

The Croatian delegation, consisting of M people, is travelling to IOI 2013 in Australia¹. They are currently waiting in a queue for check-in at the airport. There are N check-in desks open. Some officials work more efficiently than others, so the desks operate at different speeds. At the k -th desk, T_k seconds are required to finish check-in of a single passenger, and members of our delegation happen to know the exact numbers.

In the beginning, all desks are ready to accept the next passenger, and the delegation members are the only people in the queue. A person can only occupy (start check-in at) an available desk when all people in front of that person in the queue have left the queue (started, not necessarily finished, check-in) already. At that moment, the person can immediately occupy an available desk (if there is one), but can also choose to wait for another (faster) desk to become available. Our delegation members, being computer science geeks, make this decision in such a way that the **moment when all of them have finished check-in is as soon as possible**. Your task is finding that moment in time.

Let us describe the scenario from the first example below. There are two desks, with processing times of 7 and 10 seconds, respectively. Out of the six people in the delegation, the first two immediately occupy the two desks. At time 7, the first desk is freed, and the third person occupies it. At time 10, the fourth person occupies the second desk. At time 14, the fifth person occupies the first desk. At time 20, the second desk is freed again, but the sixth person decides to wait another second (time 21) for the first desk to become available, and then occupy it. This way, the check-in is completed by time 28. If the sixth person hadn't waited for the faster desk, the check-in would have taken a total of 30 seconds.

INPUT

The first line of input contains two positive integers, N ($1 \leq N \leq 100\,000$), the number of desks, and M ($1 \leq M \leq 1\,000\,000\,000$), the number of people in the delegation.

Each of the following N lines contains a number T_k from the problem statement ($1 \leq T_k \leq 10^9$).

OUTPUT

The first and only line of output must contain the required minimum time in seconds.

SCORING

In test data worth a total of 75 points, the number M will be at most 300 000.

¹ Assuming, of course, that the Apocalypse didn't happen.

SAMPLE TESTS

input 2 6 7 10	input 7 10 3 8 3 6 9 2 4
output 28	output 8