Asymptotická složitost

1. Ověřte, že platí (∀*n* ∈ **N**) ( *n* > 4 ⇒ *n*2 − 2*n* > 0.5*n*2 ). Graf funkce *f*(*n*) = 0.5*n*2 pro *n* > 4 tedy leží vždy pod grafem funkce *g*(*n*) = *n*2 − 2*n* 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*) ∈ O(*f*(*n*)).

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

     *n*∙lg(*n*2) 

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

4. Algoritmus P projde celým dvourozměrným polem velikosti N×N a prvek na pozici (*k*, *m*)  zpracuje za

A) *c*⋅(*k*+*m*), B) *c*⋅*k*⋅*m* milisekund. Konstanta *c* je stále stejná, 1 ≤ *k* ≤ N, 1 ≤ *m* ≤ N. Určete asymptotickou složitost zpracování celého pole v případě A) a B).

Reprezentace grafů

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

6. 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.

7. Druhá mocnina grafu G je graf G2, jehož množina uzlů se shoduje s množinou uzlů grafu G a jehož množina hran je určena takto: G2 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, G2 vznikne z G tak, že do G přidáme hrany mezi všemi uzly spojenými cestou délky 2 a odstraníme původní hrany. Popište, jak vytvoříte G2, když jsou grafy zadány (a) spojovou reprezentací (b) maticí sousednosti. Která varianta bude rychlejší?

8. 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 Θ(*n* *m* log(*n*)), asymptotická složitost A2 je Θ(*n*2 log(*m*)). Diskutujte, kdy je výhodnější užívat A1 a kdy A2.

BFS DFS

9. 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.

10. Když má daný graf *n* uzlů a Θ(*n*2) hran, potom asymptotická složitost algoritmu DFS je Θ(*n*2), za předpokladu, že během provádění algoritmu máme přístup v konstantním čase ke každému právě zpracovávanému uzlu a ke každé právě zpracovávané hraně. Určete, jaká bude asymptotická složitost DFS, pokud doba přístupu ke každému uzlu bude ve třídě Θ(*n1/2*) a doba také přístupu ke každé hraně bude ve třídě Θ(*n1/2*).

Řešte tuto úlohu také pro algoritmus BFS.