Problem E: Bounty Challenge

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Trafalgar Law, captain of the heart pirates, is facing a bit of a problem. In front of him there exists **n** strong enemies that he needs to fight. If the value of his current bounty is strictly greater than the strength of his enemy then he can win the fight, otherwise he will choose to run away.
The problem is that, with every fight he wins, his bounty will increase by a 100 and if his bounty exceeds the value **L** then Kizaru, one of the strongest fighters in the marine will be dispatched to get rid of him.

Given the number of fights n, the maximum allowable bounty L and an array of integers *fights* representing the strengh of the opponents, the procedure for simulating the fights is as follows:

1. Start with an initial bounty B

2. For each fight; if the current bounty is strictly greater than the strength of the opponent $fights_i$ $(1<=i<=n)$, add 100 to the current bounty.

Your objective is to find the maximum initial bounty B such that after all the fights, the final bounty is less than or equal to L.

Input :

The first line of the input contains integers n and L $(1<=n<=10^5, 0<=L<=10^{18})$, the number of fights, and the maximum allowable bounty respectively.

The second line of the input contains n space-separated integers $fights_i$, the strength of the *i-th* opponent. $(1<=fights_i <= 10^9)$

Output :

Print a single integer, the maximum initial bounty B such that after all the fights, the final bounty is less than or equal to L.

Example:

Input :

```
5 700
300 200 500 100 700
```

Output :

```
300
```

Input :

```
4 500
700 200 300 100
```

Output :

```
200
```