**128MB,1S,.xxx**

栈堆队

**问题描述**

给你n个栈，你要用这n个栈实现1个队列的m次队列push/pop（我们用0表示队列push，1表示队列pop），你已经预先知道整个序列，且仅需要保证这m次pop的结果正确，即不需要维护第m+1次及之后的操作的正确性。请问最少的((栈的(push + pop))次数之和)。

我们可以将这m次操作看作依次在时间段1到时间段m给出，输入仅在该元素执行队列push的时间段可读，输出也必须在执行队列pop的时间段输出。

在每个时间段内，你可以执行以下三种操作任意整数次

push输入的元素到某个栈的栈顶（记作1次push）

pop某个栈的栈顶到输出（记作1次pop）

pop某个栈的栈顶，并将该元素push到某个栈的栈顶（记作1次push+1次pop）

若在队列为空的时侯pop可以看作一个无效操作，什么都不需要执行。

比如你有2个栈，操作序列是000111

你不能

从输入push

pop到输出

从输入push

pop到输出

从输入push

pop到输出

**输入描述**

第一行2个整数n和m

接下来1行，一个不含空格的长度为m的0/1字符串，依次表示队列操作，0表示push，1表示pop

$2 <= n <= 10^9; 0 <= m <= 4 \times 10^6$

**输出描述**

一行一个整数，表示最少的((栈的(push+pop))次数之和)

**样例输入**

2 10

0001110011

**样例输出**

12

**数据范围及提示**