Huge Grid (Easy Version)

This is the easy version of the problem. Here yout task is to find the minimum possible path sum.

Given a binary sequence A of length N, we construct an $N \times N$ matrix B as follows:

- If $i \leq j$, then $B_{i,j} = \sum_{k=i}^{j} A_k$.
- Otherwise, $B_{i,j} = B_{j,i}$.

Define a path from $(1, 1)$ to (N, N) as a sequence of pairs:

$$P = [(i_1, j_1), (i_2, j_2), \ldots, (i_{2N-1}, j_{2N-1})],$$

where:

- $(i_1, j_1) = (1, 1)$ and $(i_{2N-1}, j_{2N-1}) = (N, N)$.
- $(i_{k+1}, j_{k+1}) = (i_k + 1, j_k)$ or $(i_k, j_k + 1)$ for $1 \leq k < 2N - 1$.
- The path sum is defined as $\sum_{(i,j) \in P} B_{i,j}$.

Your task is to find the **minimum possible path sum**.

Input Format

- The first line of input will contain a single integer T, denoting the number of test cases.
- Each test case consists of multiple lines of input.
  - The first line of each test case contains an integer N.
  - The second line contains a binary string A of length N.

Output Format

For each test case, output on a new line: the **minimum possible path sum**.

Constraints

- $1 \leq T \leq 10^5$
- $1 \leq N \leq 10^6$
- $0 \leq A_i \leq 1$
- The sum of N over all test cases won't exceed 10^6.

Sample 1:

Input	Output
3 3 011 4 0000 1 1	5 0 1

Explanute: