ETH Zurich
Institut für Theoretische Informatik
Prof. Dr. Angelika Steger

HS 2015
Exercise Sheet 0

ACM Programming Challenges Lab

Exercise 1 – *Jump*

Remember the old Kris-Kross song named "Jump"? If not, listen to it and you'll find out what the jumping is all about.

You are given n cells numbered from 1 up to n, with each cell having a number written in it. Let's denote the number written in the cell i with a_i.

You are also given a number k indicating how far you can jump. In particular, if you're standing on the cell i, you can jump to any cell j for which $|i - j| \leq k$. Finally, not every jump is as easy as the other: jumping from a cell i to the cell j costs a_j CHF.

You are currently on the cell 1 and your goal is to, by a series of consecutive jumps, reach the cell n with a minimum possible cost.

Input The first line contains $t \leq 20$, the number of test cases following. Each test case starts with a line containing integers n and k (for the constraints on the size of these integers please refer to the *points* section below), the number of cells and the jump distance. The second line of a test case contains n integers $a_1, \ldots, a_n$, where $1 \leq a_i \leq 10^9$. (Note: don't forget to set `ios_base::sync_with_stdio(false)`)

Output For each test case output a single integer which denotes the minimum total cost needed in order to get from the cell 1 to the cell n. (Note: you might want to use long long instead of int.)

Sample Input

```
2
7 2
1 2 3 4 5 6 7
10 3
1 2 2 4 5 1 10 1 5 1
```

Sample Output

```
15
5
```

Points There are two groups of test sets. The individual points are specified below; the total number of points is 100.

1. For the first group of test sets (*small*), worth 50 points, you may assume that $n, k \leq 1000$.
2. For the second group of test sets (*large*), worth 50 points, you may assume that $n, k \leq 500000$.

(100 Points)