

Domáca úloha č. 5

1-AIN-105, Zima 2023

Termín: 18.12.2023, 22:00 (google classroom)

Skôr ako sa pustíte do riešenia domácej úlohy, oboznámte sa so všeobecnými pokynmi, ktoré sú priložené na konci tohto dokumentu. Riešenia, ktoré odovzdáte, musia byť vaše vlastné. Neopisujte a nesnažte sa nájsť riešenia v literatúre alebo na internete!

1. [20 bodov] **Cesty z plánu obnovy.** Nová vláda sa rozhodla presmerovať peniaze z plánu obnovy do stavby ciest. Analytická skupina teraz potrebuje rýchlo spočítať indikátor, aký vplyv budú mať jednotlivé navrhované cesty na dopravnú obslužnosť. Indikátor dopravnej obslužnosti sa počíta tak, že sa spočíta dĺžka najkratšej cesty medzi každými dvoma mestami po pridaní novej cesty do cestnej siete a tieto dĺžky sa sčítajú pre každú dvojicu miest.

Úloha: Je daný počet miest n a pre každú dvojicu miest dĺžka $d_{i,j}$ existujúcej *priamej* cesty z mesta i do mesta j (ak medzi mestami i a j neexistuje priama cesta, tak $d_{i,j} = \infty$). Dalej je daných q navrhovaných nových ciest, kde i -ta navrhovaná cesta vedie z mesta k_i do mesta ℓ_i a má dĺžku e_i . Navrhujte algoritmus, ktorý pre každú navrhovanú cestu spočíta jej indikátor dopravnej obslužnosti. Všetky existujúce aj navrhované cesty sú obojsmerné.

Hint: Skúste popremýšľať nad riešením s časovou zložitou lepšou ako $O(q \cdot n^3)$.

2. [20 bodov] **Zápchy.** Ak sa chcete dostať autom z FMFI UK do Avionu, obvykle by ste použili diaľnicu a Prístavný most. V niektorých časoch dňa sa však na pripájačoch v Petržalke začnú tvoriť kolóny a vtedy je najlepšie sa danému úseku vyhnúť a radšej to zobrať po nábřeží.

Za účelom tohto príkladu rozdelíme deň na 96 pätnásťminútových intervalov a očísľujeme ich od 1 po 96. Všetky časové údaje budeme udávať v pätnásťminútových intervaloch.

Máte daný orientovaný graf $G = (V, E)$, kde vrcholy reprezentujú význačné miesta v meste a hrany reprezentujú cestné spojenia medzi nimi. Máte tiež danú tabuľku $D[e, t]$, ktorá udáva pre každú hranu $e \in E$ a pre každé $1 \leq t \leq 96$ čas, ktorý potrebujeme na to, aby sme prešli hranou e . Ak začneme v jej začiatčom vrchole v čase t , dôjdeme do jej koncového vrcholu v čase $t + D[e, t]$.

Úloha: Na vstupe máme vrcholy u, v a čas t . Navrhujte algoritmus, ktorý vypočíta najrýchlejšiu cestu z vrcholu u do vrcholu v , pričom začneme šoférovať v čase t . Môžete predpokladať, že šofér si nemôže dať nikde prestávku a že časy sú dané v celých pätnásťminútovkách.

Hint: Úlohu možno riešiť tak, že vstupné dáta reprezentujete ako nový graf G' a potom použijete jeden zo štandardných grafových algoritmov. Ak sa rozhodnete pre tento typ riešenia, popíšte všeobecne graf G' a na malom príklade ukážte jeho konštrukciu.

3. [20 bodov] **Priemer grafu** (programátorská úloha).

Odovzdávanie úlohy: <https://testovac.ksp.sk/tasks/eads2023-du5a/>

Na vstupe je neorientovaný neohodnotený súvislý graf. Nájdite najmenšie číslo d také, že vzdialenosť každej dvojice vrcholov je najviac d .

Formát vstupu: V prvom riadku vstupu sú čísla n, m (počet vrcholov a počet hrán). V nasledujúcich m riadkoch je popis hrán. V každom riadku sú dve čísla, ktoré vyjadrujú, ktoré dva vrcholy spája daná hrana (vrcholy grafu číslované od 0 do $n - 1$). Môžete predpokladať, že medzi každou dvojicou vrcholov vedie maximálne jedna hrana a že v grafe neexistujú slučky.

Formát výstupu: Vypíšte jeden riadok, ktorý obsahuje číslo d .

Obmedzenia a bodovanie: Na získ plného počtu bodov je nutné, aby váš program dal v časovom limite správnu odpoveď pre vstupy, kde $1 \leq n \leq 100$. Na získ polovice bodov je nutné, aby váš program dal v časovom limite správnu odpoveď pre vstupy, kde $1 \leq n \leq 15$.

Príklad vstupu:

```
5 5
0 1
1 2
2 3
3 4
4 0
```

Príklad výstupu:

```
2
```

4. [20 bodov] **Vonku je zima** (programátorská úloha).

Odvzdávanie úlohy: <https://testovac.ksp.sk/tasks/eads2023-du5b/>

V meste Montreal v Kanade nasnežilo a je tam strašná zima. Našťastie je v meste vybudovaný rozsiahly systém podzemných tunelov, ktorými sa dá presúvať medzi jednotlivými domami bez toho, aby človek čo i len vystrčil nos von. Tunelmi sa však nedá dostať od hocikiaľ hocikam. To znamená, že občas sa nedá inak, treba si obliecť teplý kabát a využiť vonkajší chodník.

Na vstupe je daný zoznam tunelov a chodníkov v meste, vrátane ich dĺžok. Našou úlohou je odpovedať dotazy, ako sa dostať z domu a do domu b optimálnou cestou. Optimálna cesta prekonáva **najmenšiu možnú vzdialenosť x po vonkajších chodníkoch**; v prípade že existuje viacero ciest s rovnakou vzdialenosťou x , tak optimálna cesta má spomedzi nich najkratšiu možnú celkovú dĺžku y .

Formát vstupu: V prvom riadku vstupu sú tri kladné celé čísla: počet domov n , počet spojení medzi nimi (tunelov alebo chodníkov) m a počet dotazov q . Platí $1 \leq n \leq 4000$, $1 \leq m \leq 40000$ a $1 \leq q \leq 30$. Domy sú očíslované od 0 po $n - 1$. Nasleduje m riadkov, každý z nich popisuje jeden tunel alebo chodník tromi číslami x (z ktorého domu), y (do ktorého domu), d (dĺžka, $0 \leq d \leq 10^6$) a jedným písmenom T ('P' ak je to tunel pod zemou alebo 'N' ak je to chodník nad zemou).

Za popisom tunelov a chodníkov nasleduje q riadkov, každý z nich obsahuje jednu otázku: dve čísla domov u, v .

Formát výstupu: Pre každú otázku vypíšte jeden riadok, ktorý hovorí ako sa vieme dostať optimálnou cestou z u do v . Ak sa z u do v nedá dostať vôbec, vypíšte '-1 -1'; v opačnom prípade vypíšte čísla x a y zodpovedajúce optimálnej ceste definovanej vyššie, pričom x je celková dĺžka použitých vonkajších chodníkov a y je celková dĺžka cesty.

Príklad vstupu:

```
2 1 1
0 1 30 P
0 1
```

Príklad výstupu:

```
0 30
```

Príklad vstupu:

```
2 1 1
0 1 30 N
0 1
```

Príklad výstupu:

```
30 30
```

Príklad vstupu:

```
4 2 3
0 1 30 N
2 1 30 P
0 1
0 2
0 3
```

Príklad výstupu:

```
30 30
30 60
-1 -1
```

Všeobecné pokyny

Písomné úlohy. Písomné úlohy odovzdávajte do *Google Classroom* ako PDF súbory v stanovenom termíne. **Každý príklad odovzdajte v osobitnom PDF súbore.** Na neskoro odovzdané riešenia sa nebude prihliadať. Píšte riešenia takým spôsobom, aby obsahovali všetku potrebnú informáciu na pochopenie vášho riešenia, ale súčasne aby boli stručné a ľahko pochopiteľné. Všetky tvrdenia je potrebné zdôvodniť (a to aj v prípade, že to nie je explicitne napísané v zadaní).

Ak sa v zadaní požaduje vyriešenie algoritmickej úlohy, odovzdajte najlepší algoritmus, aký viete navrhnúť. Základným kritériom na hodnotenie bude *správnosť algoritmu*, druhým kritériom bude jeho *časová, prípadne pamäťová zložitosť*. Správny ale pomalý algoritmus dostane podstatne viac bodov ako algoritmus, ktorý je síce rýchly, ale nedá správnu odpoveď na každý vstup. Neefektívne algoritmy spĺňajúce podmienky zadania dostanú cca 50% bodov. Súčasťou vášho riešenia musia byť nasledujúce časti:

- Najprv popíšte hlavnú myšlienku algoritmu.
- Vyjadrite algoritmus formou pseudokódu.
- Ak to nie je zrejmé na prvý pohľad, ukážte že váš algoritmus je správny.
- Nezapodniete na analýzu zložitosti algoritmu.

Ak nie je povedané inak, logaritmy majú základ 2.

Programátorské úlohy. Pri programátorských úlohách je Vašou úlohou odovzdať len funkčný program, nie je vyžadované písomné riešenie. Riešenie odovzdávate cez webové rozhranie <https://testovac.ksp.sk/tasks/>, kde bude okamžite otestované na niekoľkých vstupoch a dozviete sa, koľko bodov získalo (body získate, keď všetky vstupy z danej sady vyriešite správne v časovom limite). Riešenie môžete odovzdávať aj viackrát, hodnotí sa posledné riešenie odovzdané v stanovenom termíne. Na odovzdávanie riešení (keďže na testovač nefungujú univerzitné prihlasovacie údaje) je nutné sa na stránke zaregistrovať (vľavo na stránke testovača). Pri vytváraní účtu nastavte správne meno a priezvisko, a ako používateľské meno nastavte Váš univerzitný login. Nezapodniete tiež napísať Vaše používateľské meno do PDF súboru k ostatným úlohám, ktoré odovzdávate. Podrobnosti o tom, ako má váš program vyzeráť (vrátane povolených programovacích jazykov), nájdete v sekcii "Čo odovzdávať?". Informácie o testovači nájdete v sekcii "Odpovede testovača".

Na zoznámenie sa s rozhraním testovača si môžete vyskúšať naprogramovať niektoré z úloh z časti "Úvod do programovania".