

Наблюдатели на птици

Ограничение по време: 10 s Ограничение по памет: 512 MiB

Обществото на наблюдателите на птици на Сан Сериф има любопитно раздута и постоянно променяща се вътрешна структура. Обществото се състои от n клона, като всеки член принадлежи на точно един клон. Клоновете са номерирани с числата от 1 до n , като i -тият клон има m_i членове. Така обществото се състои от общо $M = m_1 + m_2 + \dots + m_n$ членове.

Всеки клон се ръководи от точно един от членовете си, който се нарича *секретар* на клона. Секретарите са номерирани точно както и съответните клонове, така че (за всяко $i = 1, \dots, n$) секретар i ръководи клон i .

Освен това секретарите са организирани в йерархична структура чрез система от менторства: всеки секретар без един има *ментор*, който е секретар на някой клон. Единственият секретар без ментор е *президентът* на обществото. Ако секретар a е ментор на секретар b , считаме, че секретар b е *последовател* на секретар a . Няма секретар, който е пряко или косвено ментор на себе си. Това означава, че като следваме поредицата от секретар към техния ментор, менторът на този ментор и т.н., накрая винаги достигахме президента.

Дефинираме *влиянието* на секретар като сума от броя членове на неговия клон и влиянието на неговите последователи (ако има такива). Тогава секретарят с най-голямо влияние е президентът, чието влияние е винаги равно на M . Секретар се нарича *старши секретар*, ако влиянието му е $\geq M/2$.

Уставът на обществото постановява, че старшият секретар с най-ниско влияние (сред всички старши секретари) изпълнява ролята на *ковчежник* на обществото.

От време на време секретар (освен президента) може да *смени своя ментор* и вече да стане последовател на различен ментор спрямо преди (при условие, че новият му ментор не е бил негов последовател или последовател на негов последовател и т.н.). Заради това може да се промени влиянието на някои секретари и ролята на ковчежника може да се поеме от друг секретар спрямо преди.

Задача

Напишете програма, която чете описанието на началното състояние на обществото и поредица от смени на ментори. Програмата трябва да отпечата кой е ковчежникът в началното състояние на обществото, както и след всяка смяна на ментор.

Вход

Първият ред съдържа две цели числа n и q , отделени с един интервал; n е броят на клоновете, q е броят на смените на ментори.

Следващите n реда описват началното състояние на обществото. i -тият от тези редове съдържа две цели числа, s_i и m_i , отделени с един интервал; s_i е менторът на секретар i (т.е. на секретарят, който ръководи клон i), докато m_i е броят на членовете на клон i . Стойността $s_i = 0$ обозначава, че секретар i е президент на обществото и затова няма ментор.

Оставащите q реда описват смените на ментори. j -тият от тези редове съдържа две цели числа, \hat{x}_j и \hat{z}_j , отделени с един интервал. Значението на тези числа е

следното. Нека означим с t_j (за $j = 0, \dots, q$) ковчежникът след първите j смени на ментори (така t_0 е ковчежникът в началото преди първата смяна на ментор). Тогава j -тата смяна на ментор е, че секретар z_j става новият ментор на секретар x_j , където $x_j = 1 + ((t_{j-1} + \hat{x}_j) \bmod n)$ и $z_j = 1 + ((t_{j-1} + \hat{z}_j) \bmod n)$. Целта на това представяне на стойностите на x_j и z_j е, за да е задължително Вашата програма да обработва смените на ментори в реда, в който се появяват.

Смените на ментори във входните данни ще бъдат винаги валидни, т.е. z_j няма да е равно на x_j , както и z_j няма да е последовател на x_j , няма да е последовател на последовател на x_j и т.н. Възможно е z_j вече да е бил ментор на x_j точно преди j -тата промяна (така че да не се случва реална промяна).

Обърнете внимание, че ако Вашата програма намери грешно t_j по някое време, то ще декодира грешно поредицата от входове $\hat{x}_{j+1}, \hat{z}_{j+1}$ и т.н., и може да се получи резултат RTE (грешка по време на изпълнение) вместо WA (грешен отговор), защото е възможно грешно декодираните входове да са невалидни (т.е. грешно да се получи z_{j+1} , който е последовател на x_{j+1}).

Ограничения

- $1 \leq n \leq 1\,000\,000$
- $1 \leq q \leq 30\,000$
- $1 \leq m_i$ за всяко $i = 1, \dots, n$
- $m_1 + m_2 + \dots + m_n \leq 10^9$
- $1 \leq \hat{x}_j \leq n$ и $1 \leq \hat{z}_j \leq n$ за всяко $j = 1, \dots, q$.

Подзадачи

- Подзадача 1 (15 точки): $n \leq 100$
- Подзадача 2 (10 точки): $n \leq 1000$
- Подзадача 3 (50 точки): $n \leq 300\,000$
- Подзадача 4 (25 точки): Няма допълнителни ограничения.

Изход

Отпечатайте числата t_0, t_1, \dots, t_q , всяко на отделен ред, където t_j е ковчежникът след първите j смени на ментор. Разбира се, всяко t_j трябва да е цяло число в интервала $1 \leq t_j \leq n$.

Пример

Вход	Изход
7 2	2
0 1	2
1 3	3
1 3	
2 3	
2 1	
5 2	
5 1	
3 7	
2 7	

Коментар

В началото, секретар 2 е ковчезник (така че $t_0 = 2$). При първата смяна на ментор, прочитаем $\hat{x}_1 = 3$ и $\hat{z}_1 = 7$ и пресмятаме $x_1 = 1 + ((2 + 3) \bmod 7) = 6$ и $z_1 = 1 + ((2 + 7) \bmod 7) = 3$; така че секретар 3 става новият ментор на секретар 6; секретар 2 остава ковчезник (така че $t_1 = 2$). При втората смяна на ментор, прочитаем $\hat{x}_2 = 2$ и $\hat{z}_2 = 7$ и пресмятаме $x_2 = 1 + ((2 + 2) \bmod 7) = 5$ и $z_2 = 1 + ((2 + 7) \bmod 7) = 3$; така че секретар 3 става новият ментор на секретар 5 и освен това става новият ковчезник (така че $t_2 = 3$).