



Jednosměrky (One-Way Streets)

Časový limit: 3 s Paměťový limit: 256 MB

Byla nebyla země s n městy a m obousměrnými silnicemi spojujícími tato města. Technický rozvoj vedl k rychlejšímu a větším vozidlům, což způsobilo problém – silnice se pro dvě vozidla jedoucí v opačném směru stávaly postupně příliš úzkými. Rozhodnutí vyřešit tento problém zahrnovalo změnu všech silnic na jednosměrné s jedním pruhem.

Přeměna silnic na jednosměrky si vybere svoji daň, protože některé dvojice měst, které byly původně propojené, by už po této změně nemusely být dosažitelné. Vláda sestavila seznam důležitých dvojic měst, pro které musí být možné začít v prvním městě a dorazit do druhého. Váš úkol je rozhodnout, kterým směrem vést dopravu na každé silnici. Je zaručeno, že řešení existuje.

Pokud chcete získat řešení, u některých silnic si nemůžete vybrat směr dopravy. Doprava bude vedena z prvního města do druhého (směr doprava, označený písmenem R) nebo z druhého města směrem k prvnímu (směr doleva, označený písmenem L). Pro některé silnice nicméně existuje řešení, ve kterém je doprava na této silnici směřována doleva, a další (možná odlišné) řešení, kde je doprava směřována doprava. Takové silnice byste měli označit písmenem B (both directions – oba směry).

Vypište řetězec délky m . Jeho i -tý znak by měl být

- R, pokud všechna řešení vyžadují, aby doprava na i -té cestě byla směřována doprava
- L, pokud všechna řešení vyžadují, aby doprava na i -té cestě byla směřována doleva
- B, pokud existuje řešení, v němž je doprava na i -té cestě směřována doleva, a také existuje řešení, v němž je směřována doprava

Vstup

První řádek obsahuje počet měst n a počet silnic m . Následujících m řádků popisuje silnice pomocí dvojic čísel a_i a b_i , tato dvojice značí, že mezi městy a_i a b_i existuje silnice. Mezi stejnou dvojicí měst může být více než jedna silnice, a silnice může dokonce spojovala město se sebou samým.

Následující řádek obsahuje počet dvojic měst p , které musí být dostupné. Následujících p řádků obsahuje dvojice měst x_i a y_i , taková dvojice znamená, že musí existovat způsob, jak začít v městě x_i a skončit v městě y_i .

Omezení

- $1 \leq n, m, p \leq 100\,000$
- $1 \leq a_i, b_i, x_i, y_i \leq n$

Podúloha 1 (30 bodů)

- $n, m \leq 1000$
- $p \leq 100$



Podúloha 2 (30 bodů)

- $p \leq 100$

Podúloha 3 (40 bodů)

- bez dalších omezení

Výstup

Vypište řetězec délky m tak, jak bylo popsáno v popisu úlohy.

Příklad

Vstup

```
5 6
1 2
1 2
4 3
2 3
1 3
5 1
2
4 5
1 3
```

Výstup

```
BBRBBL
```

Poznámka

Ukažme, že pátá silnice "1 3" může být směřována oběma způsoby. Dvě možné orientace cest s jiným směřováním (jinou orientací) páté cesty jsou LLRLRL a RLRRL.



Sázka na jistotu (Sure Bet)

Časový limit: 2 s Paměťový limit: 128 MB

Štěstí je nedílnou součástí sázení. Někteří lidé zlepšují své šance a příjmy tím, že dobře znají to, na co sází. My použijeme odlišný přístup.

Rozličné sázkové kanceláře nabízí pro stejný výsledek rozdílné *kurzy* či limity. (*Kurz* x znamená, že pokud vsadíte 1 euro a výsledek předpovíte správně, dostanete x euro. Pokud výsledek předpovíte špatně, nedostanete samozřejmě nic. Všimněte si, že 1 euro zaplatíte nezávisle na výsledku.) Co ale, kdybyste si mohli zajistit určitý zisk tím, že chytře uzavřete více sázek? Chtěli byste, aby tento zaručený zisk byl co možná největší.

Událost, na kterou chceme sázet, má dva možné výsledky. Existuje n sázkových kanceláří, každá nabízí nějaké kurzy. Označme kurzy nabízené i -tou sázkovou kanceláří pro první výsledek jako a_i a kurzy pro druhý výsledek jako b_i . Sázet můžete na libovolnou podmnožinu všech nabízených kurzů. Dokonce si můžete u stejné sázkové kanceláře vsadit na oba výsledky. Každá sázka nicméně musí být přesně ve výši 1 euro a u jedné sázkové kanceláře nelze na stejný výsledek sázet vícekrát.

V případě, že nastane první z výsledků, dostanete a_i euro od každé sázkové kanceláře i , u které jste si na první výsledek vsadili. Obdobně to je v případě druhého výsledku, dostanete b_i euro od odpovídajících sázkových kanceláří. V obou případech jste již samozřejmě zaplatili 1 euro za každou uzavřenou sázku.

Jaký je největší *zaručený* zisk (tedy nezávislý na výsledku), pokud budete sázet optimálně?

Vstup

První řádek obsahuje počet sázkových kanceláří, n . Následujících n řádků popisuje kurzy nabízené jednotlivými sázkovými kancelářemi, a to jako dvě mezerou oddělená reálná čísla a_i a b_i – kurzy pro první a druhý výsledek nabízené i -tou sázkovou kanceláří. Kurzy budou zadané s přesností na nejvýše 4 desetinná místa.

Omezení

- $1.0 \leq a_i, b_i \leq 1000.0$
- $1 \leq n \leq 100\,000$

Podúloha 1 (20 bodů)

- $n \leq 10$

Podúloha 2 (40 bodů)

- $n \leq 1\,000$

Podúloha 3 (40 bodů)

- bez dalších omezení



Výstup

Vypište maximální zaručený zisk zaokrouhlený na přesně 4 desetinná místa.

Následují příkazy, které vypíší desetinné číslo (floating point) v různých programovacích jazycích:

- C a C++: `printf("%.4lf", (double)x);`
- Java: `System.out.printf("%.4lf", x);`
- Pascal: `writeln(x:0:4);`
- Python 3: `print("%.4lf"%x)`
- C#: `Console.WriteLine(String.Format("0:0.0000", x));`

Příklad

Vstup

4
1.4 3.7
1.2 2
1.6 1.4
1.9 1.5

Výstup

0.5000

Poznámka

Optimální je si vsadit u první sázkové kanceláře na druhý výsledek a u třetí a čtvrté kanceláře na první výsledek. Získáme tak v případě prvního výsledku $1.6 + 1.9 - 3 = 0.5$ a v případě druhého výsledku $3.7 - 3 = 0.7$. Takže máme zaručený zisk 0.5 euro nezávisle na výsledku.



Past na myš (Mousetrap)

Časový limit: 5 s Paměťový limit: 512 MB

Slon Dumbo má veliký labyrint s n místnostmi očíslovanými $1 \dots n$ a $n - 1$ chodbami takovými, že z každé místnosti je možné dostat se do všech ostatních. Do labyrintu ale bohužel proklouzla myš. Dumbo se myši hrozně bojí, a tak na ni v místnosti t připravil past. Myš se samozřejmě místnosti s pastí vyhýbá, a tak musí Dumbo vymyslet lepší plán, aby ji do pasti nalákal. Myš pořád někde pobíhá a nikdy se nezastaví, dokud nenastane situace, že se nemá kam pohnout. Dumbo také ví, že myš nechává špinavou stopu v každé chodbě, kterou proběhne. Myš pak odmítá špinavou chodbu znovu použít. Dumbo může zablokovat chodbu kameny nebo špinavou chodbu vyčistit. Blokováním chodeb a jejich čištěním chce myš donutit vběhnout do místnosti s pastí. To by chtěl udělat na co nejméně pohybů, neb se v přítomnosti myši cítí velmi nepohodlně.

Tohle celé můžeme popsat jako hru dvou hráčů. Myš se snaží maximalizovat počet Dumbových pohybů. Dumbo se snaží vyhrát na minimální počet pohybů. Začínající hráč je Dumbo. Ve svém kole může vyčistit jednu špinavou chodbu labyrintu nebo jednu chodbu zablokovat. Nezáleží na tom, zda je zablokována chodba čistá, nebo ne. Chodbu nemůže odblokovat. Může se ale rozhodnout neudělat nic. Kola, ve kterých se Dumbo rozhodne nic nedělat, se nepočítají jako pohyby. Když je na tahu myš, vybere si čistou nezablokovanou chodbu a proběhne jí do sousední místnosti. Pokud taková chodba z místnosti, kde se myš právě nachází, nevede, myš se nepohne.

Na počátku jsem všechny chodby čisté, myš je v místnosti m , past je v místnosti t a na tahu je Dumbo. Jaký je minimální počet pohybů (čištění a blokování chodeb), které Dumbo potřebuje, pokud oba hráči hrají optimálně (cílem myši je maximalizovat počet Dumbových pohybů)?

Vstup

Na prvním řádku budou zadaná celá čísla n , t a m oddělená mezerami. Následuje $n - 1$ řádků. Na každé řádce jsou zadána a_i a b_i , oddělená mezerou; tato čísla udávají, že mezi místnostmi a_i a b_i vede chodba.

Všimněte si, že vstup je velký.

Omezení

- $1 \leq n, t, m \leq 10^6$

Podúloha 1 (20 bodů)

- $n \leq 10$

Podúloha 2 (25 bodů)

- Je zaručeno, že existuje chodba mezi místnostmi m a t .

Podúloha 3 (20 bodů)

- $n \leq 1000$



Podúloha 4 (35 bodů)

- bez dalších omezení

Výstup

Váš program by měl vypsat počet Dumbových pohybů.

Příklad

Vstup

```
10 1 4
1 2
2 3
2 4
3 9
3 5
4 7
4 6
6 8
7 10
```

Výstup

```
4
```

Poznámka

Jeden možný scénář:

- Dumbo zablokuje chodbu mezi místnostmi 4 a 7.
- Myš se pohne do místnosti 6. Chodba mezi místnostmi 4 a 6 je teď špinavá.
- Dumbo zablokuje chodbu mezi místnostmi 6 a 8.
- Myš se nemůže pohnout.
- Dumbo vyčistí chodbu mezi místnostmi 4 a 6.
- Myš se pohne do místnosti 4. Chodba mezi místnostmi 4 a 6 je špinavá.
- Dumbo zablokuje chodbu mezi místnostmi 2 a 3.
- Myš se pohne do místnosti 2. Chodba mezi místnostmi 2 a 4 je špinavá.
- Dumbo nic neudělá.
- Myš se může pohnout pouze do místnosti 1 a chytí se do pasti.

Dumbo udělal 4 pohyby.