

Zda se, že stačí, když budete vědět, jak ty algoritmy fungují a co dělají, formální pseudokod potřeba není. Minimalně pokud to p. Koubek z vašeho popisu pochopí.

přijď brzo a sedni si ke katedře = dostaneš normální otázku

nerozhodné známky mají tendenci konvergovat ke trojce

Korektní popis a pochopení funkce algoritmů a nějaká základní fakta/pozorování na trojku stačí

Hashování

- pak hashovani.ps, to už pro mě byla trochu pain, takže jsem si otevřel Introduction to Algorithms, kde máte krásně popsane universalni i perfektní hashování, ale jsou tam i dobře nějaká ostatní hashování, pak je tam krásně čas quicksortu. z drivejska si pamatuju, že tam jsou krásně i ostatní věci - jako RB trees a jiny apod.

✓ Separované řetězce (!)

(2) očekávaná délka nejd. řetězce u sep. hashování. napsal jsem očekávanou průměrnou délku, na to mi odpověděl, že na 3 to stačí. Dále jsem se pokusil dopočítat průměrný nejdelší, nedopočítal jsem, na to mi dal za 2 a šel jsem ;) Musím říct že p. Koubek je velmi milý, dává dostatek času a při standardním povědomí o věci dává známku. Několik lidí po zadání otázky odešlo, což komentoval slovy, že toho je hodně, ale že to nejsou těžké věci.

(3) já měl délky řetězců u separovaných hash.: očekávané a očekávané nejdelší (kupodivu jsem vyplodil i cca pulku dukazu toho druhého, ale celkově po cca 3 a půl hodinách za 3) rozhodně se vyplatí aspoň tusit jak do jednotlivých dukazu - jako obvykle je toho člověk schopný docela dost domyslet na zkoušku, ale když si nepamatuje co má vyjit...

(2) hešování se separovanými řetězci a důkaz očekávané délky maximálního řetězce. Popsal jsem hešování jako takové a operace, důkaz sem si nevzpomněl, tak mi řekl ať dokážu alespoň očekávanou délku řetězce. Což jsem vymyslel. Výsledek nakonec za 2. Koubek je opravdu v pohodě.

(4) Dostal jsem očekávanou délku a očekávanou nejhorší délku řetězce u Hashování se separovanými řetězci. Jelikož jsem to druhé netušil, ani nevymyslel po 15 minutách jsem se rozloučil.

(1) Tak už to mám taky za sebou. Byl jsem první a dostal jsem hashování se separovanými řetězci. Popis, algoritmy, důkaz časové složitosti v úspěšném a neúspěšném případě. O 4 a půl A-čtyřky a necelé dvě hodinky později jsem měl hotovo, a kdybych se nezamotal ve výpočtech, tak to mohlo být dřív. Musím ocenit, že dr. Koubek dává dost času a je hodnej. Jenom by mohl mluvit trochu hlasitěji 😊

- Popis, algoritmy,
- důkaz časové složitosti v úspěšném a neúspěšném případě (to samý jako nize?)
- očekávaná délka řetězce + **dukaz**
 - $E = n / m$
- očekávaná délka nejdelšího řetězce (očekávaný nejhorší případ)
 - $E(NP) = O(\log n / \log \log n)$

<http://forum.matfyz.info/viewtopic.php?f=206&t=4978>

Hešování přemístování a s dvěma ukazateli

zadný záznam ☹

✓ Srůstající (EISCH, LISCH, LICH, VICH, EICH)

(1) chtěl vědět těch pět hešování jak jsou ve skriptech + jak se liší

další otázky jak fungují algoritmy stačilo schématicky není nutné psát celý ten kód co to implementuje do posledního detailu, chtěl určitě slyšet, že očekávaná délka řetězce ve srůstajících hešováních je 3 s pomocnou pamětí 2 (Tuhle kon-

kretní hodnoty jsou přímo někde ve skriptech, nebo vyjdou po dosazení do některého z těch silnějších vzorců? Je to tam - zhrnute jednou větou. Da sa to dost jednoducho prehliadnuť, bohužiaľ.) + čo za význam pomocná pamäť má (ve zkratce - pomáhá tomu aby se v tabulce neblokovalo zbytečně místo - protože prvky samozřejmě obsazují místa pro prvky z jiné skupiny 0..m).

- 5 hashování ze script, algoritmy schematicky, jak se liší
- Očekávaná délka řetězce
 - 3 s pomocnou pamětí 2
- Význam pomocné paměti
 - ve zkratce - pomáhá tomu aby se v tabulce neblokovalo zbytečně místo - protože prvky samozřejmě obsazují místa pro prvky z jiné skupiny 0..m

<http://forum.matfyz.info/viewtopic.php?f=206&t=4133>

Lineární přidávání

žadný záznam ☺

✓ Dvojitě hashování (!)

(3) Pôvodne som počul moje zadanie ako iba "Hashovanie", preto som písal všeobecné veci o hashovaní, neskôr som začal vypisovať typy hashovania a ich princípy. Keď prišiel profesor ku mne prvý krát, tak ma upozornil, že chcel "iba" Dvojitě hashovanie. Popísal som jeho princíp, algoritmi, návrh na riešenie problému s neexistujúcim efektívnym delete. O analýzu zložitosti som nezakopol. Profesor zo mňa ešte chvíľu páčil to, že $nsd(h_2(x), m)=1$, teda slovo nesúdeliteľnosť. Potom mi povedal, že to na 3 stačí a ak chcem viac, tak mám pridať analýzu zložitosti... Zobral som 3

(2) Ja dostal to dvojitě hashování, definici, jak to funguje a nutnost nezávislosti funkcí jsem napsal v pohode, potom jsem se ale hrozne zaseknul v dukazu očekávaného počtu testu a Dr. Koubek mi musel poradit skoro celou hlavní myšlenku, pak už jsem to teda nějak dopočítal, takže nakonec docela příjemne za 2 ... ale stejně doporučuju nepodcenovat .

(3) za půl hodiny jsem popsala 4 stránky textem, žádné odhady nebo důkazy, jen algoritmy member a insert jsem měla i skoro formálně, vysvětlení proč delete nejde dobře, zbytek ve větách, třikrát se mi řekl, že tam mám něco špatně, třikrát jsem ukázala, že tu podmínku $i < m$, kterou chtěl slyšet, tam mám napsanou, pak zeptal se "je jasné, že tohle je za tři?", poděkovala jsem a šla...

- princip
 - $(h_1(x) + ih_2(x)) \bmod m$ je prázdný řádek, a tam uložíme prvek x.
- nutnost nezávislosti funkcí
- podmínka $i < m$
- algoritmy
- proč delete nejde dobře
 - návrh na řešení problému s neexistujícím efektívnym delete (na 1-2)
- analýza na složitosti (na 1-2)

Univerzální hašování (!!)

(1) **c-univerzální systém** K tom univerzálnímu systému jsem napsal definici, proč se to používá (jiné podmínky oproti třeba sep. hash. => randomizace), existenci ($c = \dots$), vlastnosti (oc. délka řetězce, dolní odhad pro vel. systému a Markov. ner.) a nakonec konstrukci. Tady bych rád podotknul, že jsem tam neměl ty silné upravy a rozepsané odhady, ale jen základní myšlenky toho důkazu. Čili pokud si pamatujete začátek, myšlenku a konec a budete to mít všechno správně, tak se max. zeptá na pár detailů a bude to stačit. Kvůli nepřesnostem (a možná tím nerozepsaným důkazům?) jsem dostal ještě otázku pro boj mezi 1-2: (a,b)-stromy - aproximace vyvazování.

(?) Při tom hashování ho zaujímalo, ako to funguje, prečo hľadáme niečo ako 3.25-univerzálne hashovanie od toho sme sa dostali k tomu ako fungujú pseudonáhodné generátory tj. generujú ďalšie číslo podľa predchádzajúcich vygenerovaných a preto nie sú dobré pre veľa vygenerovaných čísel.

(1) Dostal jsem univerzalni hashovani, tak jsem popsal tri papiry, on si je dukladne precetl (az mi z te jeho dukladnosti behal mraz po zadech), poukazal na nejakou mensi chybu, nechal me ji opravit a dal mi jednicku. Tim chci povzbudit k nauceni, on to pan Koubek oceni

(3) Ja jsem dostal univerzalni hešování (definice a konstrukce c-univerzalniho systému), což jsem si vůbec ale vůbec nepřál, a podle toho taky vypadalo. Pomrdal jsem už samotnou definici a konstrukci jsem sice popsal, ale ke správnosti jí chybělo mnohé (uměl jsem to z Čepkových slajdů z ADS1 a ne moc přesně). Koubek si to přečetl a prohlásil: "No, tak to opravíme" a chvíli mě nechal přemýšlet. Po 5 minutách ticha se do toho vložil a začal mi říkat, co by jak asi mělo být. Za mého postupného přitakávání jsme se dostali k "tak nevím jestli vás mám vyhodit nebo vám dát trojku". Dostal jsem doplňující otázku na hešování přemísťováním a s dvěma ukazateli, což jsem bez větších potíží vyplodil a dostal za 3. Na to jak jsem byl dutej při první otázce luxusní známka a nebyl potřeba jediný důkaz. Od 4 mě pravděpodobně zachránilo to, že u první otázky jsem slovy popsal, jak by to mělo celé vypadat a tím dal najevo, že alespoň rámcově tomu rozumím. Koubek je skutečně v pohodě, při samotném zkoušení docela pomáhá a pokud máte něco špatně (snad nikdo to neměl "do puntíku"), tak vám dá šanci to opravit.

(3) hashovani... konkretně universalni hashovani 😊 Sice jsem podobnou zradu trochu čekal a něco sem umel, ale čím víc sem toho psal, tím mi bylo všechno jasné. U důkazu existence sem dokonce po chvíli přemyslení začal skrtat to, co bylo dobré 😊 Nakonec sem sesmolil ten důkaz existence c-universalniho systému (s Koubkovou vydatnou pomocí) + sem měl výsledek a část odvození delky retezce. Říkal, že mi dá ještě čas na dodelání toho odvození, že jinak mi bude muset dát trojku... takže nakonec za 3 a sem rád btw: Zda se, že ani na 2 by zbytek latky k uni. hashi asi nechtěl.

- Definice
- Existence c-univerzalniho systému
- konstrukce c-univerzalniho systému
- 3.25-univerzalni hashovani

<http://forum.matfyz.info/viewtopic.php?f=206&t=4074>

http://videlectures.net/mit6046jf05_leiserson_lec08/

Perfektní hashování (!)

(?) Napsal jsem větu o dolním odhadu velikosti systému a větu o existenci perfektního hashovacího systému s důkazy. To bylo sice správné, ale ještě jsem musel naznačit i konstrukci tohoto systému, takže trochu obsáhlejší otázka 😡 Jinak klasika, jak už psali ostatní...

(info) Nabyl jsem přesvědčení, že **za 80% potíží s tímto předmětem jsou zodpovědná ta jeho skripta** 😱. Moje doporučení pro všechny, kteří budou dělat zkoušku: vyhněte se jim a tytéž důkazy se naučte odjinud. Namátkou QS - Čepek. Konstrukce primitivní hašovací fce - odvození, které dělával Koubek na přednáškách, než napsal skripta je celkem stravitelné (asi jako středně těžký mat. důkaz), viz zápisky Lenka.

(4) dolní odhad velikosti perfektního hašování. Vypsal jsem sice několik základních věcí okolo perfektního hašování, ale protože ten dolní odhad jsem neměl, nesplnil jsem zadání, takže si dám datovky ještě jednou.

- Základní definice
- věta o dolním odhadu velikosti systému
- věta o existenci perfektního hashovacího systému
- Konstrukce hashovací funkce u perfektního hashování

✓ Externí (!)

(2) Já dostal externí hešování - napsal jsem motivaci, popsal jsem tam algoritmus, zmínil počty přístupů a uvedl příklady, Koubek si to přečetl, zeptal se na pár detailů, nepřesností a nejasností v textu, které jsem zodpověděl a poté položil otázku na využití paměti a velikost adresáře. K tomu jsem byl i po hodině přemýšlení schopen říct jen, že využití je dobré a adresář velký, načež mne po děsivé chvíli přemýšlení uvedenou větou "Co já s vámi udělám" ohodnotil za dva.

(1) Já sám dostal externí hašování a musím říct, že sem si nestěžoval, sice sem věděl jen náznak z OZD (ve skriptech sem to prolítl, že to umím 😊), ale za ty celkem 3 a půl hodiny co sem tam dohromady byl sem to dal dokupy a dal dokupy dokonce na 1

(3) O tom jsem si ze zapisku pamatoval vseho vsudy, ze Insert a Delete ma amortizovane 6 operaci a Member 3, bez tuseni, proc tomu tak je a jak to vlastne funguje 😊. Potom jsem si lehce vzpomenu na kohosi zapisky, kde ukazoval postup vkladani pro binarni cisla... zapatral jsem v pameti na Zemlickova cvika z OZD a rozepsal jsem mu tam se vsim vsudy Rozsiritelne hashovani a ono to bylo ono 😊. potom tam ze mne lamal patnact minut vzorce a odvozeni zavislosti velikosti adresare proti velikosti stranky na ukladani zaznamu s tim, ze to je to hlavni a dal mi za tri s tim, ze jsem mel na vic...

- Algoritmus
- Pocty pristupu
- Příklady
- Vyuziti pameti, velikost adresare

Stromy

✓ (a,b) stromy (!)

(4) Otazka kterou jsem si myslel ze jsem dostal: Vyvazovaci operace na (a,b) stromech.. vypracovano, podminky, algoritmy.. Otazka kterou jsem skutecne dostal: Pocet vyvazovacich operaci na (a,b) stromech... UFF! To jsem skutecne spocitat nedokazal... Pane kolego, to se budeme muset videt priste..

(1) Mám to za sebou za 1. Dostal jsem (a, b)-stromy, napsal jsem mu tam: definici, větu o tom, že pro každé n ex. strom s právě n listy, odhad na výšku stromu, strukturu vnitřních vrcholů, operace MEMBER, INSERT, DELETE, JOIN, SPLIT + jejich složitosti a pomocné operace. Občas položil nějakou záludnou otázku, ale jinak se v tom moc nešťoural.

(2) Já jsem dostal (a,b)-stromy, za dvě a půl hodky jsem vyprodukoval 6 A4, kde jsem celkem podrobně popsal (slovy, nikoli kódem) snad všechny operace na těchto stromech + nějaké povídání prosáklé z OZD I. Měl jsem jinak štěpení/slévání než bylo na přednášce, což ovšem nebylo hodnoceno negativně, jen jsem na to byl upozorněn. Dr. Koubek to vše prošel a následně pokládal dotazy směřující k nejednoznačně/nepřesně popsanému chování. Ptal se buď jak něco bude fungovat na určitém vstupu, z čehož se dalo vcelku svižně odvodit, kde je třeba ještě něco ošetřit, ale také pokládal dotazy "Bude to takto fungovat vždy ?", to jsem pak obhajoval dobré chování na všech vstupech, dokud jsem sám nenašel chybu, nebo jsem nebyl zastaven dotazem "Opravdu ?". Nakonec jsem neobhájil, že se SPLIT vejde do $O(\log |S|)$ a dostal jsem za 2 se slovy "ale to jsme si ukazovali na přednášce" 😊.

- definici, větu o tom, že pro každé n ex. strom s právě n listy, odhad na výšku stromu, strukturu vnitřních vrcholů, operace MEMBER, INSERT, DELETE, JOIN, SPLIT + jejich složitosti a pomocné operace (na 1)
- aproximace vyvazovani
- amortizovanou složitostí operací na (a,b)-stromech

<http://forum.matfyz.info/viewtopic.php?f=206&t=8158>

<http://forum.matfyz.info/viewtopic.php?f=206&t=4142>

<http://forum.matfyz.info/viewtopic.php?f=206&t=3887>

+tistena skripta

pocet vyvazovacich operaci na (a,b) stromech

✓ A-sort (!)

(2) Se svou otázkou jsem byla spokojená, napsala jsem definici (a,b) stromů, jejich vlastností, strukturu vrcholu a kde se dají využít. Pak jsem popsala A-sort, operaci A-INSERT a jak je třeba (a,b) strom pro tento alg. modifikovat. Nakonec jsem odvodila složitost, tam jsem akorát měla drobnou nejasnost. Dostala jsem čas na objasnění, to se ale nekonalo, tak mi to pan Koubek sám vysvětlil a domů jsem šla s dvojkou. Je velice příjemný.

(1) A-sort. Spisal som, co bolo v skriptach (zlozitost, algoritmus len slovne+obrazky) a bez doplnujucich otazok som sa pobral domov s jednotkou.

(2) Dostal jsem dvojkou, protože jsem neumel dokazat, ze horni hranici poctu stepeni pri A-Insertu lze odhadnout poctem vnitřních vrcholu ve strome (a tzn. je to $O(n)$). Slo to rychle, az na tu rozpacitou petiminutovku, kdy jsme s Dr. K. syn-chronne mlceli a ja predstiral, ze to zkousim vymyslet. 😊 Nakonec me (potazmo sebe) netrapil a dal mi se slovy, ze lidem, kteri to umeji, dava dvojky nerad, dvojku.

(3) Napsal jsem: definici (a,b)-stromu, co to je ze strom reprezentuje S, rozsireni definice pro A-sort, alg. A-sort a pod-proc. A-insert (to pridavani na konci jsem popsal jen slovne),

složitost $O(n + \log(F/n))$ a ze když $F \leq n \log n$, tak je složitost $O(n + \log \log n)$ cili rychlejší než Quicksort, co je to F a co je to inverze. Precetl to, zeptal se jestli "prvním prvkem" myslím ten nejmenší (ano), a pak se zeptal, ze jestli se mi teda nechce nic pocitat, tak mi da za 3, coz jsem vzal.

(3) kopec reci okolo (preco sa pouziva, co su inverzie atd), toto mi islo najlepsie 😊

def a,b stromov (zabudol som dodefinovat ze podmienky neplatia pre koren, taketo veci si prednasajuci strazi ale ne-myslím ze by to vnimal ako fatalne chyby), rozsirena definicia ($a \geq 2$ b...)

struktura vrcholu a,b stromu, struktura vrcholu a,b stromu pre a-sort
algoritmus a-sort

algoritmus a-insert (oba skor slovami a pseudo-pseudo-kodom 😊, ale nastastie to bolo asi dobre..)

zlozitost, a kedze dokaz som nedal tak za 3

pisal som asi hodinu, potom uz som nemal co ponuknut tak som viacmenej ocumoval stukaturu dalsich 45min, kym prednasajuci nesiel okolo.

(3) Já jsem měl ten A-Sort, takže vám můžu podat přesné svědectví o tom, že na 3ku stačí pár obecných vět (třídící algoritmus vhodný pro předtříděné posloupnosti...), popis algoritmu (klidně svými slovy, hlavně aby to bylo správně, můžete kreslit obrázky a on asi bude jedině rád 😊) a jeho složitost. Zeptal se mě jenom na to, co myslím tím "předtříděné posloupnosti" a stačila odpověď "počet inverzí ve vstupním poli", což je stejně součástí složitosti algoritmu.

- Složitost
- Algoritmus slovne + obrazky
- na 3ku stačí pár obecných vět (třídící algoritmus vhodný pro předtříděné posloupnosti...), popis algoritmu (klidně svými slovy, hlavně aby to bylo správně, můžete kreslit obrázky a on asi bude jedině rád 😊) a jeho složitost. Zeptal se mě jenom na to, co myslím tím "předtříděné posloupnosti" a stačila odpověď "počet inverzí ve vstupním poli", což je stejně součástí složitosti algoritmu

<http://code.google.com/p/datove-struktury/downloads/>

<http://forum.matfyz.info/viewtopic.php?f=206&t=5035>

<http://forum.matfyz.info/viewtopic.php?f=206&t=5008>

<http://forum.matfyz.info/viewtopic.php?f=206&t=4153>

+tistena skripta

normální BVS

zadny zaznam ☹

✓ AVL (!)

(?) Já jsem měla AVL-stromy. Hodinu jsem psala, napsala jsem odhad na hloubku a všechny případy pro insert a delete. Trvalo mi to asi hodinu, pak si to Koubek 15 minut četl. Vadilo mu, že jsem si v definici hloubky podstromu zapoměla dodefinovat 0 pro prázdný podstrom, jinak si nestěžoval. Můj dojem je, že člověk má tolik času, kolik potřebuje, takže je lepší toho dobře využít a napsat všechno přesně. **Nevadí, když použijete jednodušší odhad třeba s horšími konstantami, ale když už něco píšete, tak je to třeba napsat dosti přesně.** Jinak Koubek se mi zdál při zkoušení velmi příjemný a nápomocný.

- Pravidla
- Hloubka
- Insert
- Delete
- napsala jsem odhad na hloubku a všechny případy pro insert a delete
 - si v definici hloubky podstromu dodefinovat 0 pro prázdný podstrom

✓ RB-stromy (!!)

"me už začíná otravovat, jak mi vsichni kreslí rotace bez popsání vrcholu"

(2) Dal jsem dohromady definici, strukturu uzlu/listu, popsal jsem mu (slovně a z velkého nadhledu!) všechny algoritmy (našel mi drobnost ve splitu a joinu), popsal, jak se to má s výskou stromu (zase žádné formální dokazování, slovně to úplně stálo), odtud jsem jen napsal, že plyne i složitost všech operací (u splitu jsem napsal jen náznak, že to je díky vlastnostem joinu a nějakou tu sumu jsem tam zvečnil). Co byla ale totalní tragédie, bylo vyvazování. U insertu jsem dal dohromady tak 2 případy z cca 4, u delete sotva dva, a to ještě některé s chybami. Říkal, že se mu začátek líbil, a ještě by to chtělo dodelat tohle. Já mu řekl, že mám vůbec problém vymyslet, jaké by případy to mám zkoušet resit - tak se mnou sednul, dali jsme dohromady, jaké jsou ty špatné konfigurace, a já mu na nich pak ukázal, jak se toho rotacíma či prebarvením zbavit. Občas jsem něco udělal špatně (člověk se přehlídně) nebo jinak, než ve slajdech (neefektivně - tady pozor, některé případy jdou resit jak prebarvením, tak rotací - a je nutné zvolit prebarvení, protože máme vetu, že insert se při rotaci zastaví, což ale v těchto případech zrovna nevycházelo).

Protože někdo před zkouškou hlásil, že se vyhazuje, když člověk umí cokoli, ale na vyvazování si nevzpomene, tak už jsem jen přemyslel, proč mě trápí. A pak najednou přišel s tím, že na jedničku už to není :-D Takže za dva a odcházel jsem šťastněj jako blecha =)

Moje resume - pan profesor dokáže něco odpustit, ale neznalosti základních věcí nepromíjí (říkal, že vyhodil člověka, protože mu předvadel rotaci, ale tu rotaci dělal špatně - tedy vyvazování už takový základ není, ale rotace, ty jo!). Když tuší, že se ve vás skrývají nějaké vědomosti, pídí se po nich (to vyvazování ze mě pácil, když už jsme tam po 3 hodinách zbyli sami, ještě další půlhodinu). A rozhodne mu nejde o nějaké formálnosti, zkouma, jestli chápete, co děláte (pravda, já zrovna neměl žádné výpočty, tak nevím, jak je to u nich). Na druhou stranu, pokud se něco, co nechápete, naučíte na paměť (:-D) a napíšete to na papír dobře, tak se k tomu asi vracet nebude.

(3) Já jsem dostal RB stromy. Hloubku jsem měl za chvíli, member taky 😊. Insert jsem napsal, ale zapomněl jsem na jeden z případů, po upozornění jsem to po chvíli opravil...). Celkem mi to ale docela dlouho trvalo vymyslet (učil jsem se to stylem insert diskuze podle strýce, delete diskuze podle bratra). Koubek trochu pospíchal (a pravda, trvalo mi to), takže mi na mazání nenechal moc času.. Dal mi trojku s tím, že jsem o ni bojoval, což mi přide až moc přísný, na 1 jsem to sice neuměl (bůhví na co jsem to uměl, ale za ty stromy bych si tu 2 dal..), ale stejně bych to časem vymyslel.

(?) Asi hodinu jsem psal. Dal jsem definici, odvození logaritmické výšky a pak insert + delete s vyvazováním. Dělal jsem to rozbořením případů, musím se neskromně přiznat, že jsem to fakt uměl, takže jsem měl všechny varianty a navíc správně. ;) Akorát mě pak zaskočil otázkou, proč při delete-vyvazování uvažuju jen černý vrchol v. Proč prý není červený? To ve skriptech myslím nebylo a že na to prý musím přijít. Odpověď: kdyby byl v obarvený červeně v (T,v) 3-pracovním cc stromě, tak ho stáčí prebarvit na černý a je po problému! ;) Pak mi řekl o indexu a na další stránku, kde jsem měl třeba popsány join se už ani nepodíval. Takže huráaaa! Jinak pan zkoušející je vážně v pohodě. Teda ne v tom smyslu, že by snad známkování bylo mírné, ale u každého se vždy na 5 - 10 minut zastaví a případně dá čas a sanci na další promyslení (+co jsem si všiml, tak na trojku nechce žádné prisernosti). Z toho ale vyplývá poměrně dlouhá doba zkoušky, myslím,

ze ti poslední tam zakysnou na hodně dlouho.... Učení: cca 3 dny lehcího učení a 3 dny fest srovnání (tj. v mém případě asi 7 hodin čistého času denně). Teprve poslední/předposlední den se začal dostavovat efekt "Ahaaa! Ted už je to jasny!". Zato jsem schrupal prakticky vše, včetně univerzálního a perfektního hesování (který jsem taky chtěl dostat) - bez toho a bez "složitých" důkazů se to snad da stihnout o den dva rychleji.

(1) Definice, insert + celý algoritmus, delete + celý algoritmus, jelikož jsem to trochu motal ale nakonec domotal, tak se zeptal ještě na split a join. To byla improvizace, chybovost odpovědi asi 50% ale vždycky jsem to nějak opravil, nakonec mi popral hezke svatky a napsal výborne do indexu

(3) Já som mal RB-stromy, som napísal definíciu, že výška je $O(\log n)$ s vysvetlením, že algoritmy sú rovnake ako u binárnych vyhľadavacích stromoch, join ako v (a,b) strome. Insert check - bez chyby (3 stromy a ich úprava + vysvetlenie, keď nastáva problém), Delete-check - som spravil zle, iba som vysvetlil, čo nastáva a keď zle. Po tom delete-check sa už nič nepýtal, iba že 3.

(1) teda pohoda, až na to, že som sa to naučil totalne inak ako bolo na prednáškach. Koubek sa v tom nejak stratil 😡 ale po chvíli bádania konštatoval, že to funguje.... Nebyť chyby u insertu 😊 snad by dal aj za 1 ale takto za 2. U RB netreba žiadne dokazy - staci Insert, Delete + definícia a dat to všetko správne na prvý sup a asi máte za jedná...

(3) - blablabla o tom, že jsou RB stromy nejefektivnější a proč jako + drobné srovnání s AVL..

- hlavní definici, jak vypadá RB strom...

- odvodil jsem maximální výšku, tj. $4 + 2 \cdot \log n$...

- pak jsem se vrhnul na popis algoritmu KONTROLA-INSERT a KONTROLA-DELETE... kreslil jsem jen obrázky, fakt nemá smysl se u těch stromů patlat s kódem.. Při těchto popisech jsem taky uvedl, co je to 2-parciální RB strom a 3-parciální RB strom... Jenže skoro vůbec jsem nevedel KONTROLA-DELETE... jsem vul, nejak jsem to prostě večer před zkouškou nedocetl...

- na závěr jsem napsal, že to má složitosti řádově $\log n$ a že je u RB stromů fajn, že se rotace u DELETE nevolají tak často jako v případě AVL stromu

Pan Koubek na to všechno koukal.. pomalu procítal a přikyvoval.. no ale pak došel k tomu mazání ze stromu a byl prusek.. snažil se to že mě vytáhat.. a tak říkal: "Tak si rozeberme případy: co když barva tohoto vrcholu bude červená..? přemyslejte!" a já jsem mlčel a přemyslel hlavně o tom, kdy konečně řekne, že to je teda jedno a že mám dejmetomu za dva nebo za tři 😊 ale nakonec jsme vlastně spolu rozebrali všechny případy toho mazání a teprve pak mne pustil se slovy: "Ale musím vám to dát asi jen za tři" nebo tak nějak, což jsem až podezřele radostně odsouhlasil

(3) Já som mal R-B stromy. Celkovo som sa učil tak 4-5 dní, pri opakovaní som ale tie rotácie a prefarbovania preskakoval, pretože sa mi to zdalo jasné a že keby som si to nepamätal, tak to tam odvodím. Na skúške som ale čumel do papierov a moc sa mi nedarilo. Základné veci boli OK, mal som ale chyby u INSERT a SPLIT, u DELETE to bolo ešte horšie. Nakoniec za 3 s odretými ušami a veľkou Koubkovou pomocou. Čas - 4,5h.

(1) Dokazal jsem logaritmickou výšku, nakreslil a posleze napsal v pseudokódu vyvazování pro **insert** a **delete**. Slovně jsem ještě musel vypočítat **join** a **split**. Po dvou a půl hodinách jsem odcházel (docela unavený), tak tak jsem stihl odjezd na hory.

(4) Třeba já měl u RB stromů vše správně i s důkazama, jenom u delete jsem měl slovně popis (předávání černé nahoru, s dvojitě černou barvou, nebo ukončením na červené) a pak 3 konkrétní ukázky stromů, kde se černá propaguje nahoru nebo kde se zastaví, ale neměl jsem úplný algoritmus (správně je rozbor případů podle barvy bratra a synovců, koukám z pohledu uzlu, co má dvojitě černou barvu). čekal jsem 2 nebo 3, ale vyhodil mě, že tohle je prej základ. To pak nevím, co člověk může mít špatně na trojku.

- Definice
- Hĺoubka
- Insert
- MEMBER
- DELETE
- JOIN, SPLIT (na 1-2)

Haldy

<http://forum.matfyz.info/viewtopic.php?f=206&t=4125>

✓ d-regulární

zadny zaznam ☹

✓ Leftist haldy (!)

(1) Ja osobne jsem dostal Leftist haldy, sice jsem pokazil definici $npl(v)$ (delka nejkratsi cesty z vrcholu v do vrcholu s nejvyse jednim synem), ale ostatni jsem mel dobre, tak jsem si odnesl 1 😊 Koubek byl hrozne mily a snazil se me ke spravne definici dohnat, i kdyz marne 😊

(?) Měla jsem leftist haldy. Chtěl slyšet definici operace MERGE a DECREASE. Naopak mi připadal dosti trpělivý (na to kolik jsem toho uměla a jak pomalu to ze mně lezlo) chyby nechal opravit, nejasnosti vysvětlil.

(1) Já jsem dostal Leftist haldy. Stačilo napsat definici + popsat operace a odcházel jsem s jedničkou v indexu

- Definice
 - $npl(v)$ (delka nejkratsi cesty z vrcholu v do vrcholu s nejvyse jednim synem)
- popsat operace
 - operace MERGE a DECREASE

<http://people.cs.clemson.edu/~pargas/courses/cs212/common/notes/ppt/14LeftistSkewHeaps.ppt>

✓ Binomiální haldy

zadny zaznam ☹

✓ Fibonacciho haldy (!)

(2) Napsal som vlastne vsetko co je v skriptach (ale dost chaoticky) - definiciu, operacie, ich zlozitosti, amortizovane zlozitosti (zlozitosti som okomentoval ked som si myslel ze to nie je zjavne) a lemmu a vety s dokazmi (o pocte synov vrcholu). Finalny dokaz kde sa odhaduje prave pocet synov v som si nepamatal, takže som ho odhadol jednoduchsie ako je v skriptach (takže viac nahrubo). Koubek si tie moje papiere precital a nebol uplne spokojny s definiciou haldy - nenapsal som ze korene stromov nemozu byt oznacene. Nechal ma si to rozmysliet ale neprisel som na to. Potom mi ukazal moj vyvazovaci algoritmus a uz to zo mna vytiahol. S ostatnym viac menej suhlasil a dal mi za 2 so slovami ze by mi chcel dat za 1 alebo ze mi ju neda.

(2) No ja mel Fibonacciho haldy. Napsal jsem vsechny ty algoritmy, tvrzeni i dukazy (7 stran psani).

V dukazech jsem mel jen asi jednu dve maly chyby a nemohl jsem si nejak vzpomenout

(resp. presne domyslet) na amortizovany slozitosti vseh tech operaci, takže nakonec za 2.

- definiciu,
 - korene stromov nemozu byt oznacene
- operacie,
- ich zlozitosti,
- amortizovane zlozitosti (zlozitosti som okomentoval ked som si myslel ze to nie je zjavne) a
- lemmu a vety s dokazmi (o pocte synov vrcholu)

<http://www.cs.princeton.edu/~wayne/cs42...ci-4up.pdf>

✓ Haldy v Dijkstrovi (aplikace hald)

Algoritmus + složitosti s regulárnima a fibonacciho haldou

Třídění

✓ Quicksort (!!)

(3) Na trojku stacilo napísať algoritmus (oneliner v haskelli + celý pseudokód zo skriptu), najhorsi / priemernú zložitosť, povedať pár viet o výbere pivot-a. Koubek mi dal ešte možnosť pokračovať ďalej v odvodzovaní zložitosti, alebo si zapísať trojku, vybral som si druhú možnosť. Inak bol strašne milý po celý čas a daval všetkým jednoduché otázky, vedľa mňa to boli RB stromy, A-B stromy.

(2) Ke quicksortu (jsem nyní po 8+ vítěznych Plznicích, proto prosím omluvte souvislost projevu) -- popsal jsem algoritmus slovně, uvedl problémy s volbou pivotu (konstatoval jsem, že "pro libovolného pivotu existuje posloupnost, kde alg. pobíží v $O(n^2)$ ", což se doc. Koubkovi nezamlouvalo -- pokud jsem to v tuto hodinu schopen dát dohromady, jde cca. o to, že pro každý pevně zadrátovaný algoritmus volby pivotu existuje nějaký vstup, pro který alg. pobíží v $O(n^2)$, zatímco při náhodné volbě pivotu je takový čas nejhorší pro všechny vstupy), uvedl, že náhodná volba chce čas, a že se typicky bere median ze tří či pěti pevně daných prvků, konstatoval, že při zohlednění multiplikativní konstanty jde o nejrychlejší známý alg. pro třídění založený na vzájemném porovnávání prvků, uvedl aplikaci při hledání k-tého nejmenšího prvku, konstatoval, že časová složitost je v nejhorším případě $O(n^2)$ a v očekávaném $O(n \log n)$, aplikoval to na odhad potřebného prostoru, popsal algoritmus též formálně, zdůvodnil složitost v nejhorším případě ("pivot je největší/nejmenší prvek, tedy v každém kroku rekurze ubýde právě jeden člen posloupnosti"), lehce naznačil důkaz složitosti v očekávaném případě, kde jsem pohnoužil značení prvku -- dle Koubkova skriptu se v prvním, "pekném" důkazu hovoří v případě prvku x_{i0} i-tým prvkem ve výstupní posloupnosti, a vůbec jsem neměl rozhodovací strom o průběhu algoritmu,... -- popsal jsem ale

$$\sum_{i,j} x_{ij}$$

sumaci x_{ij} ... -> zkratka nakonec se urodila krásná a zdravá dvojka, která mě coby tatovi udělala velkou radost 😊.

(1) Napsal jsem:

- malé úvodní povídání (třídící algoritmus, vstup, výstup...)
- popis algoritmu (dva pointery, překřížení rekurse) - neměl jsem tam přesně vyřešené všechny případy, ale Koubek říkal že to není nutné.
- důkaz složitosti v průměrném případě: o to mu šlo především. Bohužel ani jeden ze dvou důkazů ve skriptech jsem nepochopil, naučil jsem se místo toho Čepkův jednoduchý důkaz z prváku. Koubek byl spokojen (byl podle mě rád, že vidí něco nového) a dal mi za jedna.

(2) Dostal som Quicksort, popísal som teda A4-ku nejakými základnými vecami, ako je definícia, nejaký jednoduchý pseudokód (tak, že "usporiadaj prvky podľa tak, že $\forall i < \text{pivot} : \text{pole}[i] \leq \text{pole}[\text{pivot}]$, $\forall i > \text{pivot} : \text{pole}[i] > \text{pole}[\text{pivot}]$ " bola jedna inštrukcia, t.j. žiadne porno s cyklami, if-mi a iteracnými premennými), $O(N)$ pamäťovú zložitosť, $O(N^2)$ v najhoršom prípade, $O(N \log N)$ v priemernom. Koubek mal výhrady voči tomu, že som quicksort nazval in-place sortovacím algoritmom. Po krátkej dispute som pochopil, že $O(N)$ zásobník (ktorý som explicitne spomenul v papieri) nepovažuje za in-place, aj keď sa nevytvára žiadne prídavné polia s prvkami. Dôkaz $O(N \log N)$ zložitosti v priemernom prípade som nevedel (čo Koubek zamyslel ohodnotiť trojkou), ale nejak som ukázal, že beh quicksortu je analogicky stavaniu náhodného binárneho stromu a pre ten som potom dokazoval, že bude mať logaritmickú hĺbku. Dôkaz z prednášky som nevedel, ale vyplodil som nejaké cosi, čo dokazovalo, že hĺbka náhodného binárneho stromu je v priemere $O(\log n)$, zrejme to nebolo správne, pretože to bolo dosť krátke, ale Koubek to nevedel vyvrátiť a po dlhšom špekulovaní nad rôznymi časťami dôkazu nakoniec prehlasil "tak ste vyhral, dám vám dvojku, ale chci si nechať tenhle papír" a dvojku mi skutočne dal.

Celkovo mám z Koubka pocit, že

* záleží mu na konstantách (medianový pivot u quicksortu mu z nejakého dôvodu nevyhovoval)

* posobil na mňa dosť ferovo -- vetou "ja vás nechci podceňovať, ale mnozí ľudia na ten dôkaz koukali a nepodarilo sa jim ho zkrátiť" mi celkom slušne naznačil, že môj dôkaz nemože byť správny, ozač mu absolútne neveril, ale nedokázal najst konkrétnu chybu, tak mi ho uznal

* je celkom prísny, ale nie je dôvod sa ho báť, občas niečo nevážne poznamenal, usmial sa

(1) Ja jsem mela quicksort. 😊 Algoritmus jsem popsala jen slovy, k tomu jsem pridala slozitosť v nejhorsim a v ocekavane-
nem pripade vctne odvozeni a par poznamek k vyberu pivota a ke korektnosti. Protoze jsem tam nemela chyby, dostala
jsem 1.

(3) som mu tam popisal tri strany, po 20 minutach som mu ich ukazal, na vsetko prikyvoval ze dobre dobre, a potom sa
ma spytal ci by som nejak vedel este doplnit preco je slozitosť v priemernom pripade nlogn, ja som nejak vahal, a pove-
dal ze inac mi to da za 3, ked som pocul tuto moznost uz som nevahal a zobral som si trojku 😊. Uff esteze to mam za
sebou.

- napisat algoritmus (oneliner v haskelli + cely pseudokod zo skript),
- najhorsiu / priemernu zlozitosť,
- povedat par viet o vybere pivot-a

✓ Adaptilní Mergesort s posloupnostmi rozdílné délky

✓ Rozhodovací stromy

(3) Já měl rozhodovací stromy a moc jsem tam toho nenapsal: jen definici a výpočet počtu porovnání a času v nejhorším
případě. Pan Koubek se mně zeptal, zda vím, jak se spočítá odhad pro očekávaný případ, to jsem moc nevěděl, tak mi to
obecně vysvětlil a řekl, že mi teda dá trojku

(3) Ja mel rozhodovaci stromy, .. nenapsal jsem mu tam nic (teda nic plodneho), protoze jsem tohle fakt necet a netusil,
co tam chce slyset, .. no ale nakonec me dokopal k tomu, ze se tam da ukazat ocekavany i nejlepsi pocet porovnani v
typu trideni, kde je porovnani dvou prvku jedina ziskatelna informace a dal mi to za 3.. sice nic moc, ale dalsi predmet
odbyty pro pristi rok se moc hodi;)

- definici a
- výpočet počtu porovnání a času v nejhorším případě
 - ukazat ocekavany i nejlepsi pocet porovnani
- algoritmy, ktoré "porušujú" dolný odhad zložitosti

<http://forum.matfyz.info/viewtopic.php?f=206&t=2689>

http://wiki.matfyz.cz/index.php?title=Datov%C3%A9_struktury_I#Doln.C3.AD_odhad_slo.C5.BEitosti

✓ Wordsort (!)

(1) Popsal jsem Bucketsort s casovymi slozitosťmi. Zdefinoval jsem lexikograficke usporadani. Wordsort jsem popsal
jednak slovy a po strane jeste takovym metakodem. Ten algoritmus je velmi jednoduchy (spis jsem se jistil, aby to p.
Koubek nepochopil spatne). Cetl to velmi peclive snad pet minut (pro mne takova mala vecnost) a nakonec mi dal 1.

(1) Dostal som WordSort. Povedal som, aky problem wordsort riesi (lexikograficke triedenie slov), ze je zalozeny na buc-
ketsorte, popisal som algoritmus vlastnymi slovami, ukazal korektnost a slozitosť. Bez doplnujucich otazok som sa pobral
domov aj ja.

- aky problem wordsort riesi (lexikograficke triedenie slov), ze je zalozeny na bucketsorte
 - Popsal jsem Bucketsort s casovymi slozitosťmi.
 - Zdefinoval jsem lexikograficke usporadani.
- Wordsort jsem popsal jednak slovy a po strane jeste takovym metakódom
 - ukazat korektnost a slozitosť

<http://forum.matfyz.info/viewtopic.php?f=206&t=4198>

http://wiki.matfyz.cz/index.php?title=TIN066_P%C5%99ih%C3%A1dkov%C3%A9_t%C5%99%C3%ADd%C4%9Bn%C3%A4D#Word_sort

Hledání

✓ Vyhledávání v uspořádaném poli (!!)

(3) Na trojku stačilo popsat obecné hledání, varianty funkce NEXT, jejich složitosti průměrně/nejhůř/s rovnoměrným rozložením. O důkazy složitostí jsem se ani nepokoušel.

(3) Sepsal jsem snad celkem přesně obecný algoritmus i všechny základní modifikace včetně "trislovných dukazu" složitosti (to je fakt trivialní, až na očekávané $O(\log \log N)$ u interpolacního, a to se zase zřejmě nedokazovalo a nezkousi vůbec). U zobecněného kvadratického jsem chvíli musel vzpomínat, jak je to s temi odmocninami, ale s matnými vzpomínkami se dal algoritmus uhadnout, jen jsem zapomněl, že unární kroky jdou tím směrem, kterým je prvek (ne vždy dopředu) - ale to jsem rychle opravil, když se mne na to zeptal. Největší problém byl se složitostí, u očekávané jsem si jen pamatoval, že se da ukázat, že v rámci jednoho interpolacního kroku je očekávaný počet těch ostatních konstantní, ale neuměl jsem to dokázat; tak mi nabídl, že mne "jeste necha se s tím morit", nebo mi da rovnou trojku, kterouz jsem s povdekem přijal. Dokonce mi pak při zapisování známky i docela srozumitelně vysvětlil, jak se to vlastně dokazuje, jdu to připsat na wiki.

(3) Já měl to hledání v uspořádaném poli. Korektní popis a pochopení funkce algoritmů a nějaká základní fakta/pozorování na trojku stačí, to ostatně doc. Koubek říkal už na přednášce. Já jsem k tomu u zobecněného kvadratického vyhledávání vyplodil ještě horní odhad počtu bloků $O(\log \log n)$ a z něj odhad složitosti v nejhorším případě $O(\log n)$. Tušil jsem, že by se dala nějak počítat i očekávaná složitost, ale nepamatoval jsem si vůbec jak (podle Čebyševovy nerovnosti). Nevím jestli by stejně přísný u těžších otázek (tahle byla myslím objektivně jedna z těch snažších), ale na dvojku to nestačilo. Ještě mi dal možnost si to zlepšit amortizovanou složitostí operací na (a,b)-stromech... 😊, takže za tři.

(3) Dostal som vyhladavanie v usporiadanom poli, napisal som vsetko az po konkretny popis next(d,h) pri zobecnenom kvadratickom vyhladavani. To stacilo na 3, na lepsiu znamku by to chcelo presny popis algoritmu a nejake tie dokazy zlozitosti

(2) Moje téma bylo docela v pohodě. Popsal jsem obecný princip, různé metody výpočtu funkce "next" s očekávanými časy a podrobně napsal algoritmus obecného kvadratického vyhledávání (pseudokód). Pak se mě Koubek ještě ptal, proč je očekávaný čas maximálně $O(\log n)$ a průměrně $O(\log \log n)$. $O(\log n)$ jsem ukázal konkrétním případem: na posloupnosti typu 1, 2, 3, 4, ..., atd a (hafo strašně moc veliké číslo) jako poslední prvek ($A[n] \gg A[n-1]$) algoritmus degeneruje k normálnímu binárnímu prohledávání. K $O(\log \log n)$ jsem řekl základní myšlenku, čili že interval, ve kterém se hledá, se zmenšuje s odmocninou. Ukázal jsem, že trvá $\log \log n$ dlouho, než se odmocňováním dostanu na konstantu (snadný důkaz na 4 řádky) a dál jsem to neuměl, nechtělo se mi to vymýšlet a chtělo se mi na záchod 😎

- Korektní popis a pochopení funkce algoritmů a nějaká základní fakta/pozorování na trojku stačí
- popsat obecné hledání,
- varianty funkce NEXT,
 - jejich složitosti průměrně/nejhůř/s rovnoměrným rozložením (bez dk)

✓ hledání k-teho nejmenšího prvku

(4) hľadanie n-tého prvku (dotyčný nevedel, tak mu vyslovene povedal, že chce FIND a SELECT, on to asi skúšal nejak vymyslieť)

- chce FIND a SELECT