Analýza koster

Příklad 1: Opakování: Dokažte nebo si připomeňte:

- Modré lemma: Nejlehčí hrana libovolného řezu je v min.kostře
- Červené lemama: Nejtěžší hranana cyklu není v min. kostře
- Dichotonie: Pokud jsou váhy hran různé, každý hrana je buď "modrá" nebo "červená"

Příklad 2: Neunikátní váhy: Určete, které z algoritmů Kruskal, Jarník/Prim, Borůvka selžou pokud hrany v grafu nemají unikátní váhy. Navrhněte, jak tomu předejít.

Hint: k předejití problému nám stačí mít libovolné lineární uspořádání hran

Příklad 3: (+) Inverzní Kruskal: Inverzní Kruskal probírá hrany od nejtěžších. Pro každou hranu e odstraní e z grafu pokud $G \backslash e$ je nesouvislý. Ukažte, že zbylé hrany tvoří minimální kostru.

Příklad 4: Počty hran: Pro graf G a dvě hodnoty $a < b$ určete kolik hran váhy alespoň a, ale nejvýše b bude v minimální kostře. Hledáme lineární postup.

Hint: z Kruskalova alg. lze nahlédnout, že výběr konkrétních hran ve zkoumaném rozmezí nezávisí na tom, které hrany nižších vah byly vybrány

Jarník

Příklad 5: Struktury: Analyzujte složitost Jarníkova/Primova algoritmu s využitím binární a Fibonacciho haldy.

Hint: Všimněte si, že pseudokód Jarníkova algoritmu je téměř totožný známému algoritmu na nejkratší cesty

Kontrakce

Příklad 6: Obecná kontrakce: Pro graf G s vyznačenou množinou hran F (hrany na sobě mají značky) zkonstruujte kontrakci G/F. Chceme dosáhnout lineárního času.

Příklad 7: Kontrahující Borůvka: Navrhněte verzi Borůvkova algoritmu, která po každé fázi zkontrahuje hrany vybrané do kostry. Bude třeba ošetřit smyčky a násobné hrany. Chceme složitost $\mathcal{O}(m \log(n))$

Příklad 8: Rovinné grafy: Ukažte, že pro rovinné grafy kontrahující Borůvka najde minimální kostru v lineárním čase.

Hint: výsledkem kontrakce na rovinném grafu je opět rovinný graf

Domácí úkol 4:

Mějme na vstupu souvislý graf G s unikátními váhami hran w a předpočítanou minimální kostru T.

Navrhněte algoritmus, který najde (nějakou) druhou nejlehčí kostru.

V tomto úkolu nás bude zajímat především myšlenka návrhu. Chceme tedy hlavně popis postupu, a argumentaci jeho korektnosti. Pseudokód není nutný. Algortitmus by se měl vejít do časové složitosti $\mathcal{O}(m^2)$ (lze přímočaře dosáhnout $\mathcal{O}(nm)$).

Hint: Přímočarý přístup ke korektnosti já třeba následující. Jakákoliv ne-minimální kostra se musí lišit od té minimální alespoň jednou hranou. Každá odlišnost zvyšuje váhu ne-minimální kostry. Rozšířením této úvahy je možné i odvodit celý postup.