

Zadanie: GLO

Głosowanie



XXIX OI, etap III, dzień pierwszy. Plik źródłowy glo.* Dostępna pamięć: 256 MB. 6.04.2022

Prezes pewnej firmy chciałby zwiększyć morale swoich pracowników, organizując dla nich cytrusowe czwartki. Nie może się zdecydować, czy zlecić zakup pomarańczy, czy też może grejpfrutów, więc postanowił, że zdecydują o tym sami pracownicy.

Głosowanie będzie przebiegać zgodnie z hierarchią. W firmie pracuje n pracowników ponumerowanych liczbami od 1 do n , wliczając w to prezesa oznaczonego jest numerem 1. Każdy pracownik firmy jest albo kierownikiem, albo szeregowym pracownikiem, przy czym prezes na pewno jest kierownikiem. Każdy kierownik ma bezpośrednio pod sobą *nieparzystą* liczbę podwładnych, którzy mogą być szeregowymi pracownikami lub innymi kierownikami. Szeregowi pracownicy nie mają podwładnych. Każdy pracownik poza prezesem (który przełożonego nie ma) ma dokładnie jednego bezpośredniego przełożonego. W hierarchii firmy nie występują cykle.

Każdy szeregowy pracownik decyduje samodzielnie, czy zagłosować za pomarańczami, czy za grejpfrutami. Każdy kierownik głosuje tak, jak głosuje *większość* jego (bezpośrednich) podwładnych. Wynik głosowania zależy od głosu prezesa.

Bajtazar jest właścicielem hurtowni pomarańczy i zależy mu na tym, aby w głosowaniu zwyciężyły pomarańcze. Niestety, jego dobry przyjaciel Bajtoni jest właścicielem hurtowni grejpfrutów i wołałby, aby zwyciężyły grejpfruty. Przyjaciele postanowili zatem zagrać w grę: na przemian (zaczynając od Bajtazara) będą wybierać po jednym szeregowym pracowniku firmy, z którym jeszcze żaden z nich nie rozmawiał, i nakłaniać go do głosowania na swoje cytrusy (Bajtazar i Bajtoni mają bardzo duży dar przekonywania). Gra zakończy się, gdy porozmawiają ze wszystkimi szeregowymi pracownikami.

Pomóż Bajtazarowi w ustaleniu, jak ma wybierać pracowników, aby ostatecznie w głosowaniu zwyciężyły pomarańcze. Możesz założyć, że hierarchia firmy jest tak dobrana, że Bajtazar będzie mógł zawsze (niezależnie od ruchów Bajtoniego) doprowadzić do zwycięstwa.

Komunikacja

Należy zaimplementować program, który pomoże Bajtazarowi, korzystając z dostarczonej biblioteki (symulującej ruchy Bajtoniego).

Aby użyć biblioteki, należy wpisać w swoim programie:

- C/C++: `#include "glolib.h"`
- Python: `from glolib import daj_n, daj_przezonego, ruch`

Biblioteka udostępnia następujące funkcje:

- `daj_n()` – Ta funkcja zwraca liczbę całkowitą n , oznaczającą liczbę pracowników w firmie.
- `daj_przezonego(v)` – Ta funkcja zwraca numer bezpośredniego przełożonego pracownika o numerze v (dla $1 < v \leq n$). Numer przełożonego jest zawsze mniejszy od numeru pracownika.
- `ruch(x)` – Ta funkcja informuje bibliotekę, że Bajtazar wykonuje kolejny ruch, wybierając szeregowego pracownika o numerze x . Funkcja zwraca numer pracownika wybranego w kolejnym ruchu przez Bajtoniego. Jeśli ruch Bajtazara był ostatnim ruchem w całej grze, to po wywołaniu tej funkcji program automatycznie zostanie zakończony (możesz założyć, że to zawsze Bajtazar wykona ostatni ruch).

Funkcji `daj_n` i `daj_przezonego` możesz używać wielokrotnie. Twój program nie może otwierać żadnych plików ani używać standardowego wejścia i wyjścia. Może on korzystać ze standardowego wyjścia diagnostycznego (`stderr`), jednak pamiętaj, że zużywa to cenny czas. Rozwiązanie będzie kompilowane wraz z biblioteką następującymi poleceniami:

- C++: `g++ -O3 -static -std=c++17 glolib.cpp glo.cpp`
- Python: `python3 glo.py`

Uwaga: Podane na górze ograniczenie pamięci dotyczy tylko Twojego rozwiązania, a zatem nie wlicza pamięci wykorzystywanej przez bibliotekę.

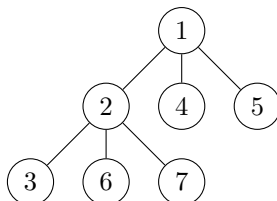
Ocenianie

Zestaw testów dzieli się na następujące podzadania. Testy do każdego podzadania składają się z jednej lub większej liczby osobnych grup testów.

Podzadanie	Ograniczenia	Punkty
1	$2 \leq n \leq 13$	20
2	$2 \leq n \leq 1000$	40
3	$2 \leq n \leq 200\,000$	40

Przykładowy przebieg programu

Poniżej przedstawiono przebieg programu dla testu przykładowego, w których hierarchia firmy wygląda następująco:



Wywołanie	Wynik	Komentarz
daj_n()	7	$n = 7$
daj_przełożonego(2)	1	Przełożonym 2 jest 1
daj_przełożonego(3)	2	Przełożonym 3 jest 2
daj_przełożonego(4)	1	Przełożonym 4 jest 1
daj_przełożonego(5)	1	Przełożonym 5 jest 1
daj_przełożonego(6)	2	Przełożonym 6 jest 2
daj_przełożonego(7)	2	Przełożonym 7 jest 2
ruch(3)	5	Bajtazar rozmawia z pracownikiem 3, Bajtoni z 5
ruch(7)	4	Bajtazar rozmawia z pracownikiem 7, Bajtoni z 4; po tym ruchu Bajtazar już nie ma szans na wygraną
ruch(6)		Bajtazar rozmawia z pracownikiem 6; program kończy działanie

Gdyby Bajtazar w swoim drugim ruchu porozmawiał na przykład z pracownikiem numer 4, zagwarantowałyby sobie zwycięstwo.

Testy „ocen”:

- 1ocen:** $n = 19$, prezes ma pod sobą trzech kierowników, a każdy z nich ma pod sobą po pięciu szeregowych pracowników;
- 2ocen:** $n = 364$, każdy kierownik ma bezpośrednio pod sobą dokładnie trzy osoby i wszyscy pracownicy szeregowi znajdują się w tej samej odległości od prezesa;
- 3ocen:** $n = 200\,000$, w firmie jest 100 001 kierowników, każdy z nich (poza prezesem) jest bezpośrednim podwładnym kierownika o numerze mniejszym o 1; ponadto 99 999 pracowników szeregowych jest podległych bezpośrednio pod kierownika o numerze 100 001.

Eksperymenty

Przykładowe **błędne** rozwiązania wraz z przykładowymi bibliotekami znajdują się w folderze **dłazaw**. Biblioteki mogą różnić się zachowaniem od tych na sprawdzaczkach i nie spełniać założeń zadania. Mają one jedynie pokazać sposób interakcji z programem.

Twoje rozwiązanie skompilowane z przykładową biblioteką wczytuje ze standardowego wejścia opis hierarchii w firmie – liczbę n , a następnie $n - 1$ liczb p_2, \dots, p_n , oddzielonych odstępami – przy czym wartość p_i oznacza numer bezpośredniego przełożonego pracownika o numerze i . W przykładowej bibliotece Bajtoni przekonuje zawsze szeregowego pracownika o najmniejszym numerze, który jeszcze nie został przekonany. Taka sama strategia jest używana na teście przykładowym i testach ocen w SIO. Przykładowa biblioteka wypisuje kolejne ruchy wykonywane przez Bajtazara i Bajtoniego.