

ALGORITMY A DATOVÉ STRUKTURY 1 (NTIN060)

Cvičení 22. 2. 2016

ÚVOD

Principy cvičení.

- **Vy** se chcete něco naučit.
- **Já** vám k tomu vytvářím prostor.
- Proto je nejdůležitější, co potřebujete **vy**:
 - ... pochopit něco z přednášky
 - ... pochopit příklad z minulého cvičení
 - ... pochopit domácí úkol atd.
- **Nebojte se** klást otázky, diskutovat s ostatními, chodit po místnosti,...
- Pár obecných **tipů**:
 - Když si nevíte rady, zkuste si hrát s příklady. (Neumíte dokázat tvrzení pro obecná n a k ? Vyzkoušejte si, co vlastně říká pro nějaké malé hodnoty.)
 - Když si pořád nevíte rady, poraďte se se sousedy. Nejdřív jim vysvětlete, čemu **rozumíte** a čemu **nerozumíte** (samotným vám to pomůže uchopit jádro problému). I pokud si sousedé též neví s úlohou rady, můžou vám pomoci s tím, kde jste se zasekli vy.
 - Když si pořád nevíte rady, přivolejte mě, snad budu umět pomoci :)
 - Řešení příkladů budete prezentovat vy – vaše myšlení si je bližší!

Technické detaily.

- Pište mi na koutecky@kam.mff.cuni.cz, předmět mailu začněte [ADS1_2016].
- Web cvičení je na <http://research.koutecky.name/teaching/>
- Na zápočet potřebujete nasbírat 100 bodů, ty můžete získat za:
 - **řešení** domácích úkolů,
 - **opravování** domácích úkolů spolužákům,
 - **zápočtovou práci**
 - (*Naopak nevyžadují prezenci – pokud se toho více naučíte jinde/jindy/jinak, klidně na cvičení nechoďte.*)
- Úkoly přijímám **pouze** elektronicky, buď jako prostý text (tzn. přímo v mailu), nebo jako PDF (ideálně vyrobené (La)TeXem, ale to není povinné, jen silně doporučené). **Nechci** formáty .doc, .docx, .odt atp. Pokud mi pošlete řešení úkolu v tomto formátu a já to zjistím až po termínu odevzdání úkolu, máte smůlu.
- Úkolů bude vypsáno za cca. 170 bodů, nebudou zadávány každý týden, spíš cca. každých 14 dní. Po následující dva týdny od zadání za ně bude možno získat plný počet bodů, poté až do konce semestru můžete úlohy odevzdávat za polovinu bodů.
- Zápočtová práce může být praktická (implementace algoritmu) či teoretická (popis algoritmu včetně pseudokodu, důkaz správnosti, analýza časové složitosti) a bude za ni možno získat podle náročnosti 20-40 bodů. Pár návrhů bude dostupných na webu, ale nejlepší je, když přijdete sami s něčím, co vás zajímá, rád vám vyjdu vstříc.
- Pokud budete potřebovat **konzultaci**, zkuste se nejdřív zeptat spolužáků. Když vás bude víc, napište mi mail nebo za mnou přijďte po cvičení, domluvíme se na konkrétním čase a místě.

email: koutecky@kam.mff.cuni.cz, url: <http://research.koutecky.name/>.

~

PŘÍKLADY

Algoritmy. Novinář má za úkol za rok napsat reportáž o pracovních podmínkách v jedné nejmenované firmě. Musí tedy vyzkoušet co nejvíce pracovních pozic. Chce ale, aby se mu neustále zvyšoval plat. Firma v různých časech vypisuje pracovní místa.

Řečeno matematicky, máme zadánu posloupnost p_1, \dots, p_n reálných čísel a hledáme v ní nejdelší ostře rostoucí vybranou podposloupnost.

Jak můžeme takový problém řešit?

- Z definice
- Rekurzivně
- Rekurzivně s pamětí
- Jako grafový problém
- S chytrou datovou strukturou

Složitost.

- Co a proč zanedbáváme?
- Co znamená slovy $O(n)$, $\Omega(n)$, $o(n)$, $\omega(n)$? A jak to napsat **formálně** (tzn. s kvantifikátory, jako formuli)?

Jak tvrdě je vajíčko? Máme N -patrový dům a víme, že hodíme-li vajíčko z alespoň K -tého patra, rozbije se. Známe N a chceme zjistit K . Jak to udělat na co nejméně pokusů, máme-li k dispozici jediné vajíčko? Dvě vajíčka? Tři? Libovolně mnoho vajíček? (Pokus = hození vajíčka z nějakého patra.)

Největší díra v matici. Mějme matici velikost $N \times M$. Díra v matici je souvislá podmatice samých nul. Jak co nejrychleji najít tu největší díru v zadané matici?

Nejlepší a nejhorší případ. Jaké znáte třídící algoritmy? Insert, Select, Bubble – zkuste rozmyslet, jaká je jejich složitost v nejlepším a nejhorším případě a jak takové případy vypadají.

Další příklady... Tyto příklady jsou vybrány z projecteuler.net, který doporučuji, chcete-li se naučit prakticky algoritmizovat (a zároveň se na tom třeba můžete naučit nový programovací jazyk).

Mocniny 3 a 5. Jak co nejrychleji vypsát n -té nejmenší číslo tvaru $3^i 5^j$?

Lexikografické permutace. Permutace lze uspořádat lexikograficky, např. permutace čísel 0, 1 a 2 mají toto pořadí:

012 021 102 120 201 210

Jak co nejrychleji najít miliontou permutaci množiny $\{0, \dots, 9\}$?

Trojúhelníková čísla s hodně děliteli. i -té trojúhelníkové číslo je číslo vzniklé součtem $1 + 2 + \dots + i$. Jak co nejrychleji najít první trojúhelníkové číslo, které má více než k dělitelů? Pro představu, prvních 7 trojúhelníkových čísel je na tom s děliteli takto:

1: 1
 3: 1, 3
 6: 1, 2, 3, 6
 10: 1, 2, 5, 10
 15: 1, 3, 5, 15
 21: 1, 3, 7, 21
 28: 1, 2, 4, 7, 14, 28

Nejdelší Collatz. Zavedeme jednoduché pravidlo, které pro každé číslo dá nějaké další číslo, a takto získáme jistý řetězec.

$n \rightarrow n/2$ je-li n sudé, $n \rightarrow 3n + 1$ je-li n liché.

začneme-li s číslem 13, dostaneme tuto posloupnost:

$13 \rightarrow 40 \rightarrow 20 \rightarrow 10 \rightarrow 5 \rightarrow 16 \rightarrow 8 \rightarrow 4 \rightarrow 2 \rightarrow 1$

(Tzv. *Collatzova domněnka* říká, že ať začneme v jakémkoliv čísle, dojdeme nakonec do jedničky. Zatím se jí nikomu nepodařilo dokázat – čeká na vás!)

Jak co nejrychleji přijít na to, které číslo pod jeden milion vyprodukuje nejdelší řetězec, než dojde do jedničky?