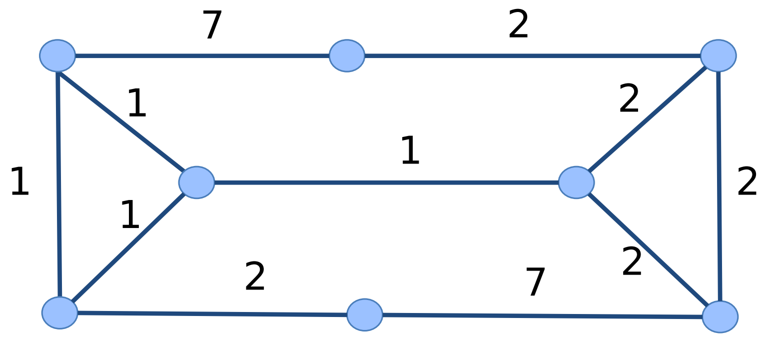
2. CVIČENÍ Z ÚVODU DO APROXIMACÍ

obchodní cestující přicestoval i k nám

Příklad první Uvažte ohodnocený graf zadaný níže:



a) Jaká je nejkratší Hamiltonovská kružnice v tomto grafu?
b) Jaké je optimální řešení TSP na tomto grafu?
c) Jaké řešení nalezne nějaký běh Christofidesova algoritmu?

Příklad druhý Nalezněte nekonečnou třídu grafů, která dokazuje, že algoritmus pro metrické TSP používající obcházení minimální kostry (a zkratkování) není lepší než 2-aproximační.

Přesněji řečeno, pro nekonečně mnoho n zkonstruujte graf G_n s n vrcholy, aby platilo, že

$$\frac{\mathrm{ALG}(G_n)}{\mathrm{OPT}(G_n)} \to 2$$

pro $n \to \infty$, kde $\mathrm{ALG}(G_n)$ je cena řešení algoritmu a $\mathrm{OPT}(G_n)$ cena optimálního řešení.

Příklad třetí Mějme souvislý graf G s délkami hran. Problém obchodního cestujícího je těžký, ale můžeme hledat nejkratší obecnější strukturu – podgraf $P \subseteq G$ minimální celkové délky takový, že P obsahuje všechny vrcholy a všechny stupně v P jsou rovny 2. Tomuto problému se říká Minimální pokrytí kružnicemi.

Hint: Znám alespoň dva způsoby, jak na to jít:

- *Od Kbelnice:* Pokud jste chodili na Optimalizační metody a přátelíte se i s takovými pojmy jako totální unimodularita, tak toto je přímá cesta. (Pozor, ne každý celočíselný program pro tento problém má totálně unimodulární matici.)
- *Od Kopidlna:* Pokud lineární programovaní ještě neovládáte, tak můžete zkusit cestu od Kopidlna, kde to vůbec potřeba není. Připomenu jenom, že nalézt perfektní párování minimální váhy lze v polynomiálním čase.

Příklad čtvrtý Je TSP řešitelné v polynomiálním čase? Nabízí se následující algoritmus využívající dynamické programování:

1. Vytvoř tabulku $d[i, x, y]$, kde význam políčka bude „délka cesty mezi x a y v i krocích“.
2. Nastav $d[0, x, x] = 0$ a $d[0, x, y] = \infty$.
3. Nyní pro každou délku $i \in \{1, 2, 3, \ldots, n\}$:
4. a pro každou dvojici vrcholů a, b:
5. Projdi sousedy a nastav $d[i, a, b] = \min_{x \text{ soused } a}(d(a, x) + d[i-1, x, b])$.
6. Nastav $d[i, b, a] = d[i, a, b]$.
7. Až výpočet skončí, vrať jako délku TSP nejmenší hodnotu $d[n, v, v]$ přes všechny v.

Zanalyzujte tento algoritmus.

PŘÍKLAD PÁTÝ Dále se nabízí heuristika „Hledání nejbližšího souseda“:

1. Začneme nejbližší dvojicí vrcholů a, b, tedy takovou dvojicí, že $d(a, b)$ je nejmenší, a počáteční kružnice přejde hranu mezi a, b tam a zpět.
2. Poté budeme opakovat následující, dokud do kružnice nepřidáme všechny vrcholy:
3. Nechť S je množina vrcholů v aktuální kružnici.
4. Najdeme dvojici vrcholů u, v takovou, že $u \in S$, $v \notin S$ a $d(u, v)$ je nejmenší.
5. Přidáme v do kružnice hned za u.

Zanalyzujte i tento algoritmus.

Hint: Vzpomeňte si na Jarníka.