algoleague

Cenk's home

Problem

Submissions

Discussion Coming Soon

Cenk wants to go to his room but there are N number of monsters located on his way. Each monster has a power and once Cenk faces a monster, Cenk's power reduces the monster's power and the monster's power reduces Cenk's power. If Cenk reaches 0 power while facing the ith monster, his journey ends at time i. If a monster dies, Cenk moves on. Thanks to Cenk's stealth skills, he can choose which monster to face at any time. Determine if Cenk can defeat all the monsters by choosing an optimum order of monsters. If Cenk defeats every monster, print "EV" (without the quotes), otherwise print the maximum time he can go. Cenk faces the first monster at the $1st$ time.

Input Format

In the first line, there are two numbers. The first one is N, the number of monsters. The second one is C, Cenk's power. In the following line, there are N numbers indicating the powers of monsters.

Output Format

In the single line, print the maximum time that Cenk can go.

Constraints

- $1 <= N <= 10^6$
- $1 <= C <= 10^9$
- $1 <=$Power of Each Monster$<= 10^9$

Sample Input 1

```
5 20
4 2 6 3 1
```

Sample Output 1

```
EV
```

Sample Input 2

```
1 9
12
```

Sample Output 2

```
1
```

C++ (GCC 9.2.0) Bright Memory Limit (kB) : 256000 Time Limit (s) : 1

```cpp
#include <iostream>
#include <vector>
#include <algorithm>
using namespace std;

int main() {
    int N;
    long long C, monster_power;
    cin >> N >> C;

    vector<long long> monsters;

    for (int i = 0; i < N; i++) {
        cin >> monster_power;
        monsters.push_back(monster_power);
    }
```

```
17
18      sort(monsters.begin(), monsters.end());
19
20      int time = 0;
21      for(int i = 0; i < N; i++) {
22          if (C <= monsters[i]) {
23              time++;
24              C -= monsters[i];
25              break;
26          }
27          else {
28              C -= monsters[i];
29              time++;
30          }
31      }
```

Upload File

Test against custom test case

Run Code

Submit

Sample Test Case 0

Sample Test Case 1

Accepted

Input(stdin)

```
1  5 20
2  4 2 6 3 1
3
```

Output(stdin)

```
1  EV
2
```

Expected Output

```
1  EV
```