Trasporto tronchi (alberi)

The preparations for the building of the OII[1] commemorative monument are starting. To free the area some trees have been cut and now they have to be loaded on a lorry to be transported to the carpentry.

Figure 1: The trees that have been cut down.

Initially the i-th tree is in position A_i and the lorry is in position 0.

Before starting the loading some trees can be pruned: pruning[2] a tree has a cost of K and makes the trunk smooth so that it can roll on other smooth trunks.

The trees can be moved in two ways:

- Any tree can be moved by one position towards the lorry at a cost of 1, given that the position in which it is being moved is empty.
- If immediately to the left of a pruned tree there are one or more consecutive pruned trees followed by an empty position the tree can be rolled on those (but not on non-pruned trees or the ground) until the first empty position at a cost of 1.

The lorry is always to be considered as an empty position.

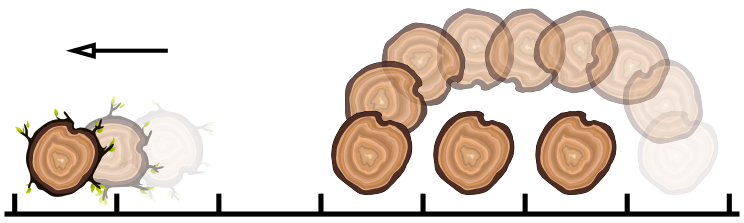


Figure 2: The two ways of moving trees.

Help the organizers determine the minimum cost to load all trees on the lorry!

[1] Italian Olympiads in Informatics

[2] Removing all branches

Implementation

You must submit a single file with the extension `.cpp`.

> ⇨ Among the attachments of this task you will find a template file `alberi.cpp` with a sample implementation.

You will have to implement the following function:

C++	`long long carica(int N, int K, vector<int> A);`

- The integer N represents the number of trees.
- The integer K represents the cost of pruning a tree.
- The array A, indexed from 0 to $N-1$, contains the initial positions of the trees.
- The function must return the minimum cost to bring all trees in position 0..

Sample Grader

Among this task's attachments you will find a simplified version of the grader used during evaluation, which you can use to test your solutions locally. The sample grader reads data from `stdin`, calls the function you have to implement and writes back on `stdout` using the following format.

The input file contains 2 lines:

- Line 1: the integers N e K.
- Line 2: the integers $A_0, ..., A_{N-1}$.

The output file contains one single line with the value returned by the `carica` function.

Constraints

- $1 \leq N \leq 500\,000$
- $0 \leq K \leq 10^9$
- $1 \leq A_i \leq 10^9$ for each $0 \leq i < N$
- $A_i < A_{i+1}$ for each $0 \leq i < N-1$

Scoring

Your program will be tested against several test cases grouped in subtasks. In order to obtain the full score of a subtask, your program needs to correctly solve all of its test cases.

- **Subtask 0 [0 points]**: Examples.
- **Subtask 1 [11 points]**: $K = 10^9$.
- **Subtask 2 [17 points]**: $K = 0$, $N \leq 500$ and $1 \leq A_i \leq 2000$ for each $0 \leq i < N$.
- **Subtask 3 [22 points]**: $K = 0$.
- **Subtask 4 [23 points]**: $K \leq 2000$, $N \leq 500$ and $1 \leq A_i \leq 2000$ for each $0 \leq i < N$.
- **Subtask 5 [27 points]**: No additional limitations.

Examples

stdin	stdout
3 2 1 5 6	11
6 3 1 4 5 10 12 14	30

Explanation

In the **first sample case** it is optimal to prune the trees in positions 5 and 6.

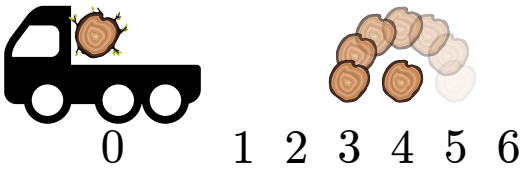
Initial situation

Two trees are pruned

After pruning the two trees 7 moves are enough: the total cost is $7 + 2 \cdot 2 = 11$.

Move 1

Move 2

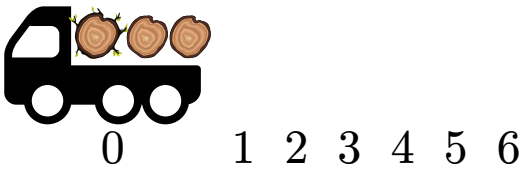


Move 3

Move 4

Move 5

Move 6



Move 7 - final situation

In the **second sample case** it is optimal to prune all trees except for the first.