Týden 11

Přednáška

Navázali jsme na diskusi z minula: třída PTIME a obecné metody návrhu (polynomiálních) algoritmů.

Speciálně „rozděl panuj“; ilustrováno na mergesortu $O(n \log n)$ a násobení matic: klasicky v $O(n^3)$ (kde n je rozměr matic), zmíněna složitost Strassenova algoritmu v $O(n^{\log_2 7})$.

Dynamické programování: jako příklad nejdelší společná podposloupnost … (zmíněn problém příslušnosti k jazyku generovanému bezkontextovou gramatikou).

Optimalizační problémy

Hltavý (greedy) přístup, příklad minimální kostra v grafu (kde tento přístup funguje)

Problém obchodního cestujícího … zde hltavý přístup nefunguje … navíc není znám polynomiální algoritmus

Nedeterministické algoritmy (Turingovy stroje), třída NPTIME …

Přednáška - druhá část

Důkazy korektnosti algoritmů, speciálně fungování hltavého přístupu v případě hledání minimální kostry v grafu … Naopak nefungování hltavého přístupu v případě problému obchodního cestujícího …

Cvičení

Příklad 11.1

Třída PTIME

Připomeňte si definici třídy PTIME. Uveďte precizně příklady alespoň dvou problémů z PTIME a prokažte, že jsou v PTIME.

Poznámka. Všechny problémy ve studijním textu (nejen ty, které jsou v PTIME) je třeba si důkladně promyslet – pak nemůže být pro nikoho problémem u zkoušky nějaký požadovaný problém přesně definovat a uvést příklady instancí s pozitivní odpovědí a instancí s negativní odpovědí.

Příklad 11.2

Nedeterministické algoritmy a jejich složitost; třída NPTIME

Připomeňte si pojem nedeterministického algoritmu (Turingova stroje) a definici toho, co to znamená, že nějaký nedeterministický Turingův stroj M rozhoduje daný problém P.

Rozšiřte pojem (časové) složitosti na nedeterministické stroje a definujte třídu NPTIME.
Navrhněte (nedeterministické) algoritmy prokazující, že problém SAT (splnitelnost booleovských formulí) a IS (Independent Set, rozhodovací verze problému nezávislé množiny v grafu) jsou v NPTIME.

Příklad 11.3

Polynomiální převeditelnost

Definujte pojem *polynomiální převeditelnosti* jako speciální případ dříve uvedené (algoritmické) převeditelnosti mezi problémy. (Příslušný převádějící algoritmus musí mít polynomiální časovou složitost, tedy časovou složitost omezenou polynomem.)
Vysvětlete nejdříve přesně, co máme udělat, chceme-li prokázat polynomiální převeditelnost problému HC (hamiltonovský cyklus v orientovaném grafu) na problém HK (hamiltonovská kružnice v neorientovaném grafu).
Pak to udělejte.