Adwokat

Dostępna pamięć: 256 MB.

Adwokat Bajtazar, współwłaściciel kancelarii *Bajtazar i wspólnicy*, jest jednym z najbardziej rozchwytywanych członków bajtockiej palestry. Nic więc dziwnego, że jest wiecznie zajęty. Każdego dnia umawia się na liczne spotkania i dawno już przestał kontrolować to, czy będzie w stanie uczestniczyć w nich wszystkich. Zatrudnił więc sekretarza, który ma mu pomóc w ogarnięciu tego chaosu. Bajtazar postanowił, że każdego dnia uda się tylko na dwa spotkania i będzie w nich uczestniczył od początku do końca. Na pozostałe spotkania zostaną oddelegowani asystenci, których w kancelarii Bajtazara nie brakuje.

Niestety, czasem w przepełnionym kalendarzu Bajtazara trudno znaleźć dwa spotkania, których terminy nie nachodzą na siebie. Przyjmujemy, że dwa spotkania nie nachodzą na siebie, jeśli jedno z nich zaczyna się *ściśle* po zakończeniu drugiego. Pomóż sekretarzowi Bajtazara i napisz program, który poradzi sobie z tym problemem.

Wejście

Pierwszy wiersz wejścia zawiera dwie liczby całkowite n i m ($2 \leqslant n \leqslant 500\,000$, $1 \leqslant m \leqslant 20$), oznaczające liczbę spotkań w terminarzu Bajtazara oraz liczbę dni, które obejmuje terminarz.

W każdym z kolejnych n wierszy opisane jest jedno spotkanie. Opis spotkania składa się z trzech liczb całkowitych a_i, b_i, d_i ($1 \leqslant a_i < b_i \leqslant 80\,000\,000$, $1 \leqslant d_i \leqslant m$), które oznaczają, że dnia d_i Bajtazar ma zaplanowane spotkanie, które rozpocznie się dokładnie a_i milisekund po północy i zakończy się b_i milisekund po północy.

Wyjście

Twój program powinien wypisać na wyjście m wierszy. W i-tym z tych wierszy powinna znaleźć się informacja, czy Bajtazar może uczestniczyć w dwóch spotkaniach i-tego dnia. Jeśli jest to niemożliwe, należy wypisać jedno słowo `NIE`. W przeciwnym razie należy wypisać `TAK`, a następnie numery dwóch spotkań, w których może uczestniczyć Bajtazar. Spotkania są ponumerowane od 1 do n, zgodnie z ich kolejnością na wejściu. Pierwsze z tych dwóch spotkań powinno się rozpoczynać wcześniej. Drugie spotkanie powinno zacząć się co najmniej milisekundę po zakończeniu pierwszego.

Jeśli istnieje wiele poprawnych odpowiedzi, Twój program powinien wypisać dowolną z nich.

Przykład

Dla danych wejściowych:

```
6 3
3 5 1
2 4 2
1 8 1
6 7 3
3 5 2
7 12 1
```

poprawnym wynikiem jest:

```
TAK 1 6
NIE
NIE
```

Benzyna

Dostępna pamięć: 256 MB.

Bajtazar pracuje w dziale transportu bajtockiego giganta paliwowego Bajtoil i planuje dostawy paliwa do stacji benzynowych.

W Bajtocji jest n skrzyżowań (ponumerowanych liczbami od 1 do n) oraz m dwukierunkowych dróg łączących pewne pary skrzyżowań. Przy niektórych skrzyżowaniach stoją stacje benzynowe Bajtoilu.

Flota transportowa firmy składa się z cystern o różnych pojemnościach baków. Każda cysterna spala 1 litr benzyny na kilometr przejechanej drogi. Można więc założyć, że cysterna o pojemności baku b litrów może przejechać maksymalnie b kilometrów bez konieczności uzupełnienia paliwa w baku. Kierowcy cystern nie mogą korzystać z paliwa przewożonego w zbiorniku cysterny, mogą za to za darmo uzupełniać paliwo w baku na stacjach benzynowych Bajtoilu.

Bajtazar w swojej pracy wielokrotnie musi sprawdzać odpowiedzi na pytania: czy cysterna o pojemności baku b litrów może przejechać ze stacji przy skrzyżowaniu x do stacji przy skrzyżowaniu y? Cysterna o pojemności baku b litrów nie może pokonać odcinka dłuższego niż b kilometrów, w trakcie którego nie będzie żadnej stacji benzynowej Bajtoilu. Cysterny zawsze rozpoczynają podróż na skrzyżowaniu, przy którym stoi stacja Bajtoilu i kończą również na skrzyżowaniu, przy którym znajduje się stacja.

Pomóż Bajtazarowi zautomatyzować odpowiadanie na zapytania.

Wejście

W pierwszym wierszu wejścia znajdują się trzy liczby całkowite n, s i m ($2 \leqslant s \leqslant n \leqslant 200\,000$, $1 \leqslant m \leqslant 200\,000$), oznaczające odpowiednio liczbę skrzyżowań, liczbę stacji benzynowych i liczbę dróg w Bajtocji. W drugim wierszu wyjścia znajduje się ciąg s parami różnych liczb całkowitych $c_1, c_2, \ldots, c_s$ ($1 \leqslant c_i \leqslant n$), oznaczających skrzyżowania, przy których stoją stacje Bajtoilu.

W kolejnych m wierszach opisane są drogi w Bajtocji; i-ty z tych wierszy zawiera trzy liczby całkowite u_i, v_i i d_i ($1 \leqslant u_i, v_i \leqslant n$, $u_i \neq v_i$, $1 \leqslant d_i \leqslant 10\,000$), oznaczające, że i-ta z dróg ma długość d_i kilometrów i łączy skrzyżowania u_i i v_i. Pomiędzy każdą parą skrzyżowań istnieje co najwyżej jedna droga.

W następnym wierszu znajduje się jedna liczba całkowita q ($1 \leqslant q \leqslant 200\,000$), oznaczająca liczbę zapytań. W kolejnych q wierszach znajdują się opisy pytań; i-ty z tych wierszy zawiera trzy liczby całkowite x_i, y_i i b_i ($1 \leqslant x_i, y_i \leqslant n$, $x_i \neq y_i$, $1 \leqslant b_i \leqslant 2 \cdot 10^9$), oznaczające zapytanie o możliwość przejazdu cysterną o pojemności baku b_i litrów, ze stacji przy skrzyżowaniu x_i do stacji przy skrzyżowaniu y_i. Można założyć, że przy obu skrzyżowaniach x_i, y_i stoją stacje Bajtoilu.

Wyjście

Twój program powinien wypisać na wyjście dokładnie q wierszy. W i-tym z tych wierszy powinno znaleźć się jedno słowo `TAK` lub `NIE`, w zależności od tego, czy cysterna o pojemności baku b_i jest w stanie przejechać ze skrzyżowania x_i do skrzyżowania y_i.

Przykład

Dla danych wejściowych:

```
6 4 5
1 5 2 6
1 3 1
2 3 2
3 4 3
4 5 5
6 4 5
4
1 2 4
2 6 9
1 5 9
6 5 8
```

poprawnym wynikiem jest:

```
TAK
TAK
TAK
NIE
```

Ceny

Dostępna pamięć: 256 MB.

Bajtazar pracuje jako zaopatrzeniowiec w pewnej bajtockiej restauracji. Każdego dnia wieczorem otrzymuje on od kierownika listę produktów spożywczych, które musi zakupić kolejnego dnia rano. Bajtazar powinien kupić dokładnie jedną sztukę każdego produktu z listy. Kierownikowi zależy na tym, by wykonanie zakupów kosztowało jak najmniej.

Bajtazar siada wieczorem do komputera i sprawdza ceny potrzebnych produktów w lokalnych hurtowniach spożywczych. Zna on także koszt dojazdu z restauracji do każdej z hurtowni i z powrotem. Teraz Bajtazar musi zdecydować, które produkty kupić w każdej z hurtowni.

Dla każdej hurtowni, w której Bajtazar postanowił kupić jakieś produkty, zrobi on co następuje. Pojedzie do tej hurtowni, zrobi w niej zakupy i od razu kupione produkty zawiezie do restauracji. Szczęśliwie, bagażnik jego samochodu jest na tyle duży, że nie musi on do żadnej hurtowni jeździć wielokrotnie. Produkty spożywcze szybko się psują, dlatego w trakcie jednego kursu Bajtazar może odwiedzić tylko jedną hurtownię.

Napisz program, który pomoże Bajtazarowi obliczać najtańszy sposób wykonania wszystkich zakupów.

Wejście

Pierwszy wiersz wejścia zawiera dwie liczby całkowite n oraz m ($1 \leqslant n \leqslant 100$, $1 \leqslant m \leqslant 16$) oznaczające liczbę hurtowni oraz liczbę produktów, które Bajtazar powinien kupić. Kolejne n wierszy zawiera opisy cen w poszczególnych hurtowniach.

Pierwsza liczba w i-tym z tych wierszy, d_i ($1 \leqslant d_i \leqslant 1\,000\,000$), oznacza koszt dojazdu z restauracji do i-tej hurtowni (wliczając powrót). Po niej następuje ciąg m liczb całkowitych $c_{i,1}, c_{i,2}, \ldots, c_{i,m}$ ($1 \leqslant c_{i,j} \leqslant 1\,000\,000$): liczba $c_{i,j}$ oznacza cenę j-tego produktu w i-tej hurtowni.

Wyjście

Twój program powinien wypisać na wyjście jeden wiersz zawierający jedną liczbę całkowitą, oznaczającą sumę cen produktów i kosztów dojazdu do wybranych przez Bajtazara hurtowni w najtańszym możliwym planie zakupów.

Przykład

Dla danych wejściowych:

```
3 4
5 7 3 7 9
2 1 20 3 2
8 1 20 1 1
```

poprawnym wynikiem jest:

```
16
```

Wyjaśnienie do przykładu: W pierwszej hurtowni Bajtazar kupuje produkt numer 2, a w drugiej wszystkie pozostałe produkty. Tak więc nie musi on odwiedzać trzeciej hurtowni.

Dzielniki

Dostępna pamięć: 256 MB.

Dany jest ciąg n liczb całkowitych $a_1, a_2, \ldots, a_n$. Należy wyznaczyć liczbę takich par uporządkowanych (i, j), że $i, j \in \{1, \ldots, n\}$, $i \neq j$ oraz a_i jest dzielnikiem a_j.

Wejście

W pierwszym wierszu wejścia znajduje się jedna liczba całkowita n ($1 \leqslant n \leqslant 2\,000\,000$). W drugim wierszu znajduje się ciąg n liczb całkowitych $a_1, a_2, \ldots, a_n$ ($1 \leqslant a_i \leqslant 2\,000\,000$).

Wyjście

W pierwszym i jedynym wierszu wyjścia należy wypisać jedną liczbą całkowitą, oznaczającą szukaną liczbę par.

Przykład

Dla danych wejściowych:

```
5
2 4 5 2 6
```

poprawnym wynikiem jest:

```
6
```

Wyjaśnienie do przykładu: Istnieje 6 par o podanych własnościach: $(1, 2)$, $(1, 4)$, $(1, 5)$, $(4, 1)$, $(4, 2)$, $(4, 5)$.

Euklidesowy Nim

Dostępna pamięć: 256 MB.

Euklides i Pitagoras to pseudonimy dwóch Bajtocczan słynących z zamiłowania do matematycznych zagadek. Ostatnio wieczory spędzają, grając w następującą grę. Na stole leży stos n kamieni. Przyjaciele wykonują na przemian ruchy. Ruch Euklidesa polega na zabraniu ze stosu dowolnej dodatniej wielokrotności p kamieni (jeśli na stosie jest co najmniej p kamieni) albo dołożeniu do stosu dokładnie p kamieni — dołożyć kamienie można jednak tylko wtedy, gdy na stosie było ich mniej niż p. Ruch Pitagorasa jest analogiczny, z tym że albo zabiera on wielokrotność q kamieni, albo dokłada dokładnie q kamieni. Wygrywa ten z graczy, który opróżni stos. Grę zaczyna Euklides.

Przyjaciele zastanawiają się, czy rozgryźli tę grę do końca. Pomóż im i napisz program, który będzie stwierdzał, jaki powinien być wynik rozgrywki w przypadku optymalnej gry obu graczy.

Wejście

Pierwszy wiersz wejścia zawiera jedną liczby całkowitą t ($1 \leqslant t \leqslant 1000$) oznaczającą liczbę zestawów testowych opisanych w dalszej części wejścia. Opis zestawu testowego składa się z jednego wiersza zawierającego trzy liczby całkowite p, q i n ($1 \leqslant p, q, n \leqslant 10^9$).

Wyjście

Na wyjściu powinno znaleźć się dokładnie t wierszy zawierających odpowiedzi do kolejnych zestawów testowych z wejścia. Odpowiedź powinna być jedną literą `E` (jeśli Euklides może doprowadzić do swojego zwycięstwa, niezależnie od ruchów Pitagorasa), `P` (jeśli Pitagoras może doprowadzić do swojego zwycięstwa, niezależnie od ruchów Euklidesa) lub `R` (jeśli gra będzie się toczyć w nieskończoność).

Przykład

Dla danych wejściowych:

```
4
3 2 1
2 3 1
3 4 5
2 4 3
```

poprawnym wynikiem jest:

```
P
P
E
R
```

Wyjaśnienie do przykładu: W rozgrywce z pierwszego zestawu testowego Euklides w swoim ruchu musi dołożyć na stos 3 kamienie. Dzięki temu Pitagoras może zabrać w swoim ruchu wszystkie 4 kamienie i tym samym wygrać.

Filary

Dostępna pamięć: 256 MB.

Bajtazar jest administratorem dużej hali magazynowej. Przewidując ciężką zimę, postanowił zamontować w hali ogrzewanie podłogowe.

Plan hali jest prostokątem o parzystych wymiarach $n \times m$ podzielonym na kwadraty jednostkowe. Większość kwadratów jednostkowych stanowi powierzchnię magazynową, jednak niektóre z nich są zajmowane przez masywne filary, które stanowią dodatkowe zabezpieczenie konstrukcyjne hali. Każdy filar na planie hali zajmuje kwadrat 2×2, złożony z kwadratów jednostkowych. Filary nie są rozmieszczone zbyt gęsto — wiadomo, że środki każdych dwóch z nich są oddalone co najmniej o 6 jednostek (w metryce euklidesowej). Ponadto środek każdego filara jest oddalony od każdej ściany zewnętrznej hali również co najmniej o 3 jednostki.

Ogrzewanie zostanie zrealizowane za pomocą jednej rury grzewczej zainstalowanej pod podłogą hali. Rura ma przebiegać przez środki wszystkich kwadratów jednostkowych z pominięciem kwadratów jednostkowych zajmowanych przez filary. Rura musi na każdym odcinku biec równolegle do którejś ze ścian hali i może zakręcać tylko w środkach kwadratów jednostkowych. Rura musi zaczynać się i kończyć w tym samym miejscu. W tym miejscu wychłodzona woda z rury będzie wyprowadzana na zewnątrz, a wprowadzana będzie woda gorąca.

Bajtazar poprosił Cię o zaplanowanie przebiegu rury pod halą. Aby Ci pomóc, na planie hali wprowadził prostokątny układ współrzędnych, w którym odcięte należą do przedziału $[0, n]$, a rzędne do przedziału $[0, m]$. Współrzędne środków wszystkich kwadratów jednostkowych na hali są liczbami postaci $k + \frac{1}{2}$ dla $k \in \mathbb{N}$.

Wejście

Pierwszy wiersz wejścia zawiera trzy liczby całkowite n, m oraz f ($1 \leqslant n, m \leqslant 1000$ oraz n i m są parzyste) oznaczające wymiary hali oraz liczbę filarów. Każdy z kolejnych f wierszy zawiera dwie liczby całkowite x_i i y_i ($0 \leqslant x_i \leqslant n$, $0 \leqslant y_i \leqslant m$) oznaczające współrzędne środka i-tego filara.

Wyjście

W pierwszym wierszu wyjścia Twój program powinien wypisać jedno słowo `TAK` lub `NIE`, w zależności od tego, czy jest możliwa realizacja ogrzewania podłogowego hali zgodnie z wymaganiami Bajtazara, czy też nie. Jeśli odpowiedzią jest `TAK`, w drugim wierszu powinien znaleźć się opis przykładowego planu przebiegu rury w postaci ciągu $nm - 4f$ liter. Zakładamy umownie, że początek rury znajduje się w punkcie o współrzędnych $(\frac{1}{2}, \frac{1}{2})$. Kolejne fragmenty rury są oznaczane następująco: przejście o wektor $[0, 1]$ oznaczamy literą `G`, o wektor $[0, -1]$ — literą `D`, o wektor $[1, 0]$ — literą `P`, a o wektor $[-1, 0]$ — literą `L`. Jeśli istnieje wiele poprawnych odpowiedzi, Twój program powinien wypisać dowolną z nich.

Przykład

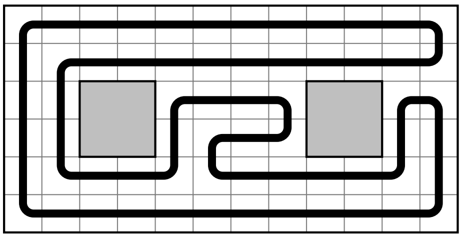
Dla danych wejściowych:

```
12 6 2
3 3
9 3
```

poprawnym wynikiem jest:

```
TAK
PPPPPPPPPPPGGGLDDLLLLLGPPGLLLDDLLLGGGPPPPPPPPPPGLLLLLLLLLLLDDDDD
```

Podane w przykładzie wyjście odpowiada poniższemu rysunkowi:



Globalne ocieplenie

Dostępna pamięć: 256 MB.

Profesor Bajtoni przygotowuje raport dla Międzybajtockiego Zespołu ds. Zmian Klimatu, z którego ma jednoznacznie wynikać, jaki jest wpływ mieszkańców Bajtocji na zmiany klimatyczne w regionie. Wprawdzie profesor ma sporo danych empirycznych, jednak aby przeniknąć do mediów głównego nurtu, nie wystarczą merytoryczne argumenty, ale równie ważne jest zaprezentowanie ich w dobitny sposób. W tym celu chce z rozmysłem wybrać dane, które przedstawi na głównym wykresie w raporcie.

Kluczowy wykres będzie zawierał informacje o średniej temperaturze powietrza na przestrzeni lat. Profesor dysponuje danymi dotyczącymi średniej rocznej temperatury dla ostatnich n lat. Chce okrasić ten wykres komentarzem w stylu „w roku r_{min} temperatura była najniższa, a w roku r_{max} była najwyższa, wobec tego jasno widać, że. . . ”. Niestety, obawia się, że taka sama minimalna lub maksymalna temperatura mogła wystąpić kilkukrotnie, co osłabiłoby dobitność tego zdania.

Profesor postanowił zatem zaprezentować na wykresie jedynie część danych. Chce teraz wybrać taki przedział lat, że *w tym przedziale* będzie dokładnie jeden rok z minimalną temperaturą w tym przedziale oraz dokładnie jeden rok z maksymalną temperaturą w tym przedziale. Wybrany przedział może nie zawierać roku z maksymalną lub minimalną średnią temperaturą w ciągu ostatnich n lat (lub żadnego z nich). Oczywiście profesor chciałby umieścić na wykresie jak najwięcej danych, więc interesuje go jak najdłuższy przedział lat.

Wejście

W pierwszym wierszu wejścia znajduje się jedna liczba całkowita n ($1 \leqslant n \leqslant 500\,000$), oznaczająca liczbę lat, dla których znamy średnie temperatury. W drugim wierszu znajduje się ciąg n liczb całkowitych $t_1, t_2, \ldots, t_n$ ($-10^9 \leqslant t_i \leqslant 10^9$). Liczba t_i oznacza średnią temperaturę w i-tym roku.

Wyjście

Na wyjście należy wypisać dwie liczby całkowite l i k. Oznaczają one, że najdłuższy przedział spełniający warunki profesora ma długość l lat, a najwcześniejszy rok, w którym taki przedział się zaczyna, to k.

Przykład

Dla danych wejściowych:

```
10
8 3 2 5 2 3 4 6 3 6
```

poprawnym wynikiem jest:

```
6 4
```

Wyjaśnienie do przykładu: Na wykresie zostaną przedstawione temperatury 5, 2, 3, 4, 6, 3. W tym przedziale jest dokładnie jeden rok z minimalną temperaturą 2 i jeden rok z maksymalną temperaturą 6.

Hit sezonu

Dostępna pamięć: 256 MB.

Bajtocki Zakład Poligraficzny (BZP) otrzymał duże zlecenie na produkcję prążkowanych tapet, stanowiących hit sezonu w projektowaniu wnętrz. Każda tapeta składa się z n jednakowej szerokości pionowych pasków w kolorach: czerwonym, zielonym i niebieskim. BZP ma zająć się zaprojektowaniem oraz wydrukowaniem tapet. Zleceniodawca określił barwy niektórych pasków na tapecie, natomiast w przypadku pozostałych pasków pozostawił BZP pełną dowolność.

Do wydruku tapet w BZP używa się matryc drukujących pewną liczbę kolejnych pasków na tapecie. Matryca ma określone barwy każdego z drukowanych pasków i może być krótsza niż cała tapeta. Przy każdym przyłożeniu matrycy jej paski muszą pokrywać się z paskami tapety; drukują się wówczas wszystkie paski matrycy. W ten sposób jeden pasek tapety może zostać zadrukowany więcej niż raz. W przypadku, gdy dany pasek zostanie zadrukowany różnymi barwami, jego ostateczny kolor będzie stanowił mieszankę tych barw. Matryca działa w tylko jednej orientacji i nie wolno jej w żaden sposób obracać.

Pracownicy BZP, niezależnie od posiadanego wyczucia estetyki, chcieliby przede wszystkim zaprojektować możliwie najkrótszą matrycę, która pozwoli wydrukować całą tapetę. Muszą oni pamiętać o tym, że w przypadku pasków określonych przez zleceniodawcę muszą użyć czystej barwy, bez domieszki innych barw. Innymi słowy, przy każdym przyłożeniu matrycy pokrywającym taki pasek, barwa paska na matrycy musi być dokładnie taka, jak określona przez zleceniodawcę. Żaden pasek tapety nie może pozostać bezbarwny.

Wejście

W pierwszym wierszu wejścia znajduje się jedna liczba całkowita t ($1 \leqslant t \leqslant 10$) oznaczająca liczbę zestawów testowych. Każdy z kolejnych t wierszy wejścia opisuje jeden zestaw testowy i zawiera napis złożony z wielkich liter `R`, `G`, `B` oraz gwiazdek (`*`), określający oczekiwany wygląd tapety. Poszczególne litery oznaczają barwy pasków, natomiast gwiazdki oznaczają paski, których barwa nie została określona przez zleceniodawcę. Napis jest niepusty, składa się z co najwyżej 3000 znaków i zawiera co najwyżej 19 gwiazdek.

Wyjście

Twój program powinien wypisać dla każdego zestawu testowego jeden wiersz zawierający jeden napis złożony z liter `R`, `G`, `B`: matrycę o minimalnej długości, która pozwala wydrukować żądaną tapetę. Jeśli dla danego zestawu testowego istnieje wiele poprawnych rozwiązań, Twój program powinien wypisać dowolne z nich.

Przykład

Dla danych wejściowych:

```
1
RRG*R*BRR**B
```

poprawnym wynikiem jest:

```
RRGB
```

Inscenizacja

Dostępna pamięć: 256 MB.

Stefan Beitberg jest reżyserem kina akcji. W ostatnim czasie pracuje nad nowym filmem, którego tematem są porachunki bajtockich mafii. Beitberg zastanawia się, jaki powinien być przebieg kulminacyjnej sceny filmu, w której odbędzie się widowiskowa strzelanina.

W scenie uczestniczy n gangsterów, ponumerowanych dla uproszczenia kolejnymi liczbami od 1 do n. Gdy napięcie sięga zenitu, każdy z gangsterów wyciąga swoją broń i wymierza ją w kierunku pewnego innego gangstera. Żadnych dwóch uczestników sceny nie mierzy do tego samego gangstera. Gangsterzy są biedni, lecz dobrze wyszkoleni — każdy z nich może oddać co najwyżej jeden, ale zawsze celny i śmiertelny, strzał.

W pewnym momencie któryś z bandziorów nie wytrzymuje napięcia i rozpoczyna się strzelanina.

Reżyser ustalił pewną początkową kolejność, w jakiej gangsterzy mają pociągać za spust. Mianowicie, gangster i strzela w kierunku gangstera p_i dokładnie w momencie t_i, chyba że do tego czasu gangster i został już zabity. Przyjmujemy, że gangster ginie dokładnie w chwili, gdy ktoś oddaje strzał w jego kierunku.

Reżyser chciałby wiedzieć, ilu gangsterów zostanie przy życiu pod koniec sceny. Beitberg nie jest jeszcze całkowicie pewien co do kolejności, w jakiej gangsterzy mają strzelać. Co pewien czas wydaje polecenie, aby zmienić jedną z wartości t_i. Po każdej takiej zmianie chciałby znać liczbę gangsterów, którzy przeżyją, dla nowej kolejności oddawania strzałów (przy uwzględnieniu wszystkich dotychczas wykonanych zmian).

Wejście

W pierwszym wierszu wejścia znajduje się jedna liczba całkowita n ($2 \leqslant n \leqslant 200\,000$), oznaczająca liczbę gangsterów biorących udział w scenie. W drugim wierszu znajduje się n liczb całkowitych $p_1, p_2, \ldots, p_n$ ($1 \leqslant p_i \leqslant n$, $p_i \neq i$, $p_i \neq p_j$ dla $i \neq j$), określających, do kogo zamierzają strzelać kolejni gangsterzy.

W trzecim wierszu znajduje się n liczb całkowitych $u_1, u_2, \ldots, u_n$ ($1 \leqslant u_i \leqslant 10^9$), opisujących początkową kolejność oddawania strzałów przez gangsterów: początkowa wartość t_i jest równa u_i.

W czwartym wierszu znajduje się jedna liczba całkowita q ($0 \leqslant q \leqslant 200\,000$), oznaczająca liczbę zmian wartości $t_1, \ldots, t_n$ planowanych przez Beitberga. Kolejne q wierszy to opis tych zmian. W i-tym z nich znajdują się dwie liczby całkowite k_i i v_i ($1 \leqslant k_i \leqslant n, 1 \leqslant v_i \leqslant 10^9$), oznaczające, że i-ta zmiana polega na ustawieniu wartości t_{k_i} na v_i. Liczby $u_1, u_2, \ldots, u_n, v_1, v_2, \ldots, v_q$ są parami różne.

Wyjście

Twój program powinien wypisać na wyjście dokładnie $q+1$ wierszy. W pierwszym wierszu powinna znaleźć się liczba gangsterów, którzy przeżyją strzelaninę, zakładając początkową kolejność strzelania. W i-tym z q kolejnych wierszy należy wypisać liczbę gangsterów pozostałych przy życiu, przy założeniu, że kolejność strzelania określa ciąg $t_1, \ldots, t_n$ po dokonaniu wszystkich zmian od pierwszej do i-tej.

Przykład

Dla danych wejściowych:

```
4
2 3 4 1
1 2 3 4
3
1 8
2 7
3 6
```

poprawnym wynikiem jest:

```
2
2
1
1
```

Jaskinia

Dostępna pamięć: 256 MB.

Grupa speleologów planuje zbadać odkrytą niedawno jaskinię. Jaskinia składa się z n komnat ponumerowanych od 1 do n. Komnaty są połączone za pomocą $n-1$ korytarzy w taki sposób, że z dowolnej komnaty można przejść do dowolnej innej. Każdy korytarz łączy dokładnie dwie komnaty.

Badanie jaskini przeprowadzi grupa m speleologów, których dla uproszczenia ponumerujemy od 1 do m. Każdy speleolog przedstawił wymagania dotyczące obszaru jaskini, który będzie badać. Speleolog i chciałby rozpocząć badanie w komnacie a_i, zakończyć je w komnacie b_i, a po drodze przemierzyć co najwyżej d_i korytarzy (każde przebycie tego samego korytarza liczymy osobno). Bajtazar, kierownik wyprawy, chciałby, by w pewnym momencie wszyscy badacze mogli spotkać się i wymienić swoimi spostrzeżeniami. Z tego powodu zastanawia się, czy może wybrać jedną z komnat jaskini i tak wytyczyć trasy speleologów, by wszystkie prowadziły przez wybraną komnatę. Oczywiście wytyczone trasy muszą spełniać wymagania postawione przez badaczy.

Wejście

W pierwszym wierszu wejścia znajduje się jedna liczba całkowita t ($1 \leqslant t \leqslant 1000$) określająca liczbę zestawów testowych. Dalej następują opisy poszczególnych zestawów. Opis jednego zestawu rozpoczyna się od wiersza z dwiema liczbami całkowitymi n i m ($2 \leqslant n, m \leqslant 300\,000$), które opisują, odpowiednio, liczbę komnat w jaskini oraz liczbę speleologów. W kolejnych $n-1$ wierszach opisane są korytarze jaskini. Każdy z nich zawiera dwie liczby całkowite u_i, w_i ($1 \leqslant u_i, w_i \leqslant n$), które oznaczają, że komnaty u_i oraz w_i są bezpośrednio połączone korytarzem.

Następne m wierszy opisuje wymagania speleologów. W i-tym z tych wierszy nich znajdują się trzy liczby całkowite a_i, b_i, d_i ($1 \leqslant a_i, b_i \leqslant n$, $1 \leqslant d_i \leqslant 600\,000$). Oznaczają one, że speleolog i rozpocznie badanie w komnacie a_i, zakończy je w komnacie b_i, a po drodze co najwyżej d_i razy przejdzie korytarzem. Możesz założyć, że da się przejść z komnaty a_i do komnaty b_i przemierzając nie więcej niż d_i korytarzy. Zarówno suma wartości n po wszystkich zestawach testowych, jak i suma wartości m nie przekraczają 300 000.

Wyjście

Twój program powinien wypisać na wyjście dokładnie t wierszy. W i-tym wierszu powinna znaleźć się odpowiedź dla i-tego zestawu testowego z wejścia. Jeśli da się tak poprowadzić trasy speleologów, by wszystkie przebiegały przez jedną komnatę, należy wypisać `TAK`, a następnie numer komnaty, w której może dojść do spotkania. W przeciwnym razie, należy wypisać jedynie słowo `NIE`. Jeśli istnieje wiele poprawnych odpowiedzi, Twój program powinien wypisać dowolną z nich.

Przykład

Dla danych wejściowych:

```
2
5 3
1 2
2 3
2 4
3 5
1 4 2
5 5 5
3 2 1
3 2
1 2
2 3
1 1 2
3 3 1
```

poprawnym wynikiem jest:

```
TAK 2
NIE
```

Kapitan

Dostępna pamięć: 256 MB.

Kapitan Bajtazar przemierza wody Morza Bajtockiego wraz ze swoim niezastąpionym pierwszym oficerem Bajtkiem. Na morzu znajduje się n wysp, które numerujemy liczbami od 1 do n. Przy wyspie numer 1 przycumował statek kapitana. W ramach wyprawy kapitan planuje popłynąć na wyspę numer n.

W trakcie rejsu statek zawsze porusza się w jednym z czterech kierunków świata: na północ, południe, wschód lub zachód. W każdym momencie przy sterach stoi albo kapitan, albo pierwszy oficer. Za każdym razem, gdy statek wykona skręt o $90°$, zmieniają się oni przy sterach.

Po drodze statek może zatrzymywać się przy innych wyspach. Po każdym postoju kapitan może zdecydować, czy obejmuje stery jako pierwszy. Innymi słowy, na każdym fragmencie trasy prowadzącym z wyspy do wyspy jeden z marynarzy obejmuje stery, gdy statek płynie na północ lub południe, a drugi z nich steruje podczas rejsu na wschód lub zachód. W szczególności, jeśli pewien fragment trasy prowadzi dokładnie w jednym z czterech kierunków świata, na tym fragmencie steruje tylko jeden z marynarzy.

Kapitan zastanawia się teraz, jak zaplanować trasę najbliższego rejsu i podział pracy, by spędzić jak najmniej czasu przy sterze. Jednocześnie kapitan nie przejmuje się, jak długa będzie wyznaczona trasa. Przyjmujemy, że statek płynie ze stałą prędkością jednej jednostki na godzinę.

Wejście

Pierwszy wiersz wejścia zawiera jedną liczbę całkowitą n ($2 \leqslant n \leqslant 200\,000$) oznaczającą liczbę wysp na morzu. Dla uproszczenia na Morze Bajtockie nanosimy układ współrzędnych, którego osie są równoległe do kierunków świata. Każdą z wysp reprezentujemy jako pojedynczy punkt. Kolejne n wierszy zawiera opisy wysp: i-ty z nich zawiera dwie liczby całkowite x_i, y_i ($0 \leqslant x_i, y_i \leqslant 1\,000\,000\,000$) oznaczające współrzędne i-tej wyspy na morzu. Każda wyspa ma inne współrzędne.

Wyjście

Twój program powinien wypisać na wyjście jedną liczbę całkowitą, oznaczającą najmniejszą liczbę godzin, przez które kapitan będzie musiał sterować statkiem na trasie z wyspy numer 1 do wyspy numer n.

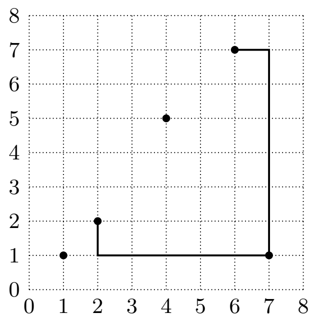
Przykład

Dla danych wejściowych:

```
5
2 2
1 1
4 5
7 1
6 7
```

poprawnym wynikiem jest:

```
2
```



Wyjaśnienie do przykładu: Kapitan może wyznaczyć trasę, którą zaznaczono na obrazku. W trakcie rejsu z wyspy 1 (współrzędne $(2, 2)$) na wyspę 4 (współrzędne $(7, 1)$) kapitan steruje tylko przez godzinę, gdy statek płynie na południe. W trakcie drugiego fragmentu podróży kapitan steruje jedynie wtedy, gdy statek porusza się na wschód.