

Explanation

In the **first sample case**

- When the barbarian starts from house 0, located 1 meter away from the shore, he destroys it, then moves to house 1, at a distance of 3 meters from house 0, then to house 2, then 3. The last house standing is number 4.
- When the barbarian starts from house 1, located 4 meter away from the shore, he destroys it, then moves to house 2, at a distance of 2 meters from house 1, then to house 3, then 4. The last house standing is number 0.
- When the barbarian starts from house 2, located 6 meter away from the shore, he destroys it, then moves to house 3, at a distance of 1 meter from house 2, then to house 1. Note that both house 1 and 4 are at a distance of 3 meters from house 3, but house 1 is closer to the shore. He then moves to house 0. The last house standing is number 4.
- When the barbarian starts from house 3, located 7 meter away from the shore, he destroys it, then moves to house 2, at a distance of 1 meter from house 3, then to house 1, then 0. The last house standing is number 4.
- When the barbarian starts from house 4, located 10 meter away from the shore, he destroys it, then moves to house 3, at a distance of 3 meters from house 4, then to house 2, then 1. The last house standing is number 0.