# Falling ball

🔒 locked

Problem Submissions Leaderboard Discussions

You have $N$ balls moving around on a horizontal pole of length $L$ meter(s), each with a constant speed 1 meter / second.
When a balls reaches an end of the pole, it immediatelly falls of it. When two balls meets they turn back and start moving in opposite directions.
You know the original position of balls on the pole, but you do not know the directions in which the balls moving. Your task is to compute the earliest and the latest possible times needed for all balls to fall off the pole.

### Input Format

First line will contains $L$ and $N$.
These two numbers are followed by $N$ integers giving the position of each balls on the pole as the distance measured from the left end of the pole, in no particular order.

### Constraints

$1 \leq L \leq 10^3$
$1 \leq N \leq 10^3$

### Output Format

Output two numbers separated by a single space.
The first number is the earliest possible time when all balls fall off the pole (if the directions of their moves are chosen appropriately) and the second number is the latest possible such time.

### Sample Input

```
10 3
2 6 7
```

### Sample Output

```
4 8
```

### Explanation

.

Submissions: 10
Max Score: 100
Difficulty: Medium

Rate This Challenge:
☆☆☆☆☆

More

C

```c
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>

int main() {

    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
    return 0;
}

```

Line: 1 Col: 1

Upload Code as File

□ Test against custom input

Run Code

Submit Code