**Задача B1. Дърво?**

**Задача В2. Балони**

Дадена е редица от **N** цветни балони. Ще наричаме *цветност* на редицата броя на различните последователности с един и същ цвят. Например, ако цветовете ги означим с естествени числа и редица от 10 балона е с цветове {2, 3, 3, 1, 1, 4, 5, 1, 1, 3}, то нейната цветност ще бъде 7: {2}, {3,3}, {1,1}, {4}, {5}, {1,1}, {3}. Може да извършваме следните три операции: премахвана на балон, вмъкваме на балон на позиция и замяна на балон. Напишете програма **baloni**, която по дадена първоначална редица от балони и операции върху нея, отговаря периодично на въпроса колко е цветността на текущата редица.

**Вход**

На първия ред е числото **N**, на следващия ред са числата A1, A2, …, AN, където Ai е цветът на i-я балон. На третия ред е числото **Т** – общият брой на операциите и запитванията. Всеки от следващите **T** реда е в един от следните видове:

* 1 K – числото K е номерът на балона, който се премахва от редицата
* 2 P C – числата P и C означават, че на позиция Р се вмъква нов балон с цвят С
* 3 Q S – балонът на позиция Q се заменя с друг, който има цвят S
* 4 – запитване за цветността на цялата редица

**Изход**

По реда от входа, за всяко запитване (редовете, започващи с 4) се извежда цветността на редицата.

**Ограничения**: 1 < **N** ≤ 105, 1 ≤ **T** ≤ 105, 1 ≤ Ai, C, S ≤104, 1 ≤ K, P, Q ≤ N’, където N’ е текущият брой балони за съответното запитване.

Подзадача 1: само операции 3 и 4 – 20% от тестовете

Подзадача 2: само операции 1, 3 и 4 – 20% от тестовете

Подзадача 3: ограниченията от условието

|  |  |  |
| --- | --- | --- |
| **Пример**  **Вход**  10  2 3 3 1 1 4 5 1 1 3  8  1 6  3 6 1  4  1 1  2 2 1  4  2 9 2  4 | **Изход**  4  5  6 | ***Пояснение на примера:***  След операцията **1 6**, редицата е  2, 3, 3, 1, 1, 5, 1, 1, 3  След **3 6 1**: 2, 3, 3, 1, 1, 1, 1, 1, 3  На запиването **4** отговорът е 4  След **1 1**: 3, 3, 1, 1, 1, 1, 1, 3  След **2 2 1**: 3, 1, 3, 1, 1, 1, 1, 1, 3  На запиването **4** отговорът е 5  След **2 9 2** 3, 1, 3, 1, 1, 1, 1, 1, 2, 3  На последното запитване **4** отговорът е 6. |

**Задача B3. Монополи**

Всички знаят българската фирма, която продава популярната игра „Дени Монопол“. Ръководството решава, че ще търси ново лого, където ще изобрази всички служители. *За целта им трябват такава тяхна наредба, че да няма служител, който да е разположен след някой от неговите непосредствени подчинени*. Поради големината на фирмата се оказва, че съществуват много подредби и първо трябва да се види колко са те. Тази важна задача възлагат на **Д**ени – най-добрата програмистка на фирмата. За съжаление тя е затрупана от много работа напоследък и няма време. **Д**ени се обръща с молба към Вас да напишете програма **monopoly**, която да изпълни възложената ѝ задача.

В резултат на големия брой реорганизации и прилагането на най-съвременни методи на управление, фирмата има, меко казано, странна организационна структура. В тази структура има две нормални неща:

- Всеки служител без обикновените работници има известен брой подчинени служители.

- Няма служител, при който някой от подчинените му служители (не непременно непосредствено) да му се окаже началник (не непременно непосредствен).

Следват странните резултати от реорганизациите:

- Един служител може да има повече от един непосредствен началник.

- Ако за даден служител са непосредствените му началници, то съществува наредба на , така че ако ги подредим в реда , ще е началник (не задължително непосредствен) на , от своя страна ще е началник (не задължително непосредствен) на и …, ще е началник на .

**Д**ени Ви дава структурата на фирмата от ***N***-те служители и ***M***-те връзки между тях и иска търсената бройка подредби на служителите. Служителите са номерирани с числата от 1 до ***N***. Понеже това може да е много голямо число, **Д**ени ще се задоволи само с остатъка му по модул 109+7.

**Вход**

От първия ред на стандартния вход се въвеждат две цели положителни числа ***N*** и ***M*** – броят служители и броят връзки между тях. От следващите ***M*** реда се въвеждат по две цели числа *x* и *y*, които показват, че служителят с номер *x* е непосредствен началник на служителя с номер *y* (съответно *y* е непосредствен подчинен на *х*).

**Изход**

Едно единствено число – остатъка при деление с 109+7 на намерения брой подредби на служителите.

**Ограничения**

* 1 ≤ ***N***≤100000
* 1 ≤ ***M*** ≤200000

**Подзадачи**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Подзадача** | **Точки** | ***N*** | ***M*** | **Други ограничения** |
| 1 | 15 | ≤ 10 | ≤ 45 | Няма допълнителни ограничения. |
| 2 | 35 | *≤* 19 | ≤ 171 | Няма допълнителни ограничения. |
| 3 | 20 | *≤* 100 | ≤ 200 | Реалният отговор е до 2.105. |
| 4 | 30 | *≤* 100000 | ≤ 200000 | Няма допълнителни ограничения. |

*Точките за дадена подзадача се получават, когато преминат успешно всички тестове за нея.*

**Примери**

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснение на примера** |
| 6 7  1 2 1 3  2 3  3 4  2 4  3 5  2 6 | 8 | Служителите, които имат повече от 1 непосредствен началник са с номера 3 и 4. Непосредствените началници на 3 са 2 и 1 и ако ги подредим в ред 1, 2 тогава 1 е началник на 2. Непосредствените началници на 4 са 3 и 2 и ако ги подредим в ред 2, 3 тогава 2 е началник на 3.  Съответно всички възможни наредби на служителите са:   * 1 2 3 6 4 5 * 1 2 3 6 5 4 * 1 2 3 4 5 6 * 1 2 3 4 6 5 * 1 2 3 5 6 4 * 1 2 3 5 4 6 * 1 2 6 3 4 5 * 1 2 6 3 5 4 |
| 4 2  1 2  3 4 | 6 | Тук всички възможни наредби на служителите са:   * 1 2 3 4 * 1 3 2 4 * 1 3 4 2 * 3 1 2 4 * 3 1 4 2 * 3 4 1 2 |