

След 150 години работа фарът до гр. Шабла спря да работи и лампите угаснаха. Затова Шабленци започнаха да търсят кой може да им го оправи. Наеха майстори, наеха инженери, наеха и вас. Задачата ви е да кажете колко най-много лампи могат да бъдат светнати.

Фарът представлява n на брой лампи, номерирани с числата от 1 до n , свързани с $n-1$ връзки. Всяка връзка, свързваща лампа i с лампа j , свързва и лампа j с лампа i . Освен това от всяка лампа „ходейки“ по връзките, може да се стигне до всяка друга лампа.

Понеже фарът е стар и кабелите не са свързани много добре, всяка лампа i има коефициент b_i , означаващ, че ако лампа i е светната, то най-много b_i лампи свързани с нея, могат да бъдат светнати. (Иначе фарът пак ще изгасне.)

Напишете програма, която по дадени n , списъкът с връзките и коефициентите b_i за всяка лампа, намира колко най-много лампи могат да бъдат светнати, спазвайки се правилата на коефициентите.

Вход

От първия ред на файла **lighthouse.in** се въвежда n - брой лампи на фара.

На следващия ред се въвеждат n числа : $b_1 b_2 \dots b_n$, показващи коефициентите на лампите с номера в реда на въвеждане.

На следващите $n-1$ реда се въвеждат по две числа – $i j$, означаващи, че лампа с номер i е свързана с лампа номер j . (И обратното)

Изход

На единствения ред във файла **lighthouse.out** отпечатайте 1 число – максималния брой лампи, които могат да бъдат светнати.

Ограничения

$$2 \leq n \leq 10^5$$

$$0 \leq b_i \leq n$$

Ограничение по време: 0.4 sec.

Ограничение по памет: 256 MB.

Примерен тест

Вход (lighthouse.in)	Изход (lighthouse.out)
8	7
1 1 1 2 2 1 1 1	
2 4	
2 3	
8 4	
3 5	
5 6	
2 1	
7 4	

Обяснение на примера

Ако светнем всяка лампа без номер 2, условията на коефициентите са изпълнени. Това е максималният брой, защото ако се светнат всички, се нарушава коефициентът за лампа номер 4.