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

糖果括号串

**问题描述**

NYN的菓子工坊上新了，——糖果括号串。

糖果括号串是一个由左括号，右括号和糖果组成的序列，三者均拥有一个非负整数的甜蜜值。当然，空串以及只包含其中一种或两种的序列也是糖果括号串。

我们定义子串为序列上连续的一段按原顺序构成的新序列，特别地，空串和原串本身也是原串的子串。

我们定义串的甜蜜值为其所有元素的甜蜜值之和。

我们用0表示左括号，1表示右括号，2表示糖果，仅考虑类型忽略甜蜜值。

我们用##表示连接，$ \empty$表示空串则以下规则定义了可匹配的串

$ \empty$ 是 可匹配的串

$ 2 $ 是 可匹配的串

$ 0 $ ## 可匹配的串 ## $ 1$ 是 可匹配的串

可匹配的串 ## 可匹配的串 是 可匹配的串

上述四条均不满足的串不是可匹配的串。

我们首先给出一个长度为N的糖果括号串，然后进行M次添加操作，每次操作由1行3个整数构成，依次命名为 position, type, value,

position：0表示在左侧插入，1表示在右侧插入，

type： 0表示左括号，1表示右括号，2表示糖果。

value：该元素的甜蜜值

你需要对初始串和每次添加一个元素后的串，输出甜蜜值最大的可匹配的子串的甜蜜值。

每次的修改是以之前串为基础的，而非每次是原始串插入一个元素。

**输入描述**

第一行两个整数N，M

第二行N个整数表示初始串的type，意义与添加时的一致

第三行N个整数依次表示初始串的value，意义与添加时的一致。

接下来M行，每行三个用空格隔开的整数，position,type,value意义见题目描述

$ 0 <= N,M <= 10^6 $

$ position = \{0,1\}, type= \{0,1,2\}, 0 <= value <= 4\times 10^9$

**输出描述**

共M+1行，第一行表示初始串的答案，

接下来M行，每行一个整数，表示每次添加后的答案。

**样例输入**

5 1

0 0 1 1 2

1 1 1 1 1

0 2 10

5 1

0 0 1 1 2

1 1 1 1 1

0 0 10

**样例输出**

5

15

5

5

**数据范围及提示**

若N==0,则输入的第二行和第三行为空行

请使用较快的输入输出方式。

你可以考虑某个在上学期出了多次且比较折磨的数据结构，当然也可以使用一些比较快速的STL。