

D. Laser Strike

Nazwa problemu	Laser Strike
Limit czasu	3 sekundy
Limit pamięci	1 gigabajt

Anna i jej przyjaciółka Kathrina odkryły ostatnio nową grę, która szybko stała się ich ulubioną: Laser Strike. W tej grze obaj gracze współpracują, aby usunąć N pionków z planszy. Gra podzielona jest na dwie fazy. Kathrina nie ma pełnej informacji o stanie gry. Aby wygrać, Anna i Kathrina muszą współpracować jednocześnie komunikując się ze sobą jak najmniej.

Na planszy znajduje się N różnych pionków ponumerowanych od 0 do $N - 1$. Obie przyjaciółki widzą te pionki. Jest również $N - 1$ połączeń między parami pionków. Z każdego pionka można przejść do każdego innego używając tych połączeń. Innymi słowy, połączenia tworzą drzewo. **Tylko Anna widzi te połączenia; Kathrina nie widzi ich wcale.**

W pierwszej fazie gry, Anna ustala kolejność $\ell_0, \ell_1, \dots, \ell_{N-2}$, w której będą usuwane pionki do czasu, aż zostanie tylko jeden. Kolejność ta nie jest znana dla Kathriny. Jeżeli Kathrina będzie potrafiła ją odtworzyć, przyjaciółki wygrywają grę. Podczas usuwania pionków musi być spełniona następująca własność: za każdym razem kiedy usuwany jest pionek, musi on mieć dokładnie jedno połączenie z jakimś nieusuniętym pionkiem. Innymi słowy, usuwany pionek musi być liściem drzewa utworzonego z nieusuniętych pionków i pozostałych między nimi połączeń. Po usunięciu $N - 1$ pionków, ostatni pionek jest usuwany automatycznie a przyjaciółki wygrywają grę. Anna musi ustalić kolejność usuwania pionków, która przestrzega powyższej własności.

Ponadto, Anna przekaże Kathrinie wiadomość w postaci binarnego napisu. Anna sama decyduje jak długa będzie wiadomość – jednak im krótsza jest, tym więcej punktów zostanie przyznane za test.

Po tym, następuje druga faza gry. Celem gry jest, aby Kathrina usunęła pionki z planszy w kolejności $\ell_0, \ell_1, \dots, \ell_{N-2}$. Wykona $N - 1$ ruchów. Przed i -tym ruchem, Anna przekaże Kathrinie parę liczb całkowitych a, b , które spełniają następujące własności:

- $a < b$,
- istnieje para bezpośrednio połączonych nieusuniętych pionków o numerach a i b oraz
- pionek a albo pionek b to pionek ℓ_i , który powinien zostać usunięty w tym ruchu.

Zwróć uwagę, że dla Anny połączenie (a, b) jest jednoznacznie wyznaczone przez liść ℓ_i w aktualnym drzewie.

Następnie Kathrina usuwa albo pionek a , albo pionek b z planszy. Jeżeli był to dobry pionek – tj. pionek ℓ_i – przyjaciółki kontynuują grę. W przeciwnym razie przegrywają grę.

Twoim zadaniem jest zaimplementować obie strategie – Anny oraz Kathriny – tak, aby wygrały grę.

Twój program zostanie oceniony zależnie od długości wiadomości, którą Anna przekaże w pierwszej fazie gry.

Implementacja

Twój program zostanie uruchomiony dwukrotnie. Za pierwszym razem Twój program powinien wykonać strategię Anny dla pierwszej fazy. Za drugim razem Twój program powinien wykonać strategię Kathriny dla drugiej fazy.

Pierwszy wiersz wejścia zawiera dwie liczby całkowite P oraz N , gdzie P jest równe 1 albo 2 (oznaczające pierwszą albo drugą fazę) oraz N oznacza liczbę pionków.

Następnie, wejście ma format zależny od fazy:

Faza 1: Anna

Po pierwszym wierszu (opisanym wyżej) znajduje się $N - 1$ wierszy opisujących drzewo. Każdy wiersz zawiera dwie liczby całkowite a oraz b ($0 \leq a < b \leq N - 1$) oznaczające, że istnieje połączenie między pionkami a oraz b .

Twój program powinien najpierw wypisać napis binarny o długości co najwyżej 1000, składający się ze znaków 0 lub 1 – wiadomość od Anny. Jeżeli napis ma mieć długość 0, wypisz pusty wiersz.

Następnie wypisz $N - 1$ liczb całkowitych $\ell_0, \ell_1, \dots, \ell_{N-2}$, każdą w oddzielnym wierszu, oznaczających kolejność, w której Anna chce usunąć pionki. Usuwanie z drzewa w tej kolejności pionki jeden po drugim powinny zawsze być liśćmi, tj. drzewo cały czas musi pozostać spójne.

Faza 2: Kathrina

Po pierwszym wierszu (opisanym wyżej) znajduje się wiersz z binarnym napisem (wiadomość Anny) z fazy 1.

Po tym następuje $N - 1$ interakcji, po jednej na każdy ruch Kathriny.

W i -tym ruchu Twój program powinien najpierw wczytać dwie liczby całkowite a oraz b ($0 \leq a < b \leq N - 1$). Jeden z tych pionków jest liściem ℓ_i w kolejności Anny, a drugi pionek jest jedynym nieusuniętym pionkiem połączonym z ℓ_i . Następnie Twój program powinien wypisać ℓ_i

oznaczające numer pionka będącego liściem, który Kathrina usuwa. Jeżeli Twój program niepoprawnie wypisze numer liścia ℓ_i , przyjaciółki przegrają grę i Twoje zgłoszenie dostanie werdykt `Wrong Answer` dla tego testu.

Szczegóły

Jeżeli *suma* czasów wykonania obu uruchomień Twojego programu przekroczy limit czasu, zgłoszenie otrzyma werdykt `Time Limit Exceeded`.

Upewnij się, że opróżniasz bufor wyjścia po wypisaniu każdej przejażdżki. W przeciwnym wypadku Twój program może dostać werdykt `Time Limit Exceeded`. W Pythonie dzieje się to automatycznie, jeżeli używasz `input()` do wczytywania wierszy. W C++, `cout << endl;` opróżnia bufor oraz wypisuje znak nowej linii; jeżeli używasz `printf`, użyj `fflush(stdout)`.

Przeczytanie pustego napisu może być trudne. Dostarczone szablony poprawnie sobie radzą z tym przypadkiem.

Ograniczenia i punktacja

- $N = 1\,000$.
- $0 \leq a < b \leq N - 1$ dla każdego połączenia.

Twoje rozwiązanie zostanie przetestowane na kilku grupach testowych, z których każda jest warta określoną liczbę punktów. Każda grupa testowa zawiera zestaw testów. Aby zdobyć punkty dla grupy testowej, należy rozwiązać wszystkie testy w danej grupie testowej.

Grupa	Maksymalna liczba punktów	Ograniczenia
1	8	Drzewo jest gwiazdą, czyli wszystkie wierzchołki poza jednym są liśćmi.
2	9	Drzewo jest ścieżką, czyli wszystkie wierzchołki poza dwoma mają dokładnie dwa sąsiadujące wierzchołki.
3	21	Drzewo jest gwiazdą z wychodzącymi ścieżkami, czyli każdy wierzchołek ma jeden lub dwa sąsiadujące wierzchołki poza jednym, który ma więcej niż dwa sąsiadujące wierzchołki.
4	36	Odległość między dowolnymi dwoma wierzchołkami nie przekracza 10.
5	26	Brak dodatkowych ograniczeń.

Za każdą grupę rozwiązaną prawidłowo przez Twój program otrzymasz wynik zgodnie z następującym wzorem:

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

gdzie S_g jest maksymalnym wynikiem dla grupy, a K to maksymalna długość wiadomości użyta przez Annę dla dowolnego testu w grupie. **Twój wynik w każdej grupie zostanie zaokrąglony do najbliższej liczby całkowitej.**

Poniższa tabela przedstawia liczbę punktów, dla kilku wartości K , jaką Twój program otrzyma, jeśli rozwiąże wszystkie grupy z tą wartością K . W szczególności, aby uzyskać wynik 100 punktów, Twoje rozwiązanie musi rozwiązać każdy test z $K \leq 1$.

K	1	5	10	50	100	500	1000
Score	100	79	70	49	39	20	11

Narzędzie do testowania

Aby ułatwić testowanie Twoich rozwiązań, dostarczone zostało narzędzie do testowania, które możesz pobrać. Zobacz "attachments" na dole strony zadania w Kattis. Korzystanie z narzędzia nie jest obowiązkowe. Rzeczywisty interaktor w Kattis różni się od dostarczonego narzędzia.

Aby użyć tego narzędzia, utwórz plik wejściowy, taki jak "sample1.in", który zaczyna się od liczby N po której następuje $N - 1$ wierszy opisujących drzewo, w tym samym formacie, jak w fazie 1. Na przykład w poniższym przykładzie:

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

W przypadku programów w języku Python, np. `solution.py` (normalnie uruchamiane jako `pypy3 solution.py`), uruchom:

```
python3 testing_tool.py pypy3 solution.py < sample1.in
```

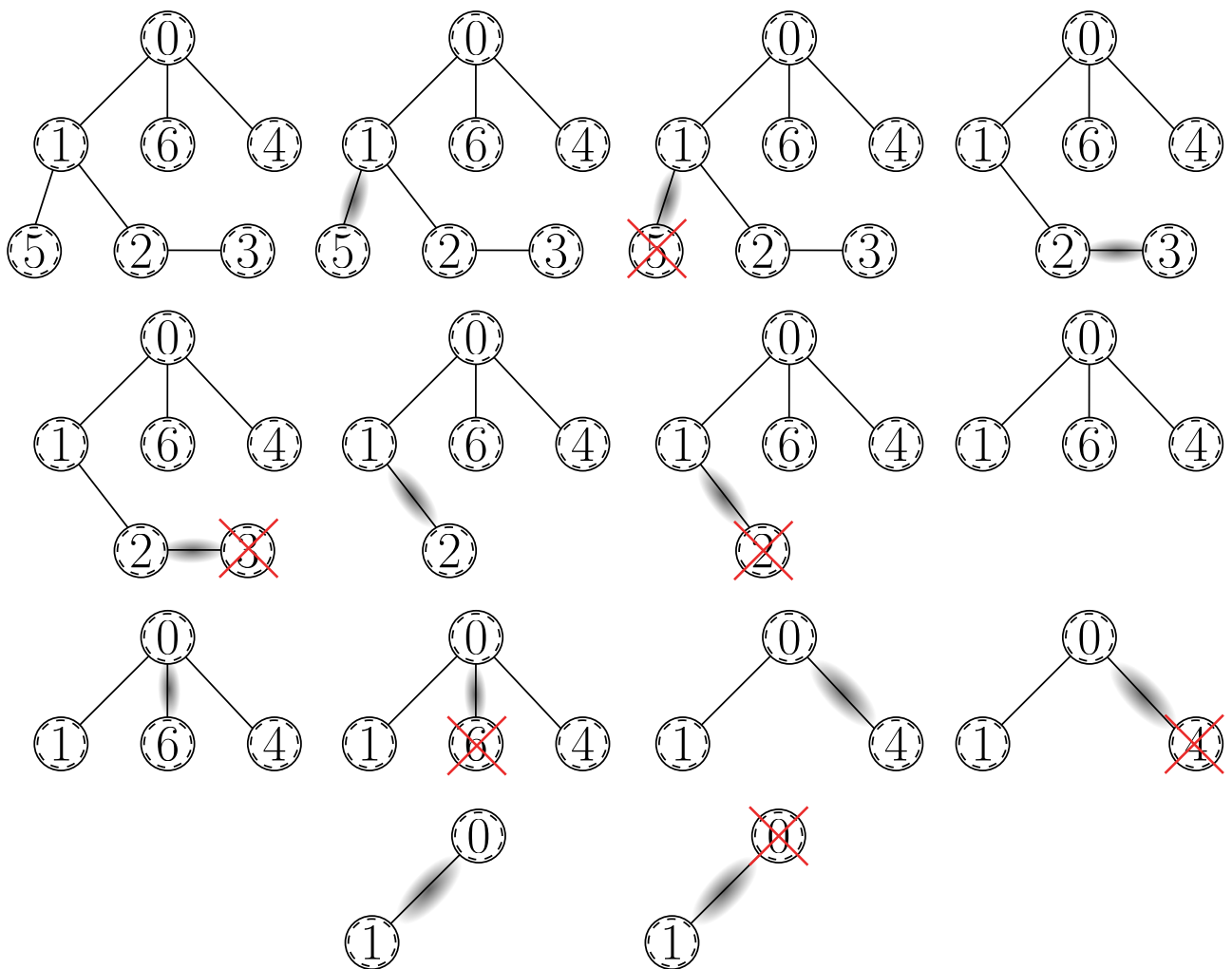
W przypadku programów C++, najpierw skompiluj go: (np. z `g++ -g -O2 -std=gnu++23 -static solution.cpp -o solution.out`) a następnie uruchom:

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

Przykład

Dla uproszczenia, przykład w tej sekcji ma wartość $N = 7$, a zatem nie jest poprawnym testem. Twój program nie musi być w stanie rozwiązać tego przypadku. Wszystkie testy będą miały wartość $N = 1000$.

W przykładzie Anna otrzymuje następujące drzewo. W pierwszej fazie Anna odczytuje drzewo, wybiera ciąg binarny "0110" do wysłania do Kathriny, a także wybiera kolejność $[\ell_0, \ell_1, \dots, \ell_{N-2}] = [5, 3, 2, 6, 4, 0]$, w której elementy powinny zostać usunięte z drzewa. W drugiej fazie Kathrina otrzymuje ciąg znaków "0110", który został wysłany w pierwszej fazie. Następnie otrzymuje parę $(1, 5)$ i decyduje się usunąć wierzchołek 5, który rzeczywiście jest liściem. W następnym ruchu otrzymuje parę $(2, 3)$ i usuwa liść 3, i tak dalej. Poniższe obrazki przedstawiają te interakcje:



Wyjście interaktora	Twoje wyjście
1 7	
0 1	
1 2	
2 3	
0 4	
0 6	
1 5	
	0110
	5
	3
	2
	6
	4
	0

Wyjście interaktora	Twoje wyjście
2 7	
0110	
1 5	
	5
2 3	
	3
1 2	
	2
0 6	
	6
0 4	
	4
0 1	
	0