

Домашно 2 - Дървета и Графи

Организирате празненство за рождения си ден и искате да направите порции ябълков пай за гостите си. За целта отивате с приятелите си в гората, в която има ябълкови дървета, от които трябва да съберете ябълки. Ябълковите дървета са вид двойични дървета, в които в определени възли на дървото има ябълки, а в останалите няма ябълки.

Когато стигнете определено дърво, вие може да започнете да катерите дървото от корена му, като движейки се по клоните на дървото, може да събирате ябълки, ако съществуват такива. Когато се катерите едно ниво нагоре по дървото, това отнема време точно 1 минута. Приемаме, че ако сте на определено ниво в дадено дърво, то имате възможност да вземете всички ябълки на това ниво в дървото. Когато съберете всички ябълки от дървото, трябва да слезете от дървото. Времето за слизане отнема също по 1 минута за ниво.

Тъй като рождения ви ден наближава, трябва да съберете всички ябълки от гората. Предполага се, че докато не се качите на дървото не знаете колко ябълки има там, така че трябва да обходите всички дървета изцяло. Времето за разходка между две дървета е предварително фиксирано и се определя от данни, въведени от стандартния вход. За съжаление, тъй като се разхождате в гора, не между всеки две дървета има директен път. Всички дървета образуват неориентиран и незадължително свързан, претеглен граф.

На стандартния изход, при прочетени от потребителя данни за пътищата между дърветата и данни за всяко индивидуално дърво, изведете времето за обхождане на достъпните в гората дървета и цялото количество ябълки, което може да съберете за рождения си ден. Обхождането на гората приключва, когато обходите всички дървета в гората, до които можете да стигнете.

Всяко дърво ще има собствен пореден номер. Самото то ще бъде зададено на стандартния вход на един ред по следния начин:

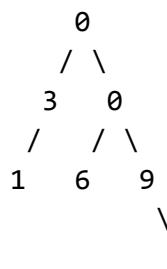
Празно дърво ще бележим с X. Елементите на непразните дървета ще са естествени числа, като положително число означава ябълка, а 0 означава празен връх.

Дърво с корен A и поддървета представени като T1 и T2 ще бележим така: (A T1 T2).

Например,

(0 (3 (1 x x) x) (0 (6 x x) (9 x (0 x x))))

е представянето на следното дърво



На първия ред на входа ще има две числа - броя дървета и броя пътеки между тях. Следват по един ред за всяко дърво. Дърветата са зададени според номерата си - първо дърво с номер 0, след това дърво с номер 1 и т.н.

Следват данните за всички пътечки в гората - по една на ред. Описанието на една пътечка е три цели числа - номерата на началното дърво, крайното дърво и времето за преминаване между тях. Приемете, че дърво с номер 0 е на входа на гората и съответно времето за придвижване на героя до него е 0.

Трябва да обработите входа коректно и да построите дърветата, след което да ги анализирате. Съблюдавайте добрите практики в езика. Обработете ситуации на появя на грешки от произволен характер, включително невалидни входни данни. Грижете се за коректно използване на паметта и всички системни ресурси. Поддържайте силна сигурност

при възникване на изключение - включително не трябва да допускате изтичане на памет, независимо от възникване на някакво изключение, както и получаване на обекти в неконсистентно или невалидно състояние.

Важно: Решенията на тази задача ще преминат през автоматизирани тестове. Важно е вашият формат на вход и изход да съвпада точно с този на дадените примери долу. НЕ добавявайте пояснителни текстове като "Please enter number of trees:" или "Result is:".

Примерен вход 1:

```
1 0  
( 0 ( 3 x x ) ( 8 ( 5 x x ) x ) )
```

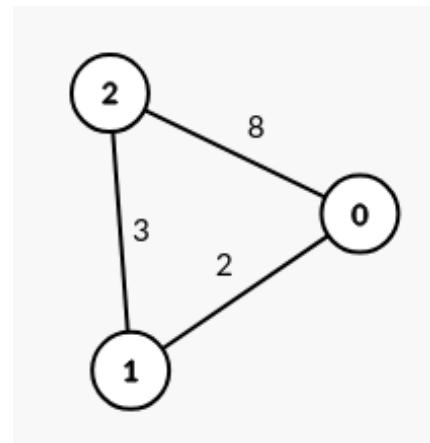
Примерен изход 1:

```
3 6
```

Обяснение към Пример 1: В гората има едно дърво. Това дърво няма ябълки на първото си ниво и има ябълки на второто си ниво и на третото си ниво. Това означава, че героят се катери по 3 нива и спира по 3 нива, което отнема 6 минути. Събраният брой ябълки е 3. Обърнете внимание, че катеренето до ниво 1 също отнема време.

Примерен вход 2:

```
3 3  
( 0 ( 1 x x ) ( 2 x x ) )  
( 1 ( 4 x x ) ( 0 x x ) )  
( 2 ( 0 x x ) ( 0 x x ) )  
  
0 1 2  
1 2 3  
0 2 8
```



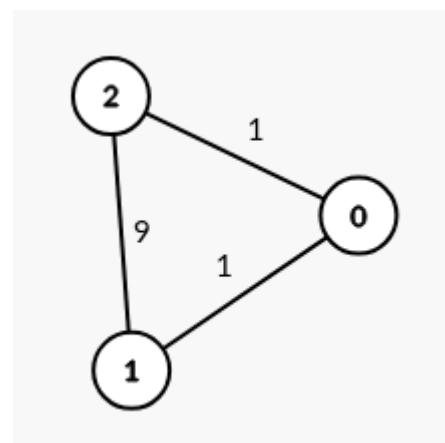
Примерен изход 2:

```
5 12
```

Обяснение към Пример 2: В гората има 3 дървета. Във всяко дърво има поне една ябълка. Обхождането на дърветата отнема съответно по 4 минути всяко, което е общо 12 минути за обхождането на самите дървета. Забележете, че дърво 2 трябва да бъде катерено до край, въпреки че в листата му може да няма никакви ябълки. Събранныте ябълки са 5, а общото време е 12 минути.

Примерен вход 3:

```
3 3  
( 1 ( 3 x x ) ( 7 x x ) )  
( 3 ( 7 x x ) ( 9 x x ) )  
( 0 ( 0 x x ) x )  
  
0 1 1  
1 2 9  
0 2 1
```



Примерен изход 3:

Обяснение към Пример 3: В гората има 3 дървета. Ябълки има само в дървета 0 и 1. Тъй като героят не знае в кое дърво има ябълки и в кое не, трябва да покатери всяко дърво. Общото време за изкачване на дърветата и тук е 12. Тук той изкачва и дървото, в което няма ябълки. Общийят брой ябълки е 6 за време 12 минути.

Забележка: В гората може да има дървета, които не са достигими от дърво 0. Достигимост между дървета с номер X и номер Y означава да съществува поредица от дървета, незадължително различни, с начало X и край Y. Когато не е възможно да се съберат всички ябълки в гората поради недостигимост, изведете подходящо съобщение.

Бонус: Пристъпете към тази стъпка само ако сте решили коректно задачата дотук. Намерете минималното време за обхождане на дърветата в гората, започвайки от дърво с номер 0, минавайки през всички достъпни дървета и приключвайки при кое да е от тях. Изведете нужното минимално време и самият път чрез с номерата на дърветата, през които минавате в съответния им ред. Ако съществува повече от един оптимален път, изведете който и да е от тях.

С данните от Пример 1 минималното време за обхождане на гората е 0. С данните от Пример 2 минималното време за обхождане на гората е 5, а самият маршрут е $0 \rightarrow 1 \rightarrow 2$. С данните от Пример 3 пътят за обхождане на всички дървета е $0 \rightarrow 1 \rightarrow 0 \rightarrow 2$, с общо време на пътуването между дърветата 3. Обърнете внимание, че в този пример път, който преминава между дърветата с номера 1 и 2 не е оптимален и не води до минимално време.