CMPUT 274/5 - Tangible Computing **Winter 2020**
Morning Problem: Race

Description

There are n cars lined up for a race. Car i is initially at position i units where i ranges from 0 to $n-1$. This means the car at position $n-1$ is at the front of the line up, the best possible starting position.

Once the race starts, each car i moves at a constant speet of $a[i] \geq 0$ units per second. After $t \geq 1$ seconds, car i is at location $i+t*a[i]$. We say that car i has passed car j if $i < j$, but the location of i is strictly greater than the location of j after t seconds. You want to know how many passes there were. That is, how many pairs $i < j$ are there such that car i passed car j?

Input

The first line of the input will contain two space-separated integers: $0 \leq t \leq 10,000$, the length of the time period in seconds, followed by $0 \leq n \leq 1000$, the number of cars to follow. The second line consists of a list of n nonnegative integers indicating the speeds of the cars. The index of each integer is the initial position of the car in the order, so the i'th such integer is the speed $0 < a[i] \leq 10,000$ of the i'th car.

Output

For each input, output a single line containing a single integer denoting the number of pairs of cars $i < j$ such that car i passed car j.

Sample Input 1

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

Sample Output 1

```
4
```

Explanation: Car 0 has a speed of 3, Car 1 has a speed of 1, Car 2 has a speed of 4, Car 3 as a speed of 1 and Car 4 has a speed of 2. After three seconds, the new order is:

```
Car #              0  1  2   3  4
Old Position:      0  1  2   3  4
New Position:      9  4  14  6  10
```

Car 0 passes Car 1 and Car 3. Car 2 passes Cars 3 and 4, so there are 4 total passes. While car 0 is faster than car 4, it does not pass it in 3 seconds.

Sample Input 2

```
100 2
1 2
```

Sample Output 2

```
0
```

Explanation: There are only two cars. The car at position 0 is slower than the car at position 1, so they will never pass.

Sample Input 3

```
1 2
2 1
```

Sample Output 3

```
0
```

Explanation: At time `t=1` the cars will be at the same position 2, so they have not passed.