

Negotiations

Input File: *standard input*

Output File: *standard output*

Time and Memory Limits: 1 second, 32 MB

In a distant eastern corner of the middle school, beyond the vast sandpits and just over the Salty Puddle, there is a small strip of playground that has been the centre of some commotion. While most of the children get along with each other and take turns with the equipment, Isobelle and Paul have been squabbling over this strip for countless lunchtimes. The feud has become so intense that many other children are now arguing about the dispute, with an increase in threats and “mysterious eggings“ that could possibly lead to an undesirable all-out playground war. As Isobelle’s chief negotiator, you are keen to bring the squabble over the disputed playground strip to an end once and for all, by separating an area for Paul to play and an area for Isobelle to play next to each other in peace.

The strip of playground in question is divided into N indivisible sections. Each section is assigned a play value which is a non-negative integer (for example, some sections have particularly climbable trees while others have good access to the much-contested water fountains). Your job is to propose a *contiguous region* of playground for Isobelle, while Paul will reluctantly concede to accepting the remaining contiguous region (or possibly two remaining disconnected contiguous regions). Note that a *contiguous region* is a set of adjacent sections of the strip: for instance, you could propose a region of the sections 2, 3 and 4, but not a region of the sections 2, 3 and 5.

Since you are Isobelle’s chief negotiator, you wish to maximise the value of her region (the sum of the individual values of the sections in the region). However, any proposal must be accepted by Paul, who will not accept a proposal if it looks like Isobelle is being too greedy. To avoid being seen as greedy, it is critical that rather than merely selecting the highest-value contiguous region (which would be the entire strip), you select the K th highest-value contiguous region for Isobelle. For instance, say the playground looks like so:

Section	1	2	3	4	5	6	7
Play value	9	0	4	5	1	8	1

In the example above, the value of the entire strip is 28. If we had to find the 5th highest-value region, we could select sections [1..5] or sections [2..7] or sections [3..7] which all would give a total of 19. The regions with a higher total value are [1..7] and [1..6]. Note that if we were asked for the 3rd or 4th highest-value region, 19 would still be the correct answer (as there are three regions tied for 3rd place).

You will be given the value of each section of playground in the strip, and your task is to determine the highest value of a contiguous region such that there are at least $K - 1$ other possible contiguous regions with an equal or higher value.

Input

The first two lines of input will contain the integers N and K respectively. The following N lines will each contain a single integer: the i th of these lines represents the value of the i th section of playground strip.

Output

Output should consist of a single integer: the total value of the K th highest-value contiguous region of playground.

Sample Input

```
7
5
9
0
4
5
1
8
1
```

Sample Output

```
19
```

Explanation

The sample input and output correspond to the example above.

Constraints & Subtasks

In all three subtasks, $2 \leq N$ and $2 \leq K \leq \frac{N(N+1)}{2}$. Each section's value will be at least 0.

- For Subtask 1 (20 points), $N \leq 100$ and each section's value will be no greater than 100.
- For Subtask 2 (20 points), $N \leq 3000$ and each section's value will be no greater than 1000.
- For Subtask 3 (60 points), $N \leq 100\,000$ and each section's value will be no greater than 1 000 000 000. Note that K can potentially be a value higher than the upper bound of a 32-bit integer.