

A. 芽芽傳說

Description

「芽芽傳說」是一款大型多人線上角色扮演遊戲，而鴨鴨是其中的一位玩家。有一天，鴨鴨想賺一些金幣，於是決定去完成一些傳說任務來賺錢。

遊戲中共有 N 個傳說任務，每個傳說任務都需消耗一個傳說盟約才能解鎖。第 i 個傳說任務有前置任務 p_i ， $p_i = -1$ 表示第 i 個傳說任務不須前置任務，否則需要先完成前置任務才能完成第 i 個傳說任務。完成後，鴨鴨可以獲得相應的收益 w_i 。

要獲得傳說盟約需向傳說商人購買，商人總共販售 K 個傳說盟約。這些傳說盟約會依序上架，商店初始僅開放第一個傳說盟約供購買，當該傳說盟約被購買後，才會上架下一個，以此類推。第 i 個傳說盟約的價格是 c_i 。

鴨鴨的目標是透過妥善規劃購買的盟約與挑選要完成的任務，來最大化最終的總收益。總收益定義為：完成任務所得的收益總和減去購買傳說盟約所需的總花費。

請幫助鴨鴨找出最佳策略，使他能獲得最大的總收益。鴨鴨很強，不需要考慮他是否能完成任務與完成任務消耗的時間。

Input

輸入的第一行是兩個正整數 N, K ，代表總共的傳說任務數量與商店販賣的傳說盟約數量。輸入的第二行是 N 個整數 w_i ，代表每個傳說任務的收益。輸入的第三行是 N 個整數 p_i ，代表每個傳說任務的前置任務， -1 表示不須前置任務。輸入的第四行是 K 個整數 c_i ，代表商店中每個傳說盟約的價格，依照上架順序排列。

- $1 \leq N \leq 100000$
- $1 \leq K \leq 5000$
- $1 \leq w_i \leq 10^9$
- $1 \leq p_i \leq N$ or $p_i = -1$
- $0 \leq c_i \leq 10^9$
- 如果有 N 個傳說盟約，則保證可以完成所有任務。

Output

輸出一個整數表示最大總收益。

Sample 1

Input	Output
-------	--------

5 3 2 4 6 8 10 -1 -1 -1 -1 -1 0 0 0	24
--	----

Sample 2

Input	Output
5 5 2 4 6 8 10 -1 -1 -1 -1 -1 3 2 7 1 20	15

Sample 3

Input	Output
5 3 2 4 6 8 10 -1 1 1 -1 4 0 0 0	20

Sample 4

Input	Output
5 3 2 4 6 8 10 2 5 2 5 -1 3 2 1	16

配分

在一個子任務的「測試資料範圍」的敘述中，如果存在沒有提到範圍的變數，則此變數的範圍為 Input 所描述的範圍。

子任務編號	子任務配分	測試資料範圍
1	0%	範例測試資料
2	5%	$p_i = -1, c_i = 0$
3	5%	$p_i = -1$
4	10%	$N, K \leq 100, c_i = 0$
5	20%	$N, K \leq 100$
6	30%	$N \leq 5000$
7	30%	無特別限制

Hint 1

本題測試資料量大，建議使用 `scanf` 進行輸入。若使用 `std::cin` 輸入，請在 `main` 函式第一行加上 `ios_base::sync_with_stdio(0); cin.tie(0);`，且請勿跟 `scanf` 混用，以免造成 `Time Limit Exceeded`。