

Problem A. calculator

Input file: calculator.in
Output file: caluclator.out
Time limit: 1s
Memory limit: 512MB

现在你手里有一个计算器，上面显示了一个数 S 。这个计算器非常奇怪，它只有两个按钮，分别可以把屏幕上显示的数值加上1或者减去1。并且，如果计算器屏幕上的数变成了负数，那么计算器就会损坏。现在你想要在 K 次操作之内把屏幕上的数字变成 T ，而且不让计算器损坏，求一个有多少种方案。对 $10^9 + 7$ 取模。

两种方案不同当且仅当按钮被按下的序列不同。

Input

一行三个整数 S, T, K 。

Output

一行一个正整数，表示答案。

Example

calculator.in	caluclator.out
0 1 3	3

Explanation

一共有3种可能的操作序列：

+1, +1, -1

+1, -1, +1

+1

Constraints

对于30%的数据， $S, T, K \leq 10$ 。

对于60%的数据， $S, T, K \leq 1000$ 。

对于100%的数据， $0 \leq S, T, K \leq 100000$ 。

Problem B. work

Input file: work.in
Output file: work.out
Time limit: 1s
Memory limit: 512MB

有一个工厂一共有 n 台排成一排的机器，其中第 i 台机器的工作效率是 e_i 。

机器有开或者关两种状态，显然当所有机器都开着的时候工作效率可以达到最大。但是由于工厂的供电系统出现了故障，不能够同时开启任意连续 $k + 1$ 台机器，否则工厂就会爆炸。

求工厂在不发生爆炸的前提下能够达到的最大效率。

Input

第一行两个正整数 n 和 k 。

接下来 n 行，每一行一个正整数，代表 e_i 。

Output

一行一个数表示答案。

Example

work.in	work.out
5 2 1 2 3 4 5	12

Explanation

最优答案显然是同时开启1,2,4,5号机器。

Constraints

对于30%的数据， $n \leq 100$ 。

对于100%的数据， $n \leq 100000, 1 \leq k, 0 \leq e_i \leq 10^9$ 。

Problem C. cubes

Input file: cubes.in
Output file: cubes.out
Time limit: 1s
Memory limit: 512MB

现在有一排 n ($n \leq 30000$) 个积木，按顺序从1到 n 编号，我们想把这些积木堆起来。

刚开始的时候， n 个积木每一个是单独的柱子，接下来我们会进行 m 次操作，把这些积木堆成更高的柱子。

具体来说，操作的类型有两种：

- 将 x 号积木所在的柱子按照原顺序堆到 y 号积木所在的柱子上面。
- 计算 x 号积木所在的柱子当中，堆在 x 积木之下的积木个数。

请你帮忙完成这两种操作。

Input

第一行一个正整数 m ，注意我们并不会给出 n 的值。

接下来 m 行，每行描述一个操作，如果这一行以'M'开头，那么接下来两个正整数 x, y ，表示进行一次操作1；如果以'C'开头，接下来一个正整数 x ，表示进行一次操作2。

Output

输出若干行，表示每个操作2的答案。

Example

cubes.in	cubes.out
6	1
M 1 6	0
C 1	2
M 2 4	
M 2 6	
C 3	
C 4	

Constraints

对于40%的数据， $m \leq 200$ 。

对于100%的数据， $m \leq 100000$ 。