

高精度 (A)

时间限制：3s

空间限制：1024MB

题目描述

你的手里有一个初值为0的数 tot 和一个初值为 "0" 的数字串 cur 。你还有一个操作序列，每个操作由一个字符表示，具体含义如下：

- 若字符为 '0'~'9' 中的一个，则在 cur 的末尾加上这个字符，例如若当前操作的字符为 '3'， cur 为 "0029"，则 cur 会变成 "00293"。然后，把 cur 所对应的值加到 tot 上，例如上述操作的效果就是 $tot = tot + 293$ 。
- 若字符为 '-'，则表示将 cur 的末尾的字符删去，例如若当前 cur 为 "00008"，则 cur 会变成 "0000"。保证在删完之后， cur 不会为空。然后，把 cur 所对应的值加到 tot 上，例如上述操作的效果就是 $tot = tot + 0$ 。

你需要回答，在所有操作进行完之后， tot 的值是多少。

输入格式

第一行包含一个正整数 n ，表示操作序列的长度。

第二行为一个长度为 n 的字符串 s ，表示操作序列。保证 s 仅由 '0'~'9' 和 '-' 构成。

输出格式

仅一行一个整数，表示操作完之后 tot 的值。

你需要注意的是，虽然 cur 可能有前导0，但是你输出的 tot 不能含有前导0。

样例一输入

```
8
0100---5
```

样例一输出

```
127
```

样例一解释

$00 + 001 + 0010 + 00100 + 0010 + 001 + 00 + 005 = 127$

样例二输入

6
2-1818

样例二输出

2020

样例三输入

2
0-

样例三输出

0

限制与约定

对于所有数据，保证 $1 \leq n \leq 5 \times 10^5$ 。

subtask1(5pts) : $n \leq 8$;

subtask2(10pts) : $n \leq 1000$;

subtask3(15pts) : 保证 s 中不存在 '-';

subtask4(15pts) : 保证 s 中 '-' 的个数 ≤ 10 ;

subtask5(35pts) : $n \leq 2 \times 10^5$;

subtask6(19pts) : 无特殊限制;

subtask7(1pt) : 时限0.75s。