# A. Minimum Difficulty

time limit per test
2 seconds
memory limit per test
256 megabytes
input
standard input
output
standard output

Mike is trying rock climbing but he is awful at it.

There are $n$ holds on the wall, $i$-th hold is at height $a_i$ off the ground. Besides, let the sequence $a_i$ increase, that is, $a_i < a_{i+1}$ for all $i$ from 1 to $n$ - 1; we will call such sequence a *track*. Mike thinks that the track $a_1$, ..., $a_n$ has *difficulty* $d = \max_{1 \le i \le n-1} (a_{i+1} - a_i)$. In other words, difficulty equals the maximum distance between two holds that are adjacent in height.

Today Mike decided to cover the track with holds hanging on heights $a_1$, ..., $a_n$. To make the problem harder, Mike decided to remove one hold, that is, remove one element of the sequence (for example, if we take the sequence $(1, 2, 3, 4, 5)$ and remove the third element from it, we obtain the sequence $(1, 2, 4, 5)$). However, as Mike is awful at climbing, he wants the final difficulty (i.e. the maximum difference of heights between adjacent holds after removing the hold) to be as small as possible among all possible options of removing a hold. The first and last holds **must** stay at their positions.

Help Mike determine the minimum difficulty of the track after removing one hold.

## Input

The first line contains a single integer $n$ ($3 \le n \le 100$) — the number of holds.

The next line contains $n$ space-separated integers $a_i$ ($1 \le a_i \le 1000$), where $a_i$ is the height where the hold number $i$ hangs. The sequence $a_i$ is increasing (i.e. each element except for the first one is strictly larger than the previous one).

## Output

Print a single number — the minimum difficulty of the track after removing a single hold.

## Examples

### input

Copy

```
3
1 4 6
```

### output

```
5
```

### input

Copy

```
5
1 2 3 4 5
```

### output

```
2
```

### input

Copy

```
5
1 2 3 7 8
```

### output

```
4
```

### Note

In the first sample you can remove only the second hold, then the sequence looks like $(1, 6)$, the maximum difference of the neighboring elements equals 5.

In the second test after removing every hold the difficulty equals 2.

In the third test you can obtain sequences $(1, 3, 7, 8)$, $(1, 2, 7, 8)$, $(1, 2, 3, 8)$, for which the difficulty is 4, 5 and 5, respectively. Thus, after removing the second element we obtain the optimal answer — 4.