

D. Лазерний удар

Задача	Лазерний удар
Обмеження часу	3 секунди
Обмеження використання пам'яті	1 ГБ

Олеся та її подруга Таня нещодавно відкрили для себе нову настільну гру, яка стала їхньою улюбленою: «Лазерний удар». У цій грі два гравці працюють разом, щоб видалити N фішок з дошки. Гра проходить у два етапи. Загвоздка в тому, що Таня не матиме повної інформації про гру. Щоб виграти гру, Олеся та Таня мають працювати разом, спілкуючись якомога менше.

На дошці є N унікальних фішок, пронумерованих від 0 до $N - 1$. Обидва гравці можуть бачити ці фішки. Також між парами фішок існує $N - 1$ зв'язків, так що можна дістатися до будь-якої фішки з будь-якої іншої фішки, слідуючи цим зв'язкам. Іншими словами, ці зв'язки утворюють дерево. **Тільки Олеся може бачити ці зв'язки; Таня їх не знає.**

На першому етапі гри Олеся визначає порядок $\ell_0, \ell_1, \dots, \ell_{N-2}$, у якому слід видаляти фішки, доки не залишиться лише одна. Цей порядок буде триматися в таємниці від Тані. Якщо вона зможе його відтворити, вони виграють гру. Видалення фішок має відповідати наступному правилу: щоразу, коли фішка видаляється, вона повинна бути з'єднана рівно з однією фішкою, що залишилася. Іншими словами, видалений шматок має бути листком дерева, утвореним рештою шматочків та ним самим. (Після того, як $N - 1$ фішок буде видалено, остання фішка видаляється автоматично, і гравці виграють.) Олеся повинна обрати порядок, який відповідає вищезазначеному правилу.

Олеся також запише повідомлення для Тані у вигляді двійкового рядка. Олеся може вибрати довжину цього повідомлення, але чим воно коротше, тим більше балів вони отримають.

Після цього починається другий етап гри. Мета гри полягає в тому, щоб Таня прибрала з дошки $N - 1$ фішок у порядку $\ell_0, \ell_1, \dots, \ell_{N-2}$. Вона зробить $N - 1$ ходів. Перед ходом i , Олеся передає Тані пару цілих чисел a, b з такими властивостями:

- $a < b$;
- все ще існує пара безпосередньо з'єднаних фішок з номерами a та b ; та
- a або b — це правильна фішка ℓ_i , яку слід видалити в цьому ході.

Зауважте, що для Олесі зв'язок (a, b) однозначно визначається листком ℓ_i у поточному дереві.

Потім Таня прибирає з дошки або a , або b . Якщо це була правильна фішка, тобто ℓ_i , то вони продовжують грати. Інакше вони програють гру.

Ваше завдання — реалізувати стратегії Олесі та Тані, щоб вони виграли гру.

Ваша програма буде оцінюватися залежно від довжини повідомлення, яке Олеся напише на першому етапі гри.

Деталі реалізації

Це задача run-twice, тобто ваша програма буде виконана двічі. Під час першого запуску має бути реалізована стратегія Олесі (Ann в оригіналі) для першої фази гри. Після цього слід реалізувати стратегію Тані (Kathrin в оригіналі) для другої фази гри.

Перший рядок вхідних даних містить два цілі числа, P та N , де P дорівнює або 1, або 2 (перша або друга фаза), а N – кількість фішок.

Наступні вхідні дані залежать від фази:

Фаза 1: Олеся

Після першого рядка (описаного вище) наступні $N - 1$ рядків вхідних даних описують дерево. Кожен рядок містить два цілі числа, a та b ($0 \leq a < b \leq N - 1$), що вказують на зв'язок між фішками a та b .

Ваша програма повинна починатися з виведення двійкового рядка, що містить не більше 1 000 символів, кожен з яких 0 або 1 - повідомлення, написане Олесею. Зверніть увагу, що для генерації рядка довжиною 0 має бути виведений порожній рядок.

Після цього слід вивести $N - 1$ цілих чисел $\ell_0, \ell_1, \dots, \ell_{N-2}$ на окремих рядках, що вказують порядок, у якому Олеся хоче видалити листки з дерева. Порядок має бути таким, щоб якщо фішки видаляти з дерева одну за одною в цьому порядку, то видалена фішка завжди мала бути листком, тобто дерево завжди має залишатися зв'язаним.

Фаза 2: Таня

Після першого рядка (описаного вище), наступний рядок вхідних даних містить двійковий рядок (повідомлення Олесі) з Фази 1.

Після цього буде $N - 1$ раундів взаємодії, по одному для кожного з ходів Тані.

На i -му ході, ваша програма повинна спочатку зчитати два числа, a та b ($0 \leq a < b \leq N - 1$). Одна з цих фішок — це листок ℓ_i у порядку Олесі, а інша — єдина фішка, що залишилася, з'єднана з ℓ_i . Потім ваша програма повинна вивести ℓ_i , що вказує на те, що Таня видаляє цей листок. Якщо ваша програма не виведе правильний листок ℓ_i , дівчата програють гру, а ваш вердикт буде "Неправильна відповідь" для цього тестового випадку.

Деталі

Якщо сума часу виконання двох окремих запусків вашої програми перевищує часовий ліміт, вашу відправку буде оцінено як таку, що перевищила обмеження часу.

Обов'язково очищуйте стандартний вивід після друку кожного рядка, інакше ваша програма може бути оцінена як така, що перевищує обмеження часу. У Python це відбувається автоматично, якщо ви використовуєте `input()` для читання рядків. У C++ `cout << endl`; зчищає рядки у тому числі; якщо використовується `printf`, використовуйте `fflush(stdout)` ; .

Зверніть увагу, що правильне читання порожнього рядка може бути складним. Надані шаблони правильно обробляють цей випадок.

Обмеження та оцінювання

- $N = 1\,000$.
- $0 \leq a < b \leq N - 1$ для всіх з'єднань.

Ваше рішення буде перевірено на наборі тестових груп, кожна з яких оцінюється в певну кількість балів. Кожна тестова група містить набір тестових випадків. Щоб отримати бали за тестову групу, вам потрібно розв'язати всі тестові випадки в тестовій групі.

Група	Максимальний бал	Обмеження
1	8	Дерево — це зірка. Тобто всі вершини, крім однієї, є листками.
2	9	Дерево є лінією. Тобто, всі вершини, крім двох листових вершин, мають рівно дві сусідні вершини.
3	21	Дерево — це зірка з лініями, що виходять з неї. Тобто, всі вершини мають одну або дві сусідні вершини, окрім однієї, яка має більше двох сусідніх вершин.
4	36	Відстань між будь-якими двома вершинами не перевищує 10.
5	26	Додаткових обмежень немає.

За кожну тестову групу, яку ваша програма розв'яже правильно, ви отримаєте бал на основі наступної формули:

$$\text{score} = S_g \cdot (1 - 0.3 \cdot \log_{10} \max(K, 1)),$$

де S_g – максимальний бал для тестової групи, а K – це максимальна необхідна довжина повідомлення Олесі для будь-якого тестового випадку в тестовій групі. **Ваш бал за кожну тестову групу буде округлено до найближчого цілого числа.**

У таблиці нижче показано кількість балів для кількох значень K , яку отримає ваша програма, якщо вона розв'яже всі тестові групи з цим K . Зокрема, щоб отримати 100 балів, ваше рішення має вирішити кожен тестовий випадок з $K \leq 1$.

К	1	5	10	50	100	500	1000
Балів	100	79	70	49	39	20	11

Інструмент тестування

Щоб полегшити тестування вашого рішення, ми надали простий інструмент, який ви можете завантажити. Дивіться «attachments» внизу сторінки у задачі на Kattis. Використання цього інструменту необов'язкове. Зверніть увагу, що офіційний градер на Kattis відрізняється від інструменту тестування.

Щоб скористатися інструментом, створіть вхідний файл, наприклад, «sample1.in», який має починатися з числа N за яким слідує $N - 1$ рядок, що описують дерево, у тому ж форматі, що й у Фазі 1. Наприклад, для прикладу нижче:

```
7
0 1
1 2
2 3
0 4
0 6
1 5
```

Для програм на Python, наприклад, `solution.py` (зазвичай виконується як `python3 solution.py`), запустіть:

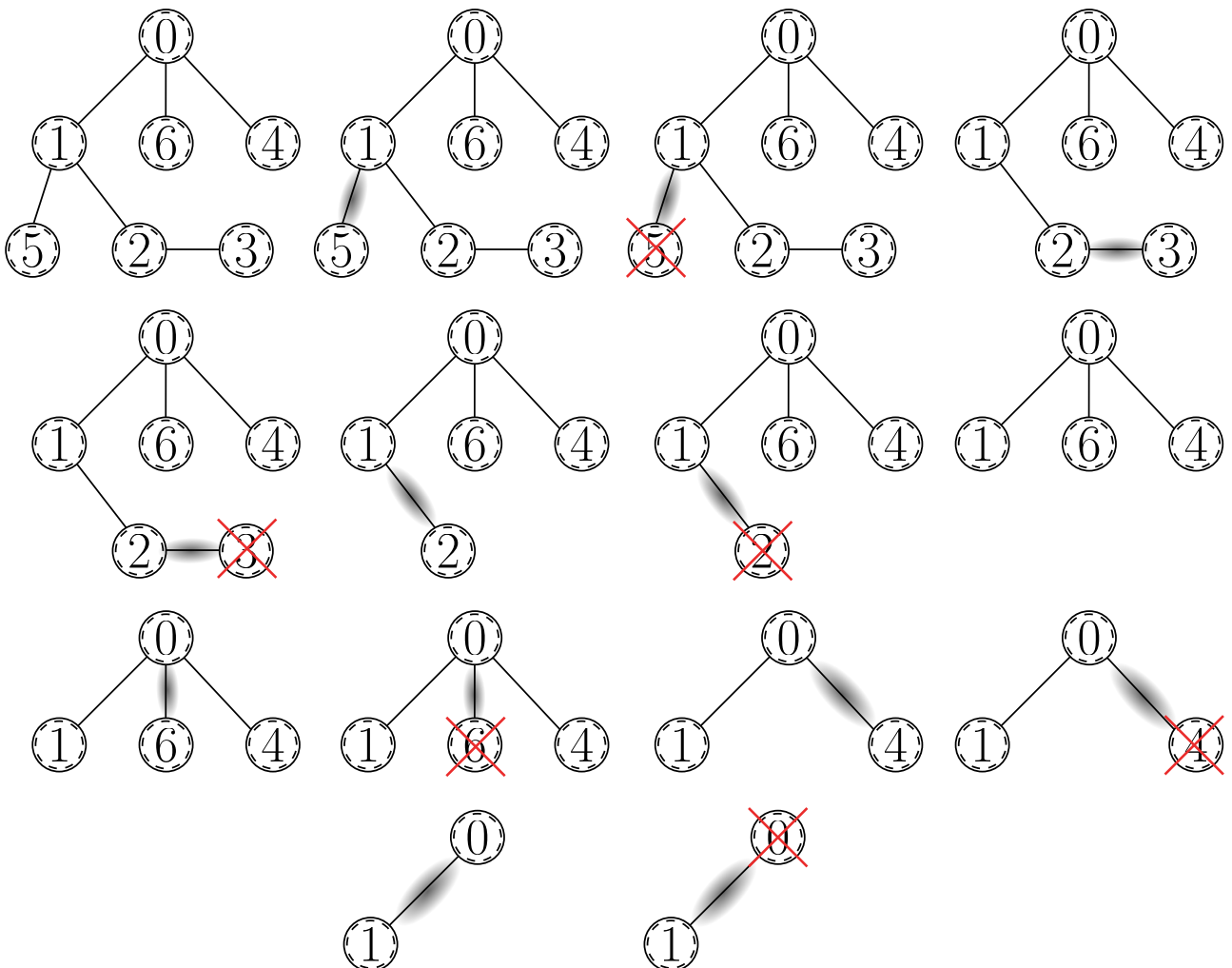
```
python3 testing_tool.py python3 solution.py < sample1.in
```

Для програм на C++ спочатку скомпілюйте їх (наприклад, за допомогою `g++ -g -O2 -std=gnu++23 -static solution.cpp -o solution.out`), а потім виконайте:

Приклад

Зверніть увагу, що для спрощення приклад у цьому розділі має $N = 7$ і тому не є коректним тестовим випадком. Не очікується, що ваша програма зможе розв'язати цей випадок. Усі тестові випадки у градері матимуть $N = 1\,000$.

У прикладі Олесі задано наступне дерево. На першому етапі Олеся зчитує дерево, вибирає двійковий рядок "0110" для надсилання Таня, а також вибирає порядок $[\ell_0, \ell_1, \dots, \ell_{N-2}] = [5, 3, 2, 6, 4, 0]$ у якому фішки слід видалити з дерева. На другому етапі Таня отримує рядок "0110", який був надісланий на першому етапі. Потім вона отримує пару $(1, 5)$ і вирішує видалити вершину 5, яка справді є листком. Для наступного ходу вона отримує пару $(2, 3)$ та видаляє листок 3 і так далі. На наступних зображеннях показано ці взаємодії:



вивід градера	ваш вивід
1 7	
0 1	
1 2	
2 3	
0 4	
0 6	
1 5	
	0110
	5
	3
	2
	6
	4
	0

вивід градера	ваш вивід
2 7	
0110	
1 5	
	5
2 3	
	3
1 2	
	2
0 6	
	6
0 4	
	4
0 1	
	0