# IA101 – Algoritmika pro těžké problémy

# Teoretické otázky

a) Definujte pojem „pseudopolynomiální algoritmus“
b) Definujte pojem h-zúžení problému
c) Dokažte, že pro celočíselný problém A existuje pseudopolynomiální algoritmus, takový, že pro každý polynom h existuje polynomiální algoritmus pro h-zúžení problému A.

a) Formulujte úlohu lin. programování v standartním tvaru a k ní duální úlohu.
b) Napiš slabou větu o dualitě a dokaž ji
c) Napiš podmínky komplementarity.

Popsat algoritmus Las Vegas nebo Monte Carlo.

# Nesetřídené otázky

Zformulujte algoritmus pro problem 1,2-OTSP (TSP, ktery je nad orientovanym grafem, jehoz hrany mohou mit ohodnoceni jen 1 a nebo 2) (22.1.2009)

---

Zformulujte algoritmus prohledavani okoli pro problem:

Je zadan graf G = (V, E) a mnozina vrcholu S = {s_1, ... , s_k} a cilem algoritmu je odstranit z grafu hrany tak, aby kazdy vrchol byl v samostatne komponenete souvislosti (jinymi slovy - aby neexistovala cesta mezi jakymikoli s_i a s_j pre i != j ). Hleda se minimalni reseni - cenou reseni je soucet vah odstranenych hran. (22.1.2009)

---

Ukázat, kde nebude fungovat (heuristický) alg. lokálního vyhledávání, dokázat nebo ukázat protipříklad (snadné) (2012-01-09)

---

Úkolem je najít rozklad množiny čísel na dvě podmnožiny tak, aby součet prvků v těch podmnožinách byl co nejpodobnější. Vstupem je tedy množina čísel M ale je (jakkoli) rozdělena na dvě stejně mohutné podmnožiny A a B (anebo v jedné z nich je o prvek víc, když je počet prvků lichý). Uvážíme algoritmus založený na heuristice prohledávání okolí: ten pracuje tak, že z vybere největší prvek množiny M, jehož přesunutím z jedné podmnožiny do druhé, se sníží rozdíl |(∑a∈A a)−(∑b∈ B b)| . Cílem bylo analyzovat časovou složitost a najít vstupní instanci takovou, že heuristika nenajde optimální řešení, ale skončí v lokálním optimu. (15.1.2013)

---

Problém výroby produktov. Máme množinu zariadení Z, ktorých inštalácia stojí cenu fi (pre každé i zo Z) a množinu produktov P, ktorých výroba stojí na zariadení i cenu cij (pre každý produkt j z P). Cieľom je určiť množinu I (podmnožina Z) zariadení, ktoré majú byť inštalované a funkciu Φ: P → I, ktorá určí, ktorý výrobky ku strojom. Cena je definovaná ako súčet cien za inštalácie zariadení z I a cena, za ktorú sa výrobky vyrábajú na príslušných strojoch. Cieľ je minimalizácia. (15b.) (7.1.2016)

---

Navrhněte parametrizovaný algoritmus pro problem vrcholového pokrytí grafu. Zdůvodněte, proč se jedná o parametrizovaný algoritmus. Nezapomeňte uvést jakou parametrizaci problému vrcholového pokrytí používáte. (14.1.2019)

---

Formulujte Christofidesův algoritmus pro problem obchodního cestujícího. Dokažte, že alg je 3/2-aproximativní a uveďte, pro jaké typy grafů je garantován tento specifický faktor. (14.1.2019) (skripta)

---

Vstupem problém je množina M přirozených čísel a přirozené číslo S. Přípustným řešením je každý množina X taková že, suma X je menší nebo rovna S. Cenou přípustného řešení je suma čísel v X. Úkolem je najít přípustné řešení z max cenou. Navrhněte pseudopolynomiální algoritmus pro problém a dokažte že algoritmus je pseudopolynomílní. (21.1.2019)

Vstupem jsou dva řetězce stejné délky a výstupem máme rozhodnout, zda-li jsou stejné. Navrhněte náhodností komunikační protokol se složitostí O(log n) a zdůvodněte, zda se jedná o Las Vegas nebo Monte Carlo a určete přesný typ. Komunikační protokol předpokláda, že problém řeší dva počítače a každý z nich zná jen jeden z řetězců, složitostí chápeme počet bitů, které si počítače mezi sebou pošlou. (21.1.2019) (skripta)

# Hopfieldova síť

Hopfieldova neuronová síť - najde Hopfieldův alg. vždy optimální konfiguraci grafu? Pokud ano, dokažte, pokud ne, najděte graf, na kterém HS najde pouze lokální optimum, ale ne globální optimum. (5.1.2011)

---

Bol definovaný problém kladnej Hopfieldovej siete (hrany sú iba kladné). Rozhodnúť, či algoritmus Hopfielfovej siete nájde optimálne riešenie. Ak áno, dokázať. Ak nie, nájsť graf a výpočet, ktorý to popiera. (7.1.2013)

---

Algoritmus lokalneho vyhladavania pre hopfieldovu siet. Popisat + aka je casova zlozitost a dokaz ze to funguje. (15b.) (7.1.2016)

---

Algoritmus lokalneho vyhladavania pre hopfieldovu siet. Popisat + aka je casova zlozitost a dokaz ze to funguje - vsetko je to v slajdoch a myslim ze som to tam aj tak napisal 15/15b. Dokaz stoji na tom ze suma abs. hodnot vah dobrych hran rastie. (20.1.2014) (skripta)

# Randomizace / derandomizace

Máme x1,...,xn proměnné, které nabývají hodnot 0/1. Máme konstantu b, která nabývá hodnot 0/1 a k rovnic tvaru (xi+xj) mod 2 = b. Napsat randomizovaný alg. polynomiální složitosti s předpokládaným počtem splněných rovnic alespoň 1/2c* (kde c* je maximální možný počet splněných rovnic). Napsat deterministický alg. s polynomiální složitostí garantující počet splněných rovnic aspoň 1/2c*. (5.1.2011)

---

Vstupem je n čísel kde každé nabývá hodnoty 0 nebo 1 a k rovnic tvaru: (xi + x j ) mod 2 = b kde i ≠ j , i , j ∈ {1,... n} a b je konstanta buď 0 nebo 1. Označme c * maximální počet splnitelných rovnic. Cílem je ohodnotit literály xi , i ∈ {1, ... , n} nulami a jedničkami a to pomocí:

1) randomizovaného algoritmu, jehož očekávaný výsledek bude splnění alespoň 1 2 c * rovnic.

2) metodou derandomizace z něj udělat deterministický alg. Který splní alespoň 1 2 c * rovnic. (15.1.2013)

---

Napsat randomizovaný algoritmus pro vektor výrazů (x_1 XOR x_2) = b, tak aby Expected(splněných řádků) bylo >= řádků / 2 a derandomizovat jej podle Expected(x). (2012-01-09)

---

Máme problém zafarbenia grafu troma farbami. Máme vybrať zafarbenie tak, aby počet hrán spájajúcich vrcholy s rovnakou farbou bol minimálny (teda dva susedné vrcholy by nemali mať rovnakú farbu). Nech c* je optimálne riešenie.

a) napísať randomizovaný algoritmus, pri ktorom bude predpokladný počet splnených hrán aspoň 2/3 c*.

b) napísať deterministický algoritmus, ktorý bude mať presnosť aspoň 2/3 c* (7.1.2013)

Problém obarvení vrcholů třemi barvami, aby co nejvíce hran mělo vrcholy s různou barvou. Zapsat pomocí randomizovaného algoritmu a poté jej převést na deterministický pomocí derandomizace. Měli jsme dokázat, že náš alg. obarví tak, že "dobrých hran" je alespoň 2/3c*, kde c* je optimum. (6.1.2014,7.1.2016)

Něco s klauzulema - randomizovaný algoritmus a derandomizace (2-2-2011)

dany 3SAT random. alg., ktery s uniformni prob. 1/2 prirazuje hodnoty; vypocitat E; derandomizovat; odhadnout pocet ok klauzuli u derandomiz. (3.2.2014)

Booleovská konj normální formula ve tvaru, že každá klauzule má 6 různých proměnných. Máme náhodnostní algoritmus, který přiřadí true a false s pravděpodobností ½. Přesně určete očekávaný počet splněných klauzulí tímto přiřazením Aplikujte derandomizaci na uvedený algoritmus a určete, jaký je aproximativní poměr výsledného deterministického algoritmu (21.1.2019)

Metodou derandomizace vytvořit deterministický algoritmus pro problém 3SAT (s omezením, že každá klauzule obsahuje různé literály), zadaný randomizovaný algoritmus přiřazuje hodnoty proměnným náhodně (1.2.2010)

---

# Lineární programování

Problém rozvrhu úloh mezi procesory - převést na úlohu 0/1-lineárního programování (1.2.2010)

Postupovat algoritmem zaokrouhlování relaxované úlohy lineárního programování pro SubsetCoverProblem k důkazu jeho f-aproximativity, kde f je prvek U s nejvíce výskyty v S_1..n. (těžké) (2012-01-09)

Napsat lineární úlohu pro maximální párování ohodnocených hran (Definice problému byla součástí zadání, je to (maximální) množina hran bez společných vrcholů.) (2012-01-09)

Zapísať ako úlohu celočíselného lineárneho programovania. Problém výroby produktov. Máme množinu zariadení Z, ktorých inštalácia stojí cenu fi (pre každé i zo Z) a množinu produktov P, ktorých výroba stojí na zariadení i cenu cij (pre každý produkt j z P). Cieľom je určiť množinu I (podmnožina Z) zariadení, ktoré majú byť inštalované a funkciu Φ: P → I, ktorá určí, ktorý výrobky ku strojom. Cena je definovaná ako súčet cien za inštalácie zariadení z I a cena, za ktorú sa výrobky vyrábajú na príslušných strojoch. Cieľ je minimalizácia. (7.1.2013, 21.1.2019)

Problem minci. Mame mince s hodnotami (h1..hk) a sumu S. Vyjadrit ako celociselnu ulohu LP. Ci sa da z danych minci tato suma vyskladat a ak ano, minimalizovat pocet minci. (15b) (20.1.2014) (řešení ve skriptech)

Byl dany lin. program jako ve skriptech (min 7x_1 + ... blah blah, 4 rovnice) - udelat dualni, podminky kompl., vztah mezi resenimi primar a dual (3.2.2014) (řešení ve skriptech)

Problem rozvrhu ako uloha LP so zaokruhlovanim. Mam pocit, ze sa to mierne lisilo oproti tej v slajdoch, ale naozaj len o chlp. Napisal som tam tu co som vedel zo slajdov. Bolo treba aj napisat aky je (kolko-aproximativny) a zaroven dokazat ze naozaj taky je.(20.1.2014)

problém váženého párování hran - do lin. Programování (5.1.2011, 15.1.2013)

# Aproximativní programování

Byl definovaný problém výběru intervalů (máme intervaly 1,...,n zadané celočíselnými hodnotami začátku a konce). Úlohou je vybrat co najvíc intervalů tak, aby se navzájem nepřekrývaly. Uvedený algoritmus vybíral od najkratšího (konec – začátek) a vybíral do výsledku ty, ktoré sa neprekrývali s doteraz vybranými. Dokázať 2-aproximativitu algoritmu a aproximativní poměr. (6. 1. 2014, 7.1.2016)

Napsat 2-aproximativní algoritmus pro problém Obchodního cestujícího s cenami hran 1,2. Určit složitost. Důkaz že je 2-aproximativní (2-2-2011)

3-aprox. alg. pro 1-3-TSP (bez predpokladani trojuhel. nerovnosti) (3.2.2014)

1/2-aprox. algoritmus pre MAX-SAT. Popisat a dokazat ze je 1/2-aprox. Takisto v slajdoch. Na konci skoro. Hodnoty premennym v klauzalach sa priraduju nahodne s pravdepod. 1/2. V slajdoch je takisto dokaz na 1/2 aproximativitu. Mam 15/15 (20.1.2014)

Vstupem problému 4C je neorientovaný graf G = (V, E). Úkolem je najít co nejmenší množinu hran E' takovou, že graf G = (V, E\E') neobsahoval žádný cyklus délky 4 (délka cyklu je počet hran cyklu). Navrhněte 4-aproximativní algoritmus polynomiální časové složitosti pro problém 4C. Dokažte, že váš algoritmus je 4-aproximativní. (14.1.2019)

Vstupem problému je neorientovaný graf G = (V, E). Úkolem je najít co nejmenší množinu hran E' takovou, že graf G = (V, E\E') neobsahoval žádný cyklus délky 3 (délka cyklu je počet hran cyklu). Navrhněte 3-aproximativní algoritmus polynomiální časové složitosti pro problém. Dokažte, že váš algoritmus je 3-aproximativní. (21.1.2019)