PAL: 1. cvičení

T. Sieger

23. 9. 2021

Př. 1/1: "Big O" notace

Ověřte, že platí $\forall n \in N : n > 4 \Rightarrow n^2 - 2n > 0.5n^2$.
Graf funkce $f(n) = 0.5n^2$ pro $n > 4$ tedy leží vždy pod grafem funkce $g(n) = n^2 - 2n$ a navíc rozdíl $g(n) - f(n)$ roste do nekonečna s rostoucím n.
Dokažte pomocí definice množiny $O(f(n))$, že i přesto platí $g(n) \in O(f(n))$.

Nerozhodnutelné

$2^{2^{\cdot^{\cdot^{\cdot^{2^k}}}}}$

2^k

Nezvládnutelné problémy	NP–úplné problémy
Zvládnutelné problémy	Třída P
n^k	
	Lineární programování
n^3	
	Násobení matic
n^2	
$n \log n$	Třídění
n	Vyhledávání

Př. 1/2*: Růst funkcí

Symbolem lg značíme logaritmus o základu 2. Uspořádejte podle řádu růstu uvedené funkce proměnné n. Zdůvodněte pořadí každých dvou sousedních funkcí v tomto uspořádání.

a: $\lg(n!)$
b: $(\sqrt{2})^{\lg(n)}$
c: $2^{\lg(\lg(n))}$
d: $4^{\lg(n)}$
e: $\sqrt{\lg(n)}$
f: $n\lg(n^2)$
g: $n\lg(n)$
h: $(\lg(n))^2$

Př. 1/3: Průchod polem

Algoritmus A projde celým polem délky N a prvek s indexem k zpracuje za ck milisekund. Konstanta c je stále stejná. Určete asymptotickou složitost zpracování celého pole.

Př. 1/4: Průchod maticí

Algoritmus P projde celým dvourozměrným polem velikosti NxN a prvek na pozici (k, m) zpracuje za:
A) $c(k+m)$ milisekund,
B) ckm milisekund.
Konstanta c je stále stejná, $1 \leq k \leq N, 1 \leq m \leq N$. Určete asymptotickou složitost zpracování celého pole v případě A) a B).

Př. 1/5: Převody grafových reprezentací

Popište jednotlivé reprezentace orientovaného grafu v paměti počítače, které znáte. Pro každou možnou dvojici reprezentací R1, R2 určete, jaká je asymptotická složitost vzájemného převodu.

Př. 1/6*: Hledání maximální komponenty grafu

V seznamu je uložena množina všech hran grafu, každá hrana je dvojice $< uzel, uzel >$. Víme, že graf má N uzlů, že je nesouvislý a že obsahuje komponentu K, která má více než $N/2$ uzlů. Máme vytvořit nový seznam obsahující (ve stejném formátu) právě všechny hrany komponenty K. Popište, jak co nejefektivněji budete tuto úlohu řešit a jaká bude asymptotická složitost vašeho řešení. Pořadí hran v obou seznamech není předepsáno a může být libovolné.

Př. 1/7*: Mocnina grafu

Druhá mocnina grafu G je graf G^2 , jehož množina uzlů se shoduje s množinou uzlů grafu G a jehož množina hran je určena takto: G^2 obsahuje hranu u, v jen a jen tehdy, pokud G obsahuje zároveň hrany u, w a w, v, kde w je libovolný uzel grafu G. Jinými slovy, G^2 vznikne z G tak, že do G přidáme hrany mezi všemi uzly spojenými cestou délky 2 a odstraníme původní hrany. Popište, jak vytvoříte G^2 , když jsou grafy zadány (a) spojovou reprezentací (b) maticí sousednosti. Která varianta bude rychlejší?

Př. 1/8: Porovnání algoritmů

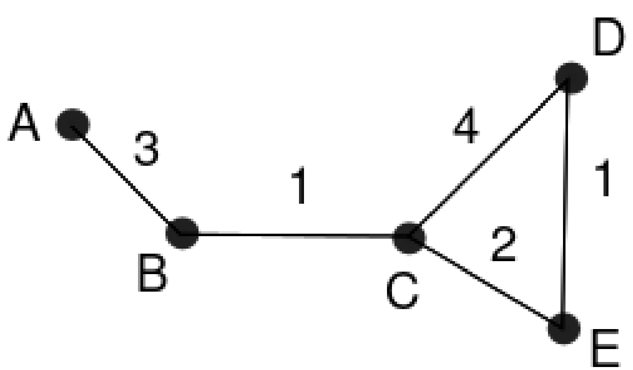
Máme dva algoritmy $A1$ a $A2$ zpracovávající obyčejný neorientovaný graf s n uzly a m hranami. Oba algoritmy řeší tutéž úlohu a vydávají stejný výsledek na všech vstupech. Asymptotická složitost $A1$ je $O(n\ m\ log(n))$, asymptotická složitost $A2$ je $O((n^2 log(m))$. Diskutujte, kdy je výhodnější užívat $A1$ a kdy $A2$.

Př. 1/9: DFS/BFS se sekvenčním přístupem

Předpokládejte, že máte k dispozici neorientovaný graf $G = (V, E)$, který je reprezentován seznamem hran. Seznam hran není nijak uspořádán a přístup k jeho jednotlivým prvkům je pouze sekvenční (k prvkům nelze přistupovat pomocí indexu). Určete, jaká je za těchto okolností asymptotická složitost algoritmů BFS a DFS.

Př. 1/10: Hledání minimální kostry

Napište pseudokód Jarníkova algoritmu, určete jeho časovou složitost a najděte pomocí něj minimální kostru následujícího grafu:



Př. 1/11: Hledání minimální kostry

Napište pseudokód Kruskalova algoritmu, určete jeho časovou složitost a najděte pomocí něj minimální kostru následujícího grafu: