Príklady na cvičenia ADS II.

Jan Hric, KTIML MFF UK
e-mail: Jan.Hric@mff.cuni.cz
http://ktiml.ms.mff.cuni.cz/~hric/vyuka/alg/cvads2.pdf(,.ps)

23. prosince 2013

Zápočty 2012/13: pre dialkove, kombinovane, ... studium a studentov z *mojich* cvicení: vypracujte: 1, 9, 39, 52, 56, 84

Sklad: 37

Domácí cvičení 2013:
15.10. Aho-Corasicková (U1:1), na cvičeních zadáno b), dodatečně vypracujte d);
22.10. Alg. tří Indů (U2:17 vše);
29.10. nezadané, pro příště: (U2a:18) Rozeberte složitost Dinicova algoritmu, pokud jsou všechny kapacity jedničkové.
5.11. na potenciály: (U3:23) Ukažte (pomocí potenciálů), že k krát opakované pričítání jedničky k počáteční nule má složitost $O(k)$, místo přímočaře dokazatelného odhadu $O(k \log k)$. Resp. pro n-ciferné číslo dokažte $O(n + k)$.
(Varianta: (19) Složitost Goldbergova alg. na jedničkové sítí.)
12.11. (U4:36) Odhadněte (hloubku a) velikost sítě pro násobení dvou n-ciferných čísel, kde v klasickém násobení sčítáte vždy trojici (mezi)vstupů na dva výstupy (vlastně přenášející výstup a carry).
19.11. nezadané (U4a:28) Síť pro počítání maxima, odhad hloubky.
26.11. (U5:65) Obsah nekonvexního mnohoúhelníka
3.12. (U6:39), FTT a IFT (2 3 0 1)
10.12. -
17.12. (U7:53) převod HAM → SAT
7.1.2014
"Kombinované" studium příklady včetně nezadaných (bez (19)), a navíc: Hranová souvislost (U99:9), ...

1 Vyhľadávanie vzorkov

1 Konštrukcia *ZAPOCET 2012/13 f); 11/12 c) stroja AC pre vzorky*
a) zakaz,ak,kaz,kakao
b) xerox,mix,mixer,komix DC
c) tvar, var, varta, tatra, vatra
d) brak, rak, ambra, barak, akra
e) brok, rok, baroko, rokoko (oko, okr, bok)
f) baroko, rokoko, oko, okr, brok

2 O Optimalizaci zpětné funkce
Pokud zpětná funkce vede ze stavu t do stavu s, ze kterého nelze pokračovat dopřednou funkci, mohli by jsme stav s vynechat a pokračovat přímo na $f(s)$.
a) popište co nejobecnější podmínku pro takovou optimalizaci, při zachování korektnosti vzhledávacího stroje.
b) jaká je časová náročnost zjišťování této informace při konstrukci stroje?
c) Při konstrukci se také používá $f()$. Popište a rozeberte situaci, kdy po optimalizaci neposkytuje $f()$ dost dat pro správnou a rychlou konstrukci (zakaz,kaz,ak,kakao - stav 11 "kak" optimalizuji a při další konstrukci mám ve stavu 12 problém).
Optimalizaci $f()$ je možné začít po ukončení konstrukce $f()$, protože f využívá přechodů "ze" stavu.

3 Vyhledávací Mooreův stroj *(z knihy)*
a)

4 Úpravy Rabin-Karp *Upravte algoritmus pro situaci, když*
a) máte několik vzorků stejné délky,
b) máte několik vzorků různé délky,
c) počítáte pro vzorek několik charakteristik (signatur) pro různé (prvočísla) q

5 Složitost Rabin-Karp *Pokud zvolíte $q \geq a^l$,*
a) kolik bude falešných hitu?
b) (Past:) jaká bude složitost algoritmu?

2 Toky v sítích

6 Trasování grafu *Je dán graf s kapacitami hran* $G = (\{z, a, b, c, d, s\}, \{< z, a, 16 >, < z, c, 13 >, < a, b, 12 >, < a, c, 10 >, < a, d, 5 >, < b, c, 9 >, < b, s, 20 >, < c, a, 4 >, < c, d, 14 >, < d, b, 7 >, <$

$d, s, 4>\}$. *Najděte vrstvené sítě generované Dinicovým algoritmem a maximální toky v nich.*

Pozn.: naplnění hrany $< a, d >$ *v první fázi pro delší cesty a snižování toku na hraně.*

7 Víc zdrojů a spotřebičů *Transformujte problém nalezení maximálního toku v síti s více zdroji (a spotřebiči) na standardní problém s jedním zdrojem a jedním spotřebičem.*

8 Párování v bipartitním grafu. *Množina hran* $M \subseteq E(G)$ *tvoří* párování *v grafu G, pokud se žádné dvě hrany z M neprotínají. Maximální párování je párování s maximálním počtem hran. Navrhněte efektivní algoritmus, který určí maximální párování v bipartitním grafu.*

Návod: Převeďte úlohu na problém určení maximálního toku v jisté síti. Pro důkaz korektnosti převodu použijte větu o celočíselnosti.

9 Hranová souvislost *(ZAPOCET) neorientovaného grafu G je minimální počet hran, po jejichž odstranění G přestane být souvislý. Ukažte, jak lze pro dané k ověřit, že graf je (hranově) k-souvislý pomocí algoritmu na toky v sítích (použitím nejvýše n sítí, každá s* $O(n)$ *vrcholy a* $O(m)$ *hranami).*

Varianta: vrcholová souvislost.

10 Inkrementální update toku *Je dána síť s celočíselnými kapacitami a známým maximálním tokem. Ukažte, jak lze po změně kapacity jedné hrany o* $+1$*, resp.* -1*, v čase* $O(n+m)$ *zjistit nový maximální tok.*

Lze tento způsob aplikovat při změně jedné hrany o libovolnou hodnotu?

Pojem: inkrementální algoritmus.

11 Složitost alg. Edmonds-Karp *(Hledá nejkratší nenasycenou cestu.) Zdůvodnete odhad* $O(nm^2)$.

12 Graf s vrcholovými kapacitami *Je dán graf s kapacitami hran a vrcholů. Kapacita vrcholu určuje maximální možný tok na všech vstupních hranách.*

Transformujte tento problém na standardní problém s ohodnocenými hranami.

13 Max. tok min. ceny *Nech každé orientované hraně h sítě S=(G,c,z,s) je přiřazena její cena w(h).* Cenou toku t *v síti S nazveme funkci* $w(t) = \sum_{h \in S} w(h).t(h)$*. Definujme ceny* w_t *hran reziduální sítě* S_t *příslušné síti S a toku t takto: Pokud* $(x, y) \in E(G)$ *a* $c(x, y) \geq t(x, y)$ *položíme* $w_t(x, y) = w(x, z)$*, jeli* $(y, x) \in E(G)$ *a* $t(y, x) \geq 0$ *položíme* $w_t(x, y) = -w(x, z)$*. Dokažte, že cenu toku lze při zachování jeho velikosti snížit právě tehdy, když v reziduální síti existuje cyklus záporné ceny.*

14 Algoritmus, viz

a) Použijte řešení předchozího problému k návrhu algoritmu pro určení toho toku maximální velikosti, který má minimální cenu.

b) Popište algoritmus pro nalezení záporného cyklu. Jaká je jeho časová složitost?

c) Varianta: Algoritmus pro nalezení maximálního párování minimální ceny.

15 Dopravní problém *Máme m dodavatelů, i-tý nabízí množství* a_i *zboží, a n spotřebitelů, j-tý požaduje* b_j *zboží. Doprava jednotkového množství zboží od i-tého dodavatele k j-tému spotřebiteli stojí částku* c_{ij}*. Celková nabídka kryje poptávku.*

Hledáme takový způsob přepravy zboží od dodavatelů ke spotřebitelům, aby všechny omezení (dodavatelů a spotřebitelů) byly splněny a celkové náklady na dopravu byly minimální. Navrhněte efektivní algoritmus řešící dopravní problém.

Varianta: nabídka nekryje poptávku.

16 Uzatváranie vrcholov v Dinicovom algoritme. *Vysvětlete, proč je celkový čas pročištění sítě ve vrstvené síti (tj. úrovňovém grafu)* $O(m+n)$.

Popište implementační detaily.

Popište, jak se liší hladové (eager) pročištění sítě od líného (lazy). Zdůvodněte, že mají stejnou asymptotickou složitost.

17 Alg. tří Indů. *Blokující tok (ne nutně maximální) ve vrstvené síti lze nalézt v čase* $O(n.n)$*. Myšlenka je vždy zpracovat vrchol s nejmenší rezervou. Vstupní a výstupní rezervy si pamatujeme samostatně. Vysvětlete detaily odhadu.*

Návod: Nasycené převedení hran (a následné pročištění) a nenasycené převedení se počítá samostatně.

a) Ukažte, že vrchol V s nejmenší rezervou najdete v $O(n)$*, včetně ceny update při dobré reprezentaci.*

b) Ukažte, že propagace toku z V do Z a S zabere čas $O(n)$*. (Pozor na impl. detaily)*

c) Celkový čas nalezení toku včetně pročištění je $O(n^2)$.

18 Dinic s jednotkovými hranami. *Rozeberte složitost Dinicova algoritmu, pokud jsou všechny kapacity jedničkové.*

Pozor: rezervy můžou být až dvojky.

19 Goldberg s jednotkovými hranami. *Rozeberte složitost Goldbergova algoritmu, pokud jsou všechny kapacity jedničkové.*

20 Heuristika výběru hrany. *Hm. Pokud si můžu vybrat hranu, kterou zvolím? Jaký vliv bude mít heuristika v obecné síti, ve vrstvené síti, v alg. tří Indů? Jakou*

heuristiku zvolím, pokud chci co největší blokující tok (ve vrstvené síti)?

Popište spravedlivou (fair) strategii výběru hrany a/nebo cesty.

21 Heuristika medzery - Goldberg. *V preflow-push algoritmu můžeme uplatnit heuristiku mezery (gap heuristic): pokud pro nějakou výšku k neexistuje vrchol u s $height(u) = k$, pak pro každé $u \neq s$ které má $height(u) > k$ lze nastavit $height(u) = max(height(u), height(s)+1)$, tj. zvýšit všechny vrcholy s výškou nad k aspoň nad s. Korektnost plyne z toho, že množiny vrcholů nad a pod k tvoří minimální řez.*

22 Potenciál nenasycených hran *Ve kterých případech zvýšení a snížení potenciálu pro nenasycené hrany nedosahuje krajních hodnot a jaká heuristika z toho plyne?*

23 Potenciály *DC2013 a):*
a) Ukažte (pomocí potenciálů), že k krát opakované přičítání jedničky k počáteční nule má složitost $O(k)$, místo přímočaře dokazatelného odhadu $O(k \log k)$. Resp. pro n-ciferné číslo dokažte $O(n+k)$.
b) potenciál stromečků v binomiální haldě
c) potenciál pro počítání zpětných přechodů v interpretaci Aho-Corasickové
d) počítání nenasycených převedení v Goldbergově alg.
e) čištění sítě v Dinicově alg. (obvykle se tomu potenciál neříká)
f) přehašování
z) Vztah k amortizované složitosti, tj. složitosti posloupnosti operací v nejhorším případě. *Potenciál datové struktury, potenciál pro počítání složitosti algoritmu. Líná a pilná úprava datových struktur. Líné (dávkové) postavení binární haldy v $O(n)$ vs. pilné (průběžné) postavení v $O(n \log n)$.*

24 Minimální pokrytí cestami *...*

3 Hradlové siete

25 *Z rekurentných vzťahov pre hĺbku d a veľkosť s zlučovacej a triediacej siete odvoďte explicitné vzťahy (indukciou).*

$$d(M_m) = \log m + 1$$

$$d(S_m) = \frac{1}{2}(\log m)(\log m + 1)$$

$$s(M_m) = m \log m + 1)$$

$$s(S_m) = \frac{m}{4}(\log m)(\log m - 1) + m - 1$$

26 Konkrétne veľkosti triediacej siete *ZAPOCET 10/11 Pre triediacu sieť a zlučovaciu sieť konštruovanú metodou z prednášky (založenú na idee mergesortu) spočítajte pre jednotlivé hodnoty šírky $2^m, m = 0...7$, konkrétne hodnoty pre hĺbku a veľkosť sietí.*

27 Datovo nezávislé algoritmy *Ktoré z nasledujúcich algoritmov používajú výhradne porovnávanie a transpozície prvkov poľa (ich výpočet nezávisí na konkrétnych hodnotách vstupov) : mergesort, quicksort, heapsort, insertsort, triedene vyberaním, shellsort, jednostranne bublinkové, obojstranné bublinkové, bublinkové s optimalizáciou.*

pozn.: model SIMD, single instruction - multiple data; počítání na grafických kartách

28 Síť pro počítání maxima *DC2013*
a) Navrhněte síť pro počítání maxima z n čísel. Určete její hloubku a velikost.
b) Dokažte, že pokud mají hradla konstantní šířku, pak hloubka sítě je aspoň $O(logn)$. (Dolní odhad)

29 Zúženie siete *Pre $m \geq 1$ konštruujeme zúženú sieť takto. Vezmeme minimálne k také, že $m \leq 2^k$ a položme $m' = 2^k$. Dokážte, že zúžená sieť $T=\{\langle j, p_1, p_2\rangle \in S_{m'}; p_1, p_2 \leq m\}$ je triediaca sieť.*

30 Pridanie hradla *Vyvráťte hypotézu, že po ľubovoľnom pridaní 1 hradla do triediacej siete S_m ostane sieť triediaca. (Stačí S_4?)*

31 Odhady *Dokážte, že ľubovoľná triediaca sieť šírky m má*
a) hĺbku aspoň $\log m$.
b) počet komparátorov aspoň $\Omega(m \log m)$.

32 Počítanie hĺbky siete *Sieť šírky n zložená z c komparátorov je daná ako zoznam dvojíc čísel v rozsahu 1 až n. Ak dve vstupné dvojice obsahujú rovnaké číslo, poradie odpovedajúcich komparátorov je určené poradím dvojíc vo vstupnom zozname. Pre túto reprezentáciu popíšte (sekvenčný) algoritmus pre určenie hĺbky siete pracujúci v čase $O(n + c)$.*

33 Dôkaz *$d_i \leq c_{i+2}$. Hm.*

34 Sw. interpretace popisu sítě *Napsat interpret trojic transpoziční sítě. Síť je v poli K (utříděném nebo neutříděném podle hladin). Hodnoty jsou v poli A, třídíme na místě.*

35 Skladanie funkcií v carry–look-ahead *Popíšte tabuľkou veľkosti 3×3 s hodnotami {g,p,z} skladanie dvoch funkcií označených g (generuj, f(x)=1), p (prenos, f(x)=x) a z (zhoď/kill, f(x)=0)) v algoritme carry–look-ahead.*

36 Násobení čísel sítí *DC2013 Odhadněte hloubku a velikost sítě pro násobení dvou n-ciferných čísel, kde v klasickém násobení sčítáte vždy trojici (mezi)vstupů na dva výstupy (vlastně přenášející výstup a carry).*

37 Bitonické třídění *ZAPOCET 2013 c) * Hm. protisměrné porovnávání;*
a) Dokažte správnost pomocí redukce na třídění 0 *a* 1.
b) Spočtěte počet hradel a hloubku sítě pro 2^k *do šířky* 64.
c) Napište rekurzivní vzorce pro výpočet hloubky $d(n)$ *a velikosti* $s(n)$ *(pro* $n = 2^k$*) pro bitonické slučování (setřídí bitonickou posloupnost) a bitonickou třídičku (setřídí lib. poslounost pomocí bitonického třídění)*

38 Nasobenie dlhych cisel *Nasobenie metodou rozdel a panuj v case* $O(n^{\log 3})$ *je alg. Karatsuba, Ofman (1962) uvedeny v sylabe.*

Varianty: Formulujte algoritmy pomocou rekurzivneho delenia. Su dve varianty:
a) koeficienty sa delia na hornu a dolnu polovicu
b) koeficienty sa delia na neparne a parne koeficienty.

Vynasobte dve komplexna cisla pomocou troch realnych nasobeni.

Vynasobte dva linearne polynomy $ax + b$ *a* $cx + d$ *pomocou troch nasobeni.*

4 FFT

39 Počítanie *ZAPOCET 2012/13 j3) l2); 2011/12 l), l2)*
a) Počítajte DFT pre vstupný vektor $(0, 1, 2, 3)$*. Správnosť overte spočítaním inverznej DFT.*
(na prednaske: (1 0 3 2))
b) Počítajte DFT pre vstupný vektor $(3, 2, 1, 0, 3, 0, 1, 2)$*. (Kosínová transformácia).*
c) Počítajte DFT pre symbolický vektor (a, b, c, d, a, d, c, b)*. Vyjde:* $(2(a + c + b + d), \sqrt{2}(b - d), 2(a - c), -\sqrt{2}(b - d), 2(a + c - b - d), -\sqrt{2}(b - d), 2(a - c), \sqrt{2}(b - d))$.
d) Upravte minulý výsledok pre symbolický vektor $(a, b, c, d, a', d, c, b)$*. Použite toto zobecnenie aj pre spočítanie IFT. (Ako sa prejaví to, že počítame s iným korenom, na poradí hodnot vo výslednom vektore?)*
e) Počítajte iteratívnu DFT pre vstupný vektor $(0, 2, 3, -1, 4, 5, 7, 9)$.
f) Počítajte DFT a IDFT pre vstupný vektor $(2, 1, -2, -1)$ *v* Z_{17}.
f2) Počítajte DFT pre vstupný vektor $(0, 1, 2, 3)$ *v* Z_{17}*. Použite* $2^8 = 4^4 \equiv 1(\mathrm{mod}17)$*. Aký koren sa použije pre IFT? (* Obecne:* $\omega = 2^t$ *je primitívny n-tý koren pre modul* $m = 2^{tn/2} + 1$*. Pretože:* $\omega^n/2 = -1 mod m$*. Pozn.: Potrebujeme* $O(n)$ *bitov)*
**g) Iny pristup (p.32-5), (lepsi modul,* $O(\log n)$ *bitov). Počítajte DFT pre vektor* $(0, 5, 3, 7, 7, 2, 1, 6)$ *v* Z_{17}*. (:-((Pozn.:* $g = 3$ *je generátor v* Z_{17}*)*
**h) Vhodné moduly sú tvaru* $p = k2^l + 1$*, ak p je prvočíslo (alebo jeho mocnina). Ktorá odmocnina jedničky je číslo 2 pre dané l?*
i) Násobenie polynómov pomocou FFT: Vynásobte $3x - 1$ *a* $x + 2$ *pomocou FFT.*
j) Násobení pomocí FFT: Vynásobte $14 * 23$ *pomocí FFT, v desítkové soustavě. Návod: čísla reprezentujte pomocí polynomů, v poziční soustavě (zde desítkové). Koeficienty polynomu určují převáděnou sekvenci (rozmyslete si pořadí). Pomocí FFT převedete koeficienty polynomy na bodovou reprezentaci, vynásobíte hodnoty v bodech a pomocí Inverzní FFT převedete zpátky na polynom P. Do něj následně dosadíte základ soustavy (zde 10) a dostanete hodnotu součinu.*
j2) Násobenie pomocou FFT: Vynásobte $35 * 28$ *pomocou FFT, v desiatkovej sústave. Skontrolujte, či výsledný polynóm je správny a či celkový výsledok je správny - pomocou klasického násobenia polynómov, resp. čísel. (Pozn. Musíme vhodne zvoliť vektory koeficientov:* $FFT(35) = (8, 5 + 3i, 2, 5 - 3i)$*,* $FFT(28) = (10, 8 + 2i, 6, 8 - 2i)$ *). Overte, že výsledný polynom je konvolúcia vstupných.*
j3) Násobenie pomocou FFT: Vynásobte $23 * 141$ *pomocou FFT nad C, v desiatkovej sústave.*
*k) Násobenie (prevzaté): For instance: 1234 * 5678*
$(x^3 + 2 * x^2 + 3 * x + 4)(5x^3 + 6x^2 + 7 * x + 8)$ $(x = 10)$
$5x^6 + 16x^5 + 34 * x^4 + 60 * x^3 + 61 * x^2 + 52 * x + 8$
which is 7006628 at $x = 10$.
l) Binárne násobenie: 1110×1011*, tj.* 14×11*. Pomocou FFT v* Z_{17}*. (Vhodný koreň viz m.) Aký rozsah potrebujeme?*
l2) Kolik a jaké kořeny lze použít?
m) Binárne násobenie: 1110×1110*, tj.* 14×14*. Aký rozsah potrebujeme?*
n) FFT a IFT: n1) (3 -1 -1 -1), n2) (-1 3 -1 -1) n3) (-2 4 -2 0)
v Z_{17} *je* $2^8 = 1 \bmod 17$ *(protože* $2^4 \equiv 16 \equiv -1 \bmod 17$*). FT(*$< 14_{10} >$*) = FT(*$< 1110_2 >$*) = FT(01110000) = (3,14,16,6,16,11,16,3); FT(*$< 196 >$*) = FT(00123210) = (9,9,1,2,1,2,1,9). (Po IFT vyjde* 8*-násobek konvoluce, tj.* $(0, 0, 8, 16, 7, 16, 8, 0)$*). Pro kořen* 2 *ve FFT použijeme v IFT kořen* $2^{-1} \equiv 9 \bmod 17$

40 Kartézska suma *Sú dané dve množiny A a B n čísel v rozsahu* 1 *až* $10n$*. Kartézska suma A a B je definovaná takto:* $C = \{x + y; x \in A \& y \in B\}$*. Hodnoty v C sú v rozsahu* 0 *až* $20n$*. Chceme určiť prvky C a ich násobnosť v multimnožine. Riešte problém v čase*

$O(n \log n)$.

41 Inv. DFT *Napíšte pseudokód pre inverznú DFT, ktorý počíta efektívne v čase* $O(n \log n)$.

42 * Lagrangeova formula *pre n-bodovu interpolaciu je*

$$A(x) = \sum_{k=0}^{n-1} y_k \frac{\prod_{j \neq k}(x - x_j)}{\prod_{j \neq k}(x_k - x_j)}$$

Dokazte, ze $A(x)$ *je polynom najviac stupna* n*, ktory splna* $A(x_k) = y_k$ *pre vsetky* $k = 0..n-1$*. (Pr. 32.1-4/783:) Koeficienty* $A(x)$ *sa daju spocitat v case* $\Theta(n^2)$.

43 DFT-3 ***
Zobecnite FFT na prípad, keď n *je mocninou* 3*. Popíšte rekurentnú rovnicu pre časovú zložitosť (a vyriešte ju).*

44 *Nakreslite butterfly obvod pre šírku* 8 *a* 16.

45 Jednobodové vstupy *Ako prebieha DFTransformácia pre vektor s jedným nenulovým členom? Odpovedá počet transformovaných vektorov (daného druhu) číslu* n*, tj. počtu jednoprvkových vektorov?*
(Aký je vzťah transformovaných k Vandermondovej matici?)

46 Obrázky, 2 x 2 *Báza v 2D, počítanie koeficientov, "význam" koeficientov.*
Aký vplyv na obrázok majú po IDFT-2D tieto kompresie : Ponechanie len prvého, vypustenie posledného, zaokrúhlenie koeficientov (kvantizácia).
DFT2D pre matice $((7,6),(6,1))$ *a* $((6,4),(4,-2))$*. Hodnoty v maticiach po kompresii a zaokrúhlení a IDFT2D.*

47 Násobenie velkých čísel *Konkrétny príklad.*

48 * Iné transformácie a funkcie *Rozširujúce. ... Haar, wavelet, Hartley*

5 P a NP

49 Prevody
Podmnožina $\leq_p$ *Rozdelenie*
Podmnožina $\geq_p$ *Rozdelenie, bolo na prednáške*
Klika na nezavislu a naopak, ...
Nezávislá množina $\leq_p$ *Vrcholové pokrytie*
Nezávislá množina $\geq_p$ *Vrcholové pokrytie*
-? Vrcholové pokrytie $\leq_p$ *Vektorová podmnožina (batoh)*
-? Vektorová podmnožina $\leq_p$ *(číselná) podmnožina*

50 3-SAT *a) Který z převodů je triviální?*
b) Dokažte: CNF $\leq_p$ *3-CNF.*
c) Dokažte: 3-CNF $\leq_p$ *CNF.*
d) Proč chceme co nejomezenejší zadání?
b2) Proč analogický převod nefunguje na 2-CNF.

51 Polynomiální algoritmy *a) Pro 2-CNF navrhněte polynomiální algoritmus*
b) Pro splnitelnost formule v Disjunktivní NF (DNF) navrhněte polynomiální algoritmus.
c) Co můžete říct o převodu DNF na CNF, ve světle části b)? (např. pomocí využití distributivity)
d) Pro barvení grafu 2 barvami navrhněte polynomiální algoritmus. (Jaká je jiná ekvivalentní formulace?)
e) Navrhněte polynomiální algoritmus pro hledání kliky dané pevné velikosti k*. Čím se tato formulace liší od Problému Kliky?*
f) Dokážete pro některý/-é NPÚ problém najít podtřídu zadání, které jsou řešitelné polynomiálně?

52 Součet podmnožiny dynamickým programováním *ZAPOCET b,c) Popište (pseudopolynomiální) algoritmus pro řešení*
a) problému batohu
b) součtu podmnožiny (viz moje slajdy)
c) Vysvětlete, proč tyto algoritmy nelze považovat za polynomiální.

53 Hamiltonovská kružnice *ZAPOCET 08/09*
Dokažte: HAM $\leq_p$ *CNFSAT. (Motivace: programování v CNF.)*
Pozn. Kódovanie hrán vs. kódovanie poradia vrcholov. A vyjadrenie (dostatočných) omedzení. (Stačia 2 zo 4: 1x aspoň 1 a 1x najviac jeden, alebo duálne)
idea: kodovanie kruznice: premenna $x_{v,i}$ *– ak vrchol* v *je* i*-ty na ham. kruznici.*
kodovanie omedzeni (v CNF!): vrchol v *je na nejakej pozicii (tj. na Ham. kruznici) prave raz (tj. aspon raz a najviac raz), na pozicii* i *je prave jeden vrchol (dtto), vrcholy* u *a* v *na poziciach* i *a* $i+1$ *smu byt sucasne, ak je medzi* u *a* v *hrana. (trikove)*
Odhadnite pocet omedzeni (a ich dlzku), resp. celkovy pocet premennych. (Je polynomialny?)
b) Převod hamiltonovské cesty na CNFSAT. (analogicky)

54 Hamiltonovská cesta, 36.2-6 Hamiltonovská cesta *v grafu je cesta, která prochází každý vrchol právě jednou.*
a) Ukažte, že problém HAM-PATH $= \{(G,u,v) :$ *existuje hamiltonovská cesta z* u *do* v *v grafu* $G\}$ *patří do NPÚ.*
b) Ukažte, že pro HAM-PATH2 $= \{G :$ *existuje hamiltonovská cesta v grafu* $G\}$ *(tj. nemám předepsaný začátek*

a konec cesty) platí HAM-PATH2 $\leq_p$ HAM.
c) Ukažte HAM $\leq_p$ HAM-PATH2.

55 Hamiltonovská cesta heuristicky *ZAPOCET 2011/12 a), b)* Hamiltonovská cesta *v grafu je cesta, která prochází každý vrchol právě jednou.*
a) Navrhněte heuristický alg. hledáni ham. cesty.: Idea: kruznici buduji jako cestu ze "Z", kdyz nemuzu prodlouzit konec cesty, otocim nejakou koncovou cast, abych mohl prodlouzit. Pokud nemuzu, koncim neuspechem (i kdyz kruznice muze existovat). Popište formálně podmínku na otočení a prodloužení cesty.
a2) Alg. pro Hamiltonovskou kružnici místo H. cesty.
b) navrhnout (nezavislé) zlepšení algoritmu pomocí heuristik. Zduvodnete, proc by navrzene heuristiky meli zlepsit chování alg. (aspoň 3 heuristiky). Uvědomte si, že cílem není najít co nejdelší cestu, ale co nejčastěji najít Ham. cestu.
b2) odhadněte složitost alg. s heuristikou. Je polymoniální?
c) rozeberte, jak se vaše heuristiky budou chovat pro určité třídy grafů, např. husté, řídké, rovinné, regulární ... (tj. zda poskytují zlepšení pouze pro určitý druh grafů)
d) navrhněte heuristiku motivovanou nebo navrženou pro určitý druh grafu

56 Nezávislá množina na CNF *ZAPOCET 2012/13 a)*
a) Dokažte: NM $\leq_p$ CNFSAT.
Idea: v řešení instance NZ (G, k) vrcholy očíslujeme (od 1 do k), zavedeme proměnné pro "vrchol v_i má číslo j" a vynutíme podmínky na "rekonstrukci" nezávislé množiny.
(b) špatný nepolymoniální postup
c) Dokažte: KLIKA $\leq_p$ CNFSAT. (analogicky)

57 Tautologie, co-NP *Formule výrokové logiky je tautologie, pokud je pravdivá při všech ohodnoceních proměnných. TAUTOLOGIE je jazyk těch formulí se spojkami negací, disjunkcí a konjunkcí, které jsou tautologie. Ukažte, že problém TAUTOLOGIE $\in$ co-NP.*

58 SAT: Získání řešení, svědka 36.4-6 *Předpokládejme, že dostaneme černou skříňku s polynomiálním algoritmem pro rozhodování splnitelnosti. Popište, jak použít tento algoritmus pro nalezení splňujícího ohodnocení (tj. svědka) v polynomiálním čase.*

59 Převod 3-SAT na HAM *Hm. Widget: právě jedna ze dvou hran (vztah prom/neg.prom a výskytů). Widget2: nejvýš dvě ze tří hran (popis faktorů).*

60 Rozdělení podle průměru *Optimalizační úloha: rozdělte množinu na 2 části tak, aby jejich aritmetický průměr byl co nejblíž.*
Rozhodovací problém: rozdělte množinu na 2 části, aby jejich průměr byl stejný. Ukažte, že tento problém P je NPÚ.
Idea: redukce z rozdělení podmnožiny - rozdělení na stejně velké a početné části - rozdělení na neceločíselný průměr z 2p prvků, p prvočíslo.

61 Riešenie NPÚ úloh *Hm. Použitie v kryptografii.*

62 Pseudopolynomiálne alg. *Doležitosť reprezentácie a merania veľkosti vstupných dát. Podmnožina a alg. (dyn prog.)*

63 Existencia NPÚ úlohy
(TS, vstup, časUnárne) a lineárna simulácia.
Dokaz prvej NPÚ úlohy. Idea prevodu výpočtu.

6 Konv. obal

64 Prechod podľa uhlov *Algoritmus, linearita výpočtu po usporiadaní, prečo potrebujeme usporiadanie.*

65 Obsah ne/konvexního mnohoúhelníka *() Zapocet2013 c)*
a) Výpočet obsahu konvexního mnohoúhelníka. Body přicházejí setříděné podle souřadnice x.
b) Stejné. Body přicházejí podle pořadí na konv. obalu, tj. hranici.
c) Výpočet obsahu nekonvexního mnohoúhelníka. Body přicházejí setříděné podle souřadnice x.
d) Výpočet obsahu nekonvexního mnohoúhelníka. Body přicházejí podle pořadí na hranici.
Návod: Nekonvexní části se odčítají (podobné principu inkluze a exkluze).

66 Konvexní obal v čase $O(nh)$ *Navrhnout algorimus pro výpočet konvexního obalu (z neuspořádaných bodů) v čase $O(nh)$, kde h je počet bodů ležících na konvexním obalu.*
Pozn.: Obecná poloha bodů.
Použít Paretovu hranici. Složitost pridávání, správy?
Po skombinování alg. složitosti $O(n \log h)$.

67 Horní most. *Hm.*

68 K předn. *2013*
Průsečíky úseček, nejen v obecné poloze.
Výpočet obsahu (a obvodu) sjednocení osových obdélníků. (Zametání roviny intervalovými stromy.)

69

7 Krypto

70 Euklidov alg. *Výpočet. x=13, y=8, dosvedčujúce konštanty. Možnosť priebežnej kontroly.*

Počítanie modulo n.

Dokaz správnosti a invariantu. !Hm.

71 Euklidov alg. *Na cvičenia: Rozobrať dôkaz asymptotickej zložitosti Euklidovho algoritmu.*

72 RSA *ZAPOCET a) Príklad. Spocitejte r, d, ciphertext.*
a) Napr. $p = 11$, $q = 13$, $e = 7$, $m = 4$ (bitovo jednoduché). Trasujte alg. "rýchleho" umocňovania
b) Kolko je moznych e pre dane p, q?

73 Čiastočná informácia *ZAPOCET*
Predpokladajte, že v RSA sa prvočísla p a q líšia od $\sqrt{n}$ najviac o 1 bit, tj. sú skoro rovnako veľké. Koľko možností rozkladu n hrubou silou by ste museli skontrolovať?

(Metoda pro rychlé řešení při malém rozdílu p a q (kvadratické zbytky), transformace přenásobením na podobné hodnoty)

74 Požiadavky na komunikáciu *Výčet: doporučené listy, doručenka, neporušenosť, chyby média ..., CA.*

Techniky obrany.

!Dokazovanie správnosti protokolov :-) . (Časté chyby s odstávkou)

75 Krypto pomocou Súčtu podmnožiny *Hm.*

76 Princípy MD5 a DES *Hm.*

77 Princípy šifrovania blokov *Hm. Bude na Základoch OS (+-)*

8 Aproximačné alg.

78 Vrcholové pokrytie *Popíšte graf, pre ktorý alg. približného vrcholového pokrytia vždy nájde len suboptimálne riešenie.*

(niekoľko možností)

Popíšte vyhody heuristiky opakovaného spúšťania a omedzenie "ťažkých" variant.

(Takýto graf je "ťažká" inštancia problému pre približné riešenie, pretože nikdy nezískame optimálne riešenie.)

79 Výber vrcholov vysokých stupňov *Heuristika vyberá vždy a opakovane vrchol najväčšieho stupňa a vylúči všetky s ním indcidentné hrany. Ukážte príklad grafu, kde táto heuristika nemá pomerovú chybu 2. Hm*

80 Pokrývanie stromu *Navrhnite hladový algoritmus pre nájdenie optimálneho vrcholového pokrytia stromu v lineárnom čase.*

(Už bolo.)

81 Prevoditeľnosť a.a. *Hm. Vzťah vrch. pokrytia ku klike.*

82 Heur.alg *...*
- ruzne omezeni pro prakticke reseni NPC problemu – barevnost: heuristicky minimalizujeme – heuristiky ... – analogicky: vrcholove pokryti - alg. s pevnou chybou – batoh: aprox. – hamK/C: s co nejvetsi pravdepodobnosti chceme najit HamK/C - odpoved je ano/ne (nezajima nas nejvetsi kruznice/nejdelsi cesta) – greedy/hladove rozhodnuti jako heuristika - staticke, dynamicke, ... - TSP (opt): asi rozsirujuca cast

83 Transformácia na TSP s trojuh.ner. *Ukážte, ako v polynomiálnom čase transformovať inštanciu problému obchodného cestujúceho na novú inštanciu, ktorej cenová funkcia spĺňa trojuholníkovú nerovnosť, pričom sa zachovajú optimálne cykly.*

Vysvetlite, prečo táto polynomiálna transformácia nie je v rozpore s neexistenciou polynomiálneho aproximatívneho algoritmu per obecný TSP, za predpokladu P/=NP.

84 TSP: Heuristika najbližšieho bodu *ZAPOCET a)*
Na začiatku vytvoríme cyklus z jednoho lub. bodu. V jednotlivých krokoch postupne hľadáme vrchol u, ktorý nie je v cykle a je najbližšie k cyklu, k nejakému vrcholu v. Rozšírime cyklus pridaním u za v a opakujeme, kým v cykle nie sú všetky vrcholy.
a) Ukážte, že táto heuristika vracia cyklus, ktorý je najviac dvakrát dlhší než optimálna cesta (pri platnosti trojuholnikovej nerovnosti).
b) Navrhnite zlepšenie k popísanej heuristike.

85 TSP s minimalizáciou najkratšej hrany. *Za predpokladu trojuholníkovej nerovnosti nájsť algoritmus, ktorý vracia najviac trojnásobok minimálnej cesty.*

Návod: prechod. min. kostrou. Protipríklad pre "len" dvojnásobok.

86 Kríženie v rovine *Ukážte, že optimálna cesta s bodmi v rovine a euklidovskou metrikou sa nekríži.*

Navrhnite ďalšie lokálne optimalizácie. (jednosegmentové, dvojsegmentové)

87 Vyškrtávacie heuristiky *Navrhnite iné heuristiky pre vyškrtávanie vrcholov z úplnej cesty. Dokážte,*

že všetky zaručujú pomerovú chybu 2. (viac možností: 2,n)

Vysvetlite použitie preusporiadania vrcholov. (Viterbiho alg., dyn. prg.)

88 Pokrytie množín je NP *Hm. Ukážte, že problém pokrytia množín je NP úplný prevodom z vrcholového pokrytia.*

... pokrytie mmnožín

89 Aprox. schéma *Navrhnite modifikáciu aprox. alg., ktorý bude počítať odhad najmenšej hodnoty, ktorá je súčtom podmnožiny daného zoznamu a nepodtečie dané t.*

9 Dynamicke programovanie

90 Maximalna cesta v DAG
a) Formulujte hladanie maximalnej cesty v acyklickom grafe (DAG) ako ulohu DP.
b) Popíšte jednotlivé varianty algoritmu DP. U ktorých potrebujete topologické usporiadanie grafu?
c) Ak hľadáte max. cestu do daného vrcholu, ktorú variantu alg. DP použijete?
d) Je možné použiť algoritmus pre hľadanie max. cesty z daného vrcholu?

91 Klasické ulohy *Odhadnite časovú a pamäťovú zložitosť.*
a) Triangulácia mnohouholníka
b) Optimalne násobenie matic
c) stavba optimalneho vyhladavacieho stromu
d) menej klasické: Viterbiho alg. - viz
e) Kontrola (lineárneho) dokazu (výrokovej) formule. (úloha na existenciu/spravnosť, rekonstrukcia grafu.) Optimalizacia, vybranie poddokazu lemy.

92 Existencne ulohy *(napr. analýza príslušnosti slova do bezkontextovej gramatiky)*
a) Formulujte ulohu dynamického programovania pre situáciu, keď zistujeme existenciu (bez ohodnotenia) nejakej struktury na zaklade dovoleneho kombinovania mensich struktur.
b) Modelovy priklad: Zistite, či existuje sposob, ako skladanim získať daný obrazok. Pravouhly ciernobiely obrazok mozete skombinovať z dvoch mensich (s odpovedajucimi rozmermi) vodorovnym alebo zvislym zložením. Elementarne obrazky su velkosti aspoň 2 a su jednofarebne. b2) Elementarne obrazky su buď biele ľubovoľných rozmerov alebo cierne veľkosti 2x1 (veľkosti kostky domina) v ľubovoľnej orientacii. b3) Obidve časti obsahuju počet čiernych polí líšiacich sa najviac o 1.
c) Pre pokročilých: formulujte "subsumpciu" obrazkov (odstraňovanie bielych okrajov a pod.).
d) (Popíšte rozdiel medzi algoritmami, ak elementarne obrazky su dane vyctom bez udania polohy a vyctom s udanou polohou. Oboje na bielom pozadi.)

93 Viterbiho alg. *Hm.*
Hľadanie najpravdepodobnejšej cesty pre daný reťazec ("vetu") v grafe s hranami označenými slovami (tj. reťazcami) a ohodnotenými pravdepodobnosťami. Hľadáme max. cestu (najpravdepodobnejšiu), ktorá pri prechádzaní hrán a zreťazovaní ich označení dá požadavaný (dopredu daný) reťazec. Pravdepodobnosti sa pri prechode násobia.
Obecnejšia formulácia dovoľuje vypúšťanie a pridávanie písmen.