Programování pro matematiky
9. cvičení - Grafy

Peter Kovács

Doporučuje si promyslet řešení každého z úkolů, pro vaše vlastní ujasnění učiva. Navíc pokud některé z úvah sepíšete můžete získat body

Každý úkol obsahuje za názvem maximální počet bodů, které lze za úkol získat. Vaším úkolem je si vybrat úkoly, které chcete řešiť. Klidně všechny. Úkoly, které jste si vybrali sepište do jednoho souboru a odvzdejte do recodexu. Za celý úkol můžete získat maximálne **4 body**. Po odvezdání ohodnotím každou odevzdanou úlohu. Body sečtu a přidelím vám $min(body, 4)$. Úkoly řešte sami. Pokud spolupracujete v skupině dostanete své body podělené počtem lidí v skupině.

Za úkoly je možné udělit pouze celočíselné body. Pokud získáte neceločíselný výsledek, bude zaokrouhlen nadol.

Reprezentácia multigrafov (1 bod):

Na cvičení sme si ukazovali různé způsoby reprezentace grafu. Navrhnite reprezentaci multigrafu v paměti. Udělejte návrh pro ohodnocený a neohodnocený multigraf.

Řešení: Jedna možnost pro neohonocený multigraf je vzít matcici sousednosti a místo 1 značíci extistuje hrana na toto místo uložím počet hran. Pro ohodnocený graf můžu mít pro každou případnou hranu seznam délek hran. ideálne seřazných dle velikosti.

Reprezentace řídkých grafů (1 bod):

Navrhněte reprezentaci grafu, která bude efektivní pro řídké grafy, a přitom dokáže rychle testovat existenci hrany mezi zadanými vrcholy.

Řešení: Hrany ukládame do hešovací tabulky. Vzhledem na řídkost grafu, by nemělo být kolizí mnoho. Dotaz na existenci hrany bude v O(1).

Význam mocnin matic sousednosti (1 bod):

Je-li A matice sousednosti grafu, co popisuje matice A^2? A co A^k? (Mocniny matic definujeme takto: $A^1 = A$, $A^k = A^{k-1}$.)

Řešení: Když si vyjádříme jak vypadá prvek v matici A^2 dostaneme

$$A^2_{ij} = \sum_{k \in V} A_{ik} \cdot A_{kj}$$

Uvědomme si, že jednotlivé členy v sumě budou rovny jedné právě tehdy, když pro dané k jsou A_{ik} i A_{kj} rovny jedné a tedy existuje hrana z i do k a zároveň z k do j. Suma tedy spočíta počet cest z i do j o délce dva. Kdyže si stejnopu věc rozepíšeme obecně pro A^k můžeme pozorvat, že na pozici ij bude počet sledů z i do j. Pozor ne cest, protože sledy s opakujícími se hranami započítávame také.

Počet nejkratších cest (1 bod):

Upravte BFS tak, aby pro každý dosažitelný vrchol zjistilo, kolik do něj vede nejkratších cest z počátečního vrcholu. Zachovejte časovou složitost $O(n + m)$, kde m je počet hrán a n je počet vrcholů.

Řešení: Procházíme graf BFS přičem každý vrchol zařazujeme do fronty jako dvojicu (vrchol,vzdialenost). Pokud vybereme vrchol z fronty, podívame sa jestli je navštíven. Pokud není uložíme do pole dvojici

(1,vzdálenost), kde 1 značí počet nejkratších cest. Pokud už byl nalezen skontrolujem jestli se vzdálenost shoduje. Pokud ano připočítame 1 do počtu nejkratších cest.

Uřezávání grafu (1 bod):

Mějme souvislý orientovaný graf. Chceme mazat jeho vrcholy jeden po druhém tak, aby graf zůstal stále souvislý. Jak takové pořadí mazání najít? Popište algoritmus.

Řešení: Spustíme vyhledávání (DFS nebo BFS). Z neho dostaneme poťadí vrcholů v jaké sme jich zpracovávali. Pokud toto pořadí otočíme, dostaneme chtěné pořadí. Stačí si uvědomit jak vypadá strom průchodu grafem.