

Domáca úloha č. 5

1-AIN-105, Zima 2017

Termín: na prvej skúške

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] **Polynomiálne redukcie.** V tejto úlohe sa budeme zaoberať niekoľkými párami problémov A, B , pre ktoré ste na prednáškach alebo cvičeniach videli redukciu $A \leq_p B$ (všetky tieto redukcie sú spísané v poznámkach na stránke predmetu). Pre každý pár je zadaný vstup x pre problém A a vašou úlohou je:

- Zostrojiť vstup $f(x)$ pre problém B podľa konštrukcie prezentovanej v poznámkach.
- Zistiť, či $f(x)$ je pozitívna alebo negatívna inštancia problému B .
- Ak je odpoveď “áno”, nájdite príslušný kombinatorický objekt, ktorý demonštruje riešenie problému B pre vstup $f(x)$ a ukážte, ako sa toto riešenie vzťahuje k riešeniu problému A pre vstup x .
Napríklad pre $A = \text{HAM}$, $B = \text{TSP-D}$, ukážte obchádzku obchodného cestujúceho pre graf $f(x)$ s požadovanou dĺžkou a ukážte, ako sa táto obchádzka zmení na Hamiltonovskú kružnicu v pôvodnom grafe.
- Ak je odpoveď “nie”, zdôvodnite ju.

- a) $A = \text{HAM}$, $B = \text{TSP-D}$, x je neorientovaný graf $G = (V, E)$, $V = \{1, 2, 3, 4\}$, $E = \{(1, 2), (1, 3), (2, 3), (2, 4)\}$
- b) $A = \text{3-SAT}$, $B = \text{VC}$, $x = (u_1 \vee u_2 \vee \neg u_3) \wedge (u_1 \vee \neg u_2 \vee u_4) \wedge (\neg u_1 \vee \neg u_2 \vee \neg u_4) \wedge (u_3 \vee u_1 \vee \neg u_2)$
- c) $A = \text{3-SAT}$, $B = \text{SUBSET-SUM}$, $x = (u_1 \vee u_2 \vee \neg u_3) \wedge (u_1 \vee \neg u_2 \vee u_4) \wedge (\neg u_1 \vee \neg u_2 \vee \neg u_4) \wedge (u_3 \vee u_1 \vee \neg u_2)$
- d) $A = \text{SUBSET-SUM}$, $B = \text{COIN}$, x : set $S = \{1, 3, 4, 5, 6\}$, cieľová suma $T = 9$

2. [20 bodov] **Najdlhšia jednoduchá cesta.** Uvažujme nasledujúci problém: Daný je neorientovaný ohodnotený graf $G = (V, E)$ a dva vrcholy u a v . Váhy hrán môžu byť kladné aj záporné. Úlohou je nájsť najdlhšiu jednoduchú cestu z vrcholu u do vrcholu v . (Jednoduchá cesta je taká cesta, v ktorej sa žiaden vrchol neopakuje.)

- a) Sformulujte tento problém ako rozhodovací problém.
- b) Ukážte, že váš rozhodovací problém je v NP.
- c) Ukážte, že váš rozhodovací problém je NP-ťažký.

3. [20 bodov] **Programátorská úloha.** Uvažujme nasledujúce dva rozhodovacie problémy:

CLIQUE: Vstupom je graf a číslo k . Obsahuje graf kliku veľkosti k ? (Klika je taká množina vrcholov, kde je každý vrchol susedný s každým.)

VERTEX-COVER: Vstupom je graf a číslo k' . Dá sa vybrať podmnožina vrcholov X s veľkosťou najviac k' tak, aby každá hrana mala aspoň jeden z koncov v X ?

Dostanete vstup pre problém **CLIQUE**. Vašou úlohou je prerobiť ho na vstup pre **VERTEX-COVER** tak, aby odpoveď pre oba vstupy bola rovnaká.

Formát vstupu: Na prvom riadku sú čísla n, m, k , kde n je počet vrcholov grafu, m je počet hrán a k je požadovaná veľkosť kliky. Nasleduje m riadkov. Každý z nich obsahuje dve čísla—čísla koncových vrcholov jednej hrany. Vrcholy grafu majú čísla $1, 2, \dots, n$. Môžete predpokladať, že graf neobsahuje násobné hrany a sľučky.

Formát výstupu: Na prvý riadok vypíšete čísla n', m', k' , kde n' je počet vrcholov nového grafu, m' je počet jeho hrán a k' je maximálny počet vrcholov pre množinu pokrývajúcu hrany. Následne vypíšete m' riadkov. Každý z nich má obsahovať dve čísla—čísla koncových vrcholov jednej hrany. Vrcholy grafu majú čísla $1, 2, \dots, n'$. Váš graf môže mať najviac 60 vrcholov a nesmie obsahovať hrany a sľučky. Taktiež k' musí byť nezáporné celé číslo.

Obmedzenia a bodovanie: Na získanie plného počtu bodov je nutné, aby váš program dal správnu odpoveď pre všetky vstupové údaje, kde n je najviac 40.

Príklad vstupu:

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

Príklad výstupu:

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

Vo vstupnom grafe existuje klika veľkosti 3 (vrcholy 1, 2, 3), taktiež tam existujú kliky veľkosti 0, 1, 2. Hrany výstupného grafu sa dajú pokryť pomocou vrcholov 1 a 2. Takže odpoveď na oba vstupy je áno.

Všeobecné pokyny

Písomné úlohy. 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.
- Nezabudnite na analýzu zložitosti algoritmu.

Písomné úlohy odovzdávajú na *papieri* (či už vytlačené alebo písané rukou) pod dvere kancelárie M-163 v stanovenom termíne. Na neskoro odovzdané riešenia sa nebude prihliadať. Nezabudnite jasne napísať svoje plné meno a priezvisko na prvú stranu a svoje riešenia pevne zopnúť spinkovacím strojčekom.

Programátorské úlohy. Pri programátorských úlohách je vašou úlohou odovzdať len funkčný program, nie je vyžadované písomné riešenie. Riešenie odovzdávate cez webové rozhranie `foja.dcs.fmph.uniba.sk/eval`, 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. Navyše si dajte pozor, či v systéme máte správne vyplnené meno a priezvisko (sekcia Mój účet). Podrobnosti o tom, ako má váš program vyzeráť (vrátane povolených programovacích jazykov), nájdete v sekcii Návod.