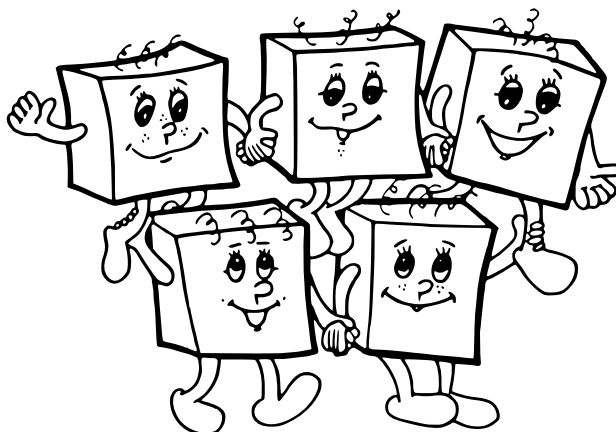


OLYMPIÁDA V INFORMATIKE

NA STREDNÝCH ŠKOLÁCH

<http://oi.sk/>



dvadsiaty piaty ročník

školský rok 2009/10

zadania krajského kola

kategória B

Priebeh krajského kola

Krajské kolo 25. ročníka Olympiády v informatike, kategória B, sa koná 12. januára 2010 v dopoludňajších hodinách. Na riešenie úloh majú súťažiaci 4 hodiny čistého času. Rôzne úlohy riešia súťažiaci na samostatné listy papiera. Akékoľvek pomôcky okrem písacích potrieb (napr. knihy, výpisy programov, kalkulačky) sú zakázané.

Čo má obsahovať riešenie úlohy?

- Slovné popíšte algoritmus.
Slovný popis riešenia musí byť jasný a zrozumiteľný i bez nahliadnutia do samotného algoritmu/programu.
- Zdôvodnite správnosť vášho algoritmu.
- Uveďte a zdôvodnite jeho časovú a pamäťovú zložitosť.
- Podrobne uveďte dôležité časti algoritmu, ideálne vo forme programu v Pascale alebo C/C++.
- V prípade, že používate vo svojom programovacom jazyku knižnice, ktoré obsahujú implementované dátové štruktúry a algoritmy (napr. STL pre C++), v popise algoritmu stručne vysvetlite, ako by ste napísali program s rovnakou časovou zložitou bez použitia knižnice.

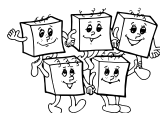
Hodnotenie riešení

Za každú úlohu môžete získať od 0 do 10 bodov.

Pokiaľ nie je v zadaní povedané ináč, základným kritériom hodnotenia riešenia je **správnosť** a **efektívnosť** navrhnutého algoritmu. Hodnotí sa aj kvalita popisu riešenia a zdôvodnenie tvrdení o jeho správnosti a efektívnosti.

Efektívnosť algoritmu posudzujeme vypočítaním jeho časovej zložitosti – funkcie, ktorá hovorí, ako dlho vykonanie algoritmu trvá v závislosti od veľkosti vstupných parametrov.

V zadaní úlohy môžu byť uvedené limity na veľkosť premenných. Tieto môžete použiť na odhad toho, ako dobré vaše riešenie je. Na počítači, ktorý vykoná miliardu inštrukcií za sekundu, zrieši vzorové riešenie ľubovoľný povolený vstup do niekoľko sekúnd.



B-I-1 Telemánia

Nemenovaný mobilný operátor by chcel v rámci série vianočných prekvapení udeliť špeciálny bonus tomu svojmu zákazníkovi, ktorý v poslednom roku telefonoval najviac. Vašou úlohou bude napísať pre tohto mobilného operátora program, ktorý dotyčného zákazníka nájde.

K dispozícii máte dáta uložené v dvoch tabuľkách. V prvej tabuľke sú telefónne čísla a mená všetkých zákazníkov nášho operátora, druhá tabuľka obsahuje zoznam všetkých hovorov za posledný rok, v ktorých aspoň jeden z účastníkov (t. j. volajúci, volaný, alebo obaja) bol zákazníkom nášho operátora.

Súťažná úloha

Napíšte program, ktorý načíta dáta o zákazníkoch a hovoroch za posledný rok a vypíše meno zákazníka nášho mobilného operátora, ktorý za posledný rok v súčte telefonoval najdlhší čas.

Formát vstupu

Prvý riadok vstupu obsahuje jedno celé číslo N ($1 \leq N \leq 1\,000\,000$) – počet všetkých zákazníkov nášho mobilného operátora. Každý z nasledujúcich N riadkov obsahuje telefónne číslo a meno jedného zákazníka. Môžete predpokladať, že telefónne číslo bude obsahovať presne 10 cifier a meno bude obsahovať najviac 200 znakov. Pre jednoduchosť môžete tiež predpokladať, že meno neobsahuje medzery.

Ďalej nasleduje jeden riadok, obsahujúci jedno celé číslo M ($1 \leq M \leq 1\,000\,000$) – počet zaznamenaných hovorov. Každý z nasledujúcich M riadkov obsahuje dve telefónne čísla oddelené medzerou, za ktorými nasleduje jedno celé číslo určujúce dĺžku hovoru v sekundách.

Pozor, telefónne čísla v tabuľke hovorov nemusia byť nutne len zákazníci nášho operátora!

Formát výstupu

Výstup má obsahovať jediný riadok a v ňom telefónne číslo a meno nášho zákazníka, ktorý podľa záznamov hovorov dohromady telefonoval najdlhší čas.

Ak je takýchto zákazníkov viac, stačí vypísať ľubovoľného jedného z nich.

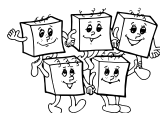
Príklady

vstup

```
3
0972125726 Jozko
0975342583 Ferko
0972654124 Jano
5
0975342583 0972654124 80
0972125726 0955956114 137
0955956114 0975342583 293
0972125726 0972654124 36
0972125726 0975342583 54
```

výstup

```
0975342583 Ferko
Ferko dohromady telefonoval 427, Jozko 227 a Jano
116 sekúnd.
Všimnite si, že medzi hovormi sa tiež nachádza číslo
0955956114, na ktoré sa uskutočnilo až 430 sekúnd
hovorov. Toto číslo ale nepatrí žiadnemu z našich zá-
kazníkov, preto nemôže vyhrať.
```



B-II-2 Kráľovské cesty

Santoland je nádherná krajina, ale má veľmi zlú infraštruktúru. Telefóny ledva fungujú, o internete môžu len snívať a cesty sú v celej krajine len tri.

Každá z týchto troch ciest začína v hlavnom meste a žiadne dve sa už potom nikde nekrižujú. Každé mesto v Santolande (okrem hlavného) leží na práve jednej z týchto ciest. V každom meste vedia obyvatelia, cez ktoré susedné mesto sa dostanú do hlavného mesta, a taktiež vedia vzdialenosť do tohto mesta v kilometroch.

Kráľ Banto sa rozhodol, že v rámci podpory turistického ruchu zriadi telefonickú linku, na ktorú bude môcť ktokoľvek zavolať a spýtať sa na vzdialenosť ľubovoľných dvoch miest.

Ako už však odznelo, telefóny ledva fungujú a hovor nikdy nevydrží dlhšie ako minútu bez prerušenia. Preto je potrebné napísať program, ktorý bude na linke odpovedať na otázky volajúcich *veľmi* efektívne.

Súťažná úloha

Vašou úlohou bude napísať program, ktorý načíta popis krajiny, prípadne ho nejak spracuje, a potom bude čo najrýchlejšie spracovávať jednotlivé telefonáty.

Popis krajiny sa nachádza v textovom súbore `mapa.txt`.

V prvom riadku súboru je celé číslo N , ktoré označuje počet miest v Santolade. Hlavné mesto má číslo 1 a ostatné mestá majú čísla $2, \dots, N$, pričom čísla miest nesúvisia s ich umiestnením na cestách.

Druhý až N -tý riadok popisujú mestá 2 až N . Riadok popisujúci mesto i obsahuje dve kladné celé čísla m_i a d_i , kde m_i je číslo najbližšieho mesta na ceste z mesta i do hlavného mesta (mesta 1) a d_i je dĺžka cesty medzi mestami i a m_i .

Môžete predpokladať, že vstup je korektný, teda medzi hodnotami m_x sa číslo 1 nachádza práve $3 \times$ (do hlavného mesta vedú 3 cesty), každé iné číslo sa tam nachádza najviac raz a z každého mesta sa dá dostať do hlavného.

Po tom, ako váš program načíta a spracuje tieto údaje, musí fungovať v nekonečnom cykle, v ktorom striedavo načíta zo štandardného vstupu dve čísla miest a a b a vypíše vzdialenosť $d_{a,b}$ medzi týmito dvomi mestami.

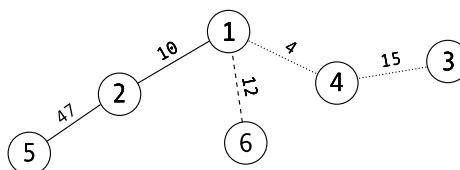
Bodovanie

Prvoradým kritériom hodnotenia riešení bude efektívnosť spracovania jednotlivých telefonátov. (Následne prihliadneme aj k časovej zložitosti predspracovania a pamätovej zložitosti vášho programu.)

Príklad

`mapa.txt`

```
6
1 10
4 15
1 4
2 47
1 12
```



Obrázok znázorňuje krajinu popísanú v súbore `mapa.txt`. Napr. keďže 5. riadok vstupu je `2 47`, tak vieme, že z mesta 5 vedie do mesta 2 cesta dĺžky 47. Každá cesta z hlavného mesta je kreslená iným typom čiary.

vstup

```
5 2
1 6
1 3
5 3
5 6
...
```

výstup

```
47
12
19
76
69
...
```



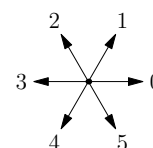
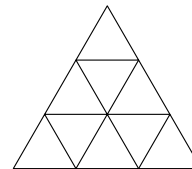
B-II-3 Triplex

Hrací plán pre hru Triplex má tvar rovnostranného trojuholníka so stranou dĺžky N , ktorý je rozdelený na jednotkové trojuholníky. Príklad hracieho plánu pre $N = 3$ vidíte na prvom obrázku vpravo.

Slovom *vrcholy* označujeme všetky vrcholy všetkých jednotkových trojuholníkov tvoriacich hrací plán. Slovom *hrany* označujeme všetky ich strany. Hrací plán na obrázku má teda 10 vrcholov a 18 hrán.

Na hracom pláne sa nachádza jedna figúrka. Na začiatku hry je táto figúrka v „hornom“ vrchole trojuholníka. Existuje 6 smerov, ktorými môžeme figúrkou hýbať. Tieto smery si očísľujeme tak, ako je znázornené na druhom obrázku vpravo.

Figúrka nesmie pri svojom pohybe opustiť hrací plán. Napr. v prvom ťahu sa teda musí pohnúť buď smerom 4, alebo smerom 5.



Súťažná úloha

- (5 bodov) Napíšte program, ktorý načíta hodnotu N a vypíše postupnosť ťahov figúrkou, pri ktorej figúrka navštívi práve raz každý vrchol a na konci sa vráti do vrcholu, kde začínala.
- (5 bodov) Napíšte program, ktorý načíta hodnotu N a vypíše postupnosť ťahov figúrkou, pri ktorej figúrka prejde práve raz po každej hrane a na konci sa vráti do vrcholu, kde začínala.

Riešenie každej podúlohy bude hodnotené samostatne, môžete riešiť podúlohu b) aj ak neviete riešiť a).

Formát vstupu

Na vstupe je najskôr písmeno (A alebo B), udávajúce, ktorú podúlohu má program riešiť, a potom kladné celé číslo N .

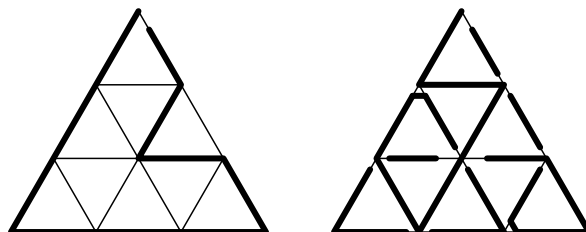
Formát výstupu

Výstup má obsahovať jediný riadok a v ňom postupnosť číier 0 až 5 popisujúcich jednotlivé ťahy. Ak existuje viacero riešení, stačí vypísať jedno ľubovoľné.

Príklady

| vstup | výstup |
|-------|--------------------|
| A 3 | 4440002312 |
| vstup | výstup |
| B 3 | 404421553310053122 |

Na nasledujúcich obrázkoch sú znázornené postupnosti ťahov zodpovedajúce vstupom a výstupom z príkladu.



Čiara na obrázku vpravo je miestami prerušovaná, aby bolo lepšie vidieť, ktoré jej časti na seba nadväzujú na miestach, kde križuje samú seba.



B-II-4 Gaštanový snehuliak

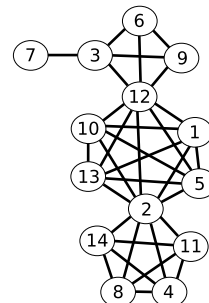
Pamätáte si zo zadania domáceho kola Zuzkinu gaštanovú žirafu? Keď prišla zima, Zuzka sa rozhodla rozobrať ju a namiesto nej postaviť z gaštanov objekt, ktorý sa k zime hodí omnoho viac – snehuliaka.

Gaštanový snehuliak sa, podobne ako klasický, skladá z troch guľí. Guľa sa z gaštanov vyrobí tak, že zoberieme niekoľko gaštanov (aspoň štyri) a každé dva spojíme záparkou.

Aby snehuliak dobre držal pokope, vyrobila ho Zuzka tak, že stredná guľa má práve jeden gaštan spoločný s hornou a práve jeden iný gaštan s dolnou guľou.

A aby bol snehuliak dokonalý, ešte dostal nos-mrkvu. Teda nos-gaštan: jeden gaštan, ktorý je záparkou spojený s niektorým jedným gaštanom v hornej guľi. (Nie s tým, ktorý patrí aj do strednej guľe.)

Príklad takéhoto snehuliaka nájdete na obrázku napravo. Horná guľa je tvorená štyrmi, stredná šiestimi a spodná piatimi gaštanmi. Spoločný gaštan pre hornú a strednú guľu má číslo 12, spoločný gaštan pre strednú a dolnú guľu má číslo 2. Gaštan predstavujúci nos má číslo 7.



Keď Zuzka dostávala snehuliaka, zavolala brata Tomáška, aby sa opäť zahrali jej obľúbenú hru. Pripomeňme si jej pravidlá: Na začiatku Zuzka oznámi Tomáškovi hodnotu N – celkový počet gaštanov, ktoré tvoria snehuliaka. Tieto gaštany sú nejako náhodne očíslované od 1 po N . Tomáškovým cieľom je o každom gaštane od 1 do N zistiť, ktorú časť snehuliaka predstavuje. Tomáško však snehuliaka nevidí, lebo ho Zuzka pred ním schovala. Môže sa jej len pýtať otázky tvaru „sú gaštany a a b spojené záparkou?“

Príklad hry pre snehuliaka z obrázka

Z: Počet gaštanov tvoriacich snehuliaka je 14.

T: Sú gaštany 1 a 2 spojené? Z: Áno.

T: Sú gaštany 1 a 3 spojené? Z: Nie.

T: Sú gaštany 2 a 4 spojené? Z: Áno.

...

T: Sú gaštany 7 a 3 spojené? Z: Áno.

T: Nos je gaštan 7, hornú guľu tvoria gaštany 3 6 9 12, strednú 1 2 5 10 12 13 a dolnú 2 4 8 11 14.

Súťažná úloha

Navrhňte čo najlepšiu stratégiu pre túto hru. Presnejšie, nájdite čo najefektívnejší algoritmus, ktorý bude túto hru hrať namiesto Tomáška a vždy vyhrá. Algoritmus stačí v riešení uviesť ako pseudokód (slovný popis), nie je potrebné implementovať ho v programovacom jazyku.

Na získanie 10 bodov potrebujete nájsť algoritmus, ktorého časová zložitosť bude lineárna od N . Za 7 bodov bude riešenie, ktorého časová zložitosť bude od N závisieť kvadraticky.

Pri odhade časovej zložitosti predpokladajte, že Zuzka dokáže na každú otázku odpovedať okamžite. Uvedomte si, že 10-bodové riešenie si ani zďaleka nemôže dovoliť opýtať sa Zuzky na všetky dvojice gaštanov.