

100%的数据中， $1 \leq N, M \leq 200, K \leq 200, T \leq 40000$ 。

维护数列

【问题描述】

请写一个程序，要求维护一个数列，支持以下 6 种操作：（请注意，格式栏中的下划线‘_’表示实际输入文件中的空格）

操作编号	输入文件中的格式	说明
1. 插入	INSERT_posi_tot_c1_c2..._ctot	在当前数列的第 $posi$ 个数字后插入 tot 个数字： c_1, c_2, \dots, c_{tot} ；若在数列首插入，则 $posi$ 为 0
2. 删除	DELETE_posi_tot	从当前数列的第 $posi$ 个数字开始连续删除 tot 个数字
3. 修改	MAKE-SAME_posi_tot_c	将当前数列的第 $posi$ 个数字开始的连续 tot 个数字统一修改为 c
4. 翻转	REVERSE_posi_tot	取出从当前数列的第 $posi$ 个数字开始的 tot 个数字，翻转后放入原来的位置
5. 求和	GET-SUM_posi_tot	计算从当前数列开始的第 $posi$ 个数字开始的 tot 个数字的和并输出
6. 求和最大的子列	MAX-SUM	求出当前数列中和最大的一段子列，并输出最大和

【输入格式】

输入文件的第 1 行包含两个数 N 和 M ， N 表示初始时数列中数的个数， M 表示要进行的操作数目。

第 2 行包含 N 个数字，描述初始时的数列。

以下 M 行，每行一条命令，格式参见问题描述中的表格。

【输出格式】

对于输入数据中的 GET-SUM 和 MAX-SUM 操作，向输出文件依次打印结果，每个答案（数字）占一行。

【输入样例】

```
9 8
2 -6 3 5 1 -5 -3 6 3
```

```

GET-SUM 5 4
MAX-SUM
INSERT 8 3 -5 7 2
DELETE 12 1
MAKE-SAME 3 3 2
REVERSE 3 6
GET-SUM 5 4
MAX-SUM
    
```

【输出样例】

```

-1
10
1
10
    
```

【样例说明】

初始时，我们拥有数列

2 -6 3 5 1 -5 -3 6 3

执行操作 GET-SUM 5 4, 表示求出数列中从第 5 个数开始连续 4 个数字之和, 如下图中的灰色部分 $1+(-5)+(-3)+6 = -1$:

2 -6 3 5 1 -5 -3 6 3

执行操作 MAX-SUM, 表示要求求出当前数列中最大的一段和, 即如下图所示, 应为 $3+5+1+(-5)+(-3)+6+3 = 10$:

2 -6 3 5 1 -5 -3 6 3

执行操作 INSERT 8 3 -5 7 2, 即在数列中第 8 个数字后插入 -5 7 2, 如下所示的灰色部分:

2 -6 3 5 1 -5 -3 6 -5 7 2 3

执行操作 DELETE 12 1, 表示删除第 12 个数字, 即最后一个:

2 -6 3 5 1 -5 -3 6 -5 7 2

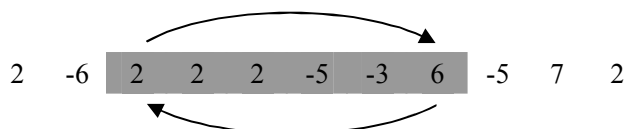
执行操作 MAKE-SAME 3 3 2, 表示从第 3 个数开始的 3 个数字, 即下图中的灰色部分, 统一修改为 2:

2 -6 3 5 1 -5 -3 6 -5 7 2

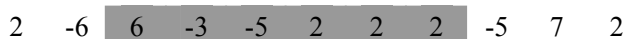
改为

2 -6 2 2 2 -5 -3 6 -5 7 2

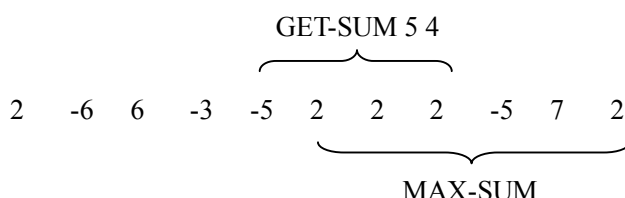
执行操作 REVERSE 3 6, 表示取出数列中从第 3 个数开始的连续 6 个数:



如上所示的灰色部分 2 2 2 -5 -3 6, 翻转后得到 6 -3 -5 2 2 2, 并放回原来位置:



最后执行 GET-SUM 5 4 和 MAX-SUM, 不难得到答案 1 和 10。



【评分方法】

本题设有部分分, 对于每一个测试点:

如果你的程序能在输出文件正确的位置上打印 GET-SUM 操作的答案, 你可以得到该测试点 60% 的分数;

如果你的程序能在输出文件正确的位置上打印 MAX-SUM 操作的答案, 你可以得到该测试点 40% 的分数;

以上两条的分数可以叠加, 即如果你的程序正确输出所有 GET-SUM 和 MAX-SUM 操作的答案, 你可以得到该测试点 100% 的分数。

请注意: 如果你的程序只能正确处理某一种操作, 请确定在输出文件正确的位置上打印结果, 即必须为另一种操作留下对应的行, 否则我们不保证可以正确评分。

【数据规模和约定】

你可以认为在任何时刻, 数列中至少有 1 个数。

输入数据一定是正确的, 即指定位置的数在数列中一定存在。

50% 的数据中, 任何时刻数列中最多含有 30 000 个数;

100% 的数据中, 任何时刻数列中最多含有 500 000 个数。

100% 的数据中, 任何时刻数列中任何一个数字均在 $[-1\ 000, 1\ 000]$ 内。

100% 的数据中, $M \leq 20\ 000$, 插入的数字总数不超过 4 000 000 个, 输入文件大小不超过 20MBytes。

智慧珠游戏