HackerLand National Bank has a simple policy for warning clients about possible fraudulent account activity. If the amount spent by a client on a particular day is *greater than or equal to* $2\times$ the client's median spending for a trailing number of days, they send the client a notification about potential fraud. The bank doesn't send the client any notifications until they have at least that trailing number of prior days' transaction data.

Given the number of trailing days d and a client's total daily expenditures for a period of n days, find and print the number of times the client will receive a notification over all n days.

For example, $d = 3$ and $expenditures = [10, 20, 30, 40, 50]$. On the first three days, they just collect spending data. At day 4, we have trailing expenditures of $[10, 20, 30]$. The median is 20 and the day's expenditure is 40. Because $40 \geq 2 \times 20$, there will be a notice. The next day, our trailing expenditures are $[20, 30, 40]$ and the expenditures are 50. This is less than 2×30 so no notice will be sent. Over the period, there was one notice sent.

Note: The median of a list of numbers can be found by arranging all the numbers from smallest to greatest. If there is an odd number of numbers, the middle one is picked. If there is an even number of numbers, median is then defined to be the average of the two middle values. [(Wikipedia)](Wikipedia)

Function Description

Complete the function *activityNotifications* in the editor below. It must return an integer representing the number of client notifications.

activityNotifications has the following parameter(s):

- *expenditure*: an array of integers representing daily expenditures
- *d*: an integer, the lookback days for median spending

Input Format

The first line contains two space-separated integers n and d, the number of days of transaction data, and the number of trailing days' data used to calculate median spending.
The second line contains n space-separated non-negative integers where each integer i denotes $expenditure[i]$.

Constraints

- $1 \leq n \leq 2 \times 10^5$
- $1 \leq d \leq n$
- $0 \leq expenditure[i] \leq 200$

Output Format

Print an integer denoting the total number of times the client receives a notification over a period of n days.

Sample Input 0

```
9 5
2 3 4 2 3 6 8 4 5
```

Sample Output 0

```
2
```

Explanation 0

We must determine the total number of *notifications* the client receives over a period of $n = 9$ days. For the first five days, the customer receives no notifications because the bank has insufficient transaction data: $notifications = 0$.

On the sixth day, the bank has $d = 5$ days of prior transaction data, $\{2, 3, 4, 2, 3\}$, and $median = 3$ dollars. The client spends 6 dollars, which triggers a notification because $6 \geq 2 \times median$: $notifications = 0 + 1 = 1$.

On the seventh day, the bank has $d = 5$ days of prior transaction data, $\{3, 4, 2, 3, 6\}$, and $median = 3$ dollars. The client spends 8 dollars, which triggers a notification because $8 \geq 2 \times median$: $notifications = 1 + 1 = 2$.

On the eighth day, the bank has $d = 5$ days of prior transaction data, $\{4, 2, 3, 6, 8\}$, and $median = 4$ dollars. The client spends 4 dollars, which does not trigger a notification because $4 < 2 \times median$: $notifications = 2$.

On the ninth day, the bank has $d = 5$ days of prior transaction data, $\{2, 3, 6, 8, 4\}$, and a transaction median of 4 dollars. The client spends 5 dollars, which does not trigger a notification because $5 < 2 \times median$: $notifications = 2$.

Sample Input 1

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

Sample Output 1

```
0
```

There are 4 days of data required so the first day a notice might go out is day 5. Our trailing expenditures are $[1, 2, 3, 4]$ with a median of 2.5 The client spends 4 which is less than 2×2.5 so no notification is sent.