

Some contestants have gathered to play the game „Resistance“ before an upcoming contest. This time they decided to change the rules. Again we have „good“ and „bad“ guys but their count is not fixed. For every player two values are known: how much he will contribute to the team of the good guys and how much he will contribute to the team of the bad guys. Furthermore, for some pairs of people it is known the value of their friendship. If two friends are in different teams their friendship is „broken“ for that game. An interesting fact is that for each group which contains at least one but not all people there is at least one friendship between a member of the group and a person outside the group.

In addition, a new rule is that the people distribute to the two teams before the beginning of the game. Of course, all of the contestants decided that **Deni** would make the distribution. She wants to make the distribution with maximum value. The value of a distribution is calculated when we add how much every player contributes to the team, he joins, and we subtract the values of the „broken“ friendships. **Deni** needs your help. Write a program **resistance** that finds the value of the maximal distribution.

But that's not all folks. After some time, some of the people in the group leave. At some other moment in time, some of the people, who have left, return and so on. This means that for every new game a new distribution has to be calculated. The described changes are the following. In the beginning, all **N** contestants in the group are present. After that, there are these changes: change from type 2 is for the leaving of some player and a change from type 1 is for the returning of some player, not present at the moment. Change from type 3 is for the returning of all players, not present at the moment and the change from type 4 is for the leaving of players with numbers from 1 to $\lfloor N/5 \rfloor$ inclusive (this is the whole part of the division of **N** by 5). Now the problem for **Deni** becomes much harder.

Input

From the first line of the standard input read two positive numbers **N** and **M** – the number of contestants and the number of known friendships. From the second line of the standard input read **N** numbers – how much every player contributes to the team of the good guys (the first number is for the first player, the second number – for the second player and so on). From the third line of the standard input read **N** numbers – how much every player contributes to the team of the bad guys (the first number is for the first player, the second number – for the second player and so on). From the next **M** lines read three numbers – x, y and t , which show that the value of the friendship between contestants with numbers x and y is t units (the contestants are numbered with numbers from 1 to **N**). From the next line read the number **Q** – the number of changes. From the last **Q** lines read the changes. If the change is from type 3 or 4, there will be only one number at the line – 3 or 4 respectively. When the change is from type 1 or 2, there will be two numbers – $type(= 1 \text{ or } 2)$ and x , which is the player number.

Output

On the first line output the value of the maximal distribution when all **N** players are in the game. For every change from type 1 and 2 for the current players after the change output on a separate line the value of the maximal distribution.

Constraints

- ♣ $2 \leq N \leq 10^3$
- ♣ $1 \leq M \leq 10^5$
- ♣ $0 \leq Q \leq 1.5 \cdot 10^3$
- ♣ All contributions for the teams and all values for the friendships are whole numbers from 0 to 1000.

Subtasks

Subtask	Points	N	M	Q	Further constraints
1	10	≤ 10	≤ 45	$\leq 10^2$	There are no further constraints.
2	35	$\leq 10^3$	$\leq 10^5$	0	There are no further constraints.
3	10	≤ 500	$\leq 10^4$	$\leq 1.5 \cdot 10^3$	There are no changes from the first type. The changes from type 3 are no more than 10.
4	45	≤ 500	$\leq 10^4$	$\leq 1.5 \cdot 10^3$	There are no further constraints.

Points for each subtask will be received only if the program solves correctly all test cases given for that particular subtask.

Example

Input	Output	Explanation of the example
5 4	100	When all players are present, the maximal distribution between the teams is when the third player is in the team of the good guys and the others are in the other team. The value for the game with this distribution is $10+14+22+25+31-2=100$ (2 is subtracted because players with numbers 1 and 3 are in different teams). After the change from type 3 all players are present and after the next change – from type 4 the players with numbers from 1 to $\lfloor N/5 \rfloor$ leave the game but in our case – only the player with number 1.
10 15 22 20 31	69	
10 14 10 25 31	47	
1 4 10	69	
2 4 10	61	
1 3 2	61	
4 5 10		
7		
2 5		
2 4		
1 4		
2 1		
3		
4		
2 5		