Složitost – cvičení

13.1 Rozhodněte, které z následujících vztahů platí. Odpovědi zdůvodněte.

a) $2n \in O(n)$

b) $n^2 \in O(n)$

c) $n \log_2 n \in O(n^2)$

d) $n \log_2 n \in O(n)$

e) $3^n \in 2^{O(n)}$

f) $3n^2 + 4n + 17 \in O(n^2 - n + 1)$

g) $(2n)! \in O(n!^2)$

Řešení:

a) **Platí.** Zvolme $n_0 = 1 \wedge c = 2$. Pak určitě platí, že $2n = f(n) \leq c \cdot g(n) = 2n$ pro všechna $n \geq 1$.

b) **Neplatí.** Sporem předpokládejme, že existují konstanty n_0, c takové, že $n^2 \leq c \cdot n$ pro $n \geq n_0$. Nerovnici podělíme nenulovým kladným číslem *n*: $n \leq c$. Ať volíme konstanty c, n_0, jak chceme, vždy budou existovat množina přirozených čísel $M = \{n_k, n_k + 1, \dots\}$, kde $n_k = \lceil c \rceil$, taková, že pro všechna $x \in M$ neplatí vztah $x \leq c$.

c) **Platí.** Zvolme $n_0 = 1 \wedge c = 1$. Pak určitě platí, že $n \log_2 n = f(n) \leq c \cdot g(n) = n^2$ pro všechna $n \geq 1$.

d) **Neplatí.** Sporem předpokládejme, že existují konstanty n_0, c takové, že $n \log_2 n \leq c \cdot n$ pro $n \geq n_0$. Nerovnici podělíme nenulovým kladným číslem *n*: $\log_2 n \leq c$. Ať volíme konstanty c, n_0, jak chceme, vždy budou existovat množina přirozených čísel $M = \{n_k, n_k + 1, \dots\}$, kde $n_k = 2^{c+1}$, taková, že pro všechna $x \in M$ neplatí vztah $\log_2 x \leq c$.

e) **Platí.** Z definice platí, že $f(n) \in 2^{O(n)} \Leftrightarrow \exists c, n_0 \in \boldsymbol{N}: f(n) \leq 2^{c \cdot n}$ pro všechna $n \geq n_0$. My hledáme tyto konstanty tak, aby
$(*)\ 3^n \leq 2^{c \cdot n} = (2^n)^c$.
Zvolme $c = 2, n_0 = 1$. Pak $(2^n)^c = (2^n)^2 = 2^{2n} = 4^n$. Jistě tedy pro libovolné $n \geq 1$ platí $3^n \leq 2^{c \cdot n} = 4^n$

f) **Platí.** Zvolme $c = 4$. K optimální volbě n_0 dospějeme řešením nerovnice
$3n^2 + 4n + 17 \leq 4 \cdot (n^2 - n + 1)$
$(*)\ 0 \leq n^2 - 5n - 16$
Buď spočítáním diskriminantu nebo odhadem usoudíme, že pro $n \geq 8$ vztah $(*)$ platí. Hledanou konstantou je $n_0 = 8$.

g) **Neplatí.** Sporem předpokládejme, že
$(*)$ existují konstanty n_0, c takové, že $(2n)! \leq c \cdot (n!)^2$ pro všechna $n \geq n_0$.
Zvolme $m = \max\{1, c, n_0\}$ a dosaďme do vztahu $(*)$:
$(2m)! \leq c \cdot (m!)^2$
$2m \cdot (2m - 1) \cdot \ldots \cdot (m + 1) \cdot m! \leq c \cdot m! \cdot m!$ (podělíme nerovnici kladným výrazem $m!$)
$(**)\ 2m \cdot (2m - 1) \cdot \ldots \cdot (m + 1) \leq c \cdot m \cdot (m - 1) \cdot \ldots \cdot 2 \cdot 1$
Protože $c \leq m$, vztah $(**)$ nemůže nikdy platit. Spor s předpokladem $(*)$.

13.3 Dokažte, že třída P je uzavřená na operace sjednocení, komplement a zřetězení. Rozhodněte, na které z těchto operací je uzavřena třída NP. Odpověď zdůvodněte.

Řešení:

Nejprve připomeňme: P je třída všech jazyků L, pro které existuje deterministický TM s polynomiální časovou složitostí, který rozhoduje L. Zejména tedy platí, že každý jazyk $L \in P$ je rekurzivní. Z toho však vyplývá, že třída P je uzavřena na všechny tři operace.

Třída NP obsahuje všechny jazyky L, pro které existuje nedeterministický TM s polynomiální časovou složitostí, který rozhoduje L. V případě operací sjednocení a zřetězení použijeme postup, který byl aplikován v kapitole o uzávěrových vlastnostech tříd rekurzivních a rekurzivně spočetných jazyků. Buďte M_A, M_B nedeterministické TM, které rozhodují jazyky A, B v čase $O(n^k)$, resp. $O(n^l)$, kde k, l jsou vhodná přirozená čísla.

Nedeterministický Turingův stroj M_S pro jazyk $A \cup B$ sestrojíme tak, že se na začátku nedeterministicky rozhodne, zda bude simulovat práci TM M_A či M_B. V případě, že simulace skončí v akceptujícím stavu, tak M_S akceptuje. V opačném případě zamítá. Stroj M_S na začátku učiní jeden krok (výběr stroje k simulace), následně v čase $O(n^k)$ či $O(n^l)$ simuluje výpočet vybraného stroje, a poté na závěr učiní poslední krok, v němž se přepne buď do akceptujícího nebo do zamítajícího stavu. V každém případě bude opět pracovat v polynomiálním čase.

Nedeterministický Turingův stroj M_Z pro jazyk $A.B$ sestrojíme tak, že se na začátku nedeterministicky rozdělí vstup w na dvě části $w_1.w_2$. Následně spustí simulaci TM M_A na slově w_1 v čase $O(n^k)$. Akceptuje-li M_A slovo w_1, spustí M_Z výpočet M_B na slově w_2. V případě, že simulace skončí v akceptujícím stavu, tak M_Z akceptuje, v každém jiném případě, kdy M_A zamítá w_1 nebo M_B zamítá slovo w_2, zamítá i stroj M_Z. Stroj M_Z na začátku učiní jeden krok (nedeterministické rozdělení slova w), následně v čase $O(n^k) + O(n^l)$ simuluje výpočet obou strojů na slovech w_1, w_2. V každém případě bude opět pracovat v polynomiálním čase.

O uzavřenosti třídy NP na doplněk nelze rozhodnout. Řešením by mohla být záměna akceptujícího a zamítajícího stavu, která fungovala v případě důkazu uzavřenosti třídy rekurzivních jazyků na doplněk. Výpočtů na nějakém slově w však může být v případě nedeterministického TM *M* mnoho. Platí:

1. $w \in L(M) \Leftrightarrow$ existuje akceptující výpočet *M* na *w*;
2. *w* nepatří do $L(M) \Leftrightarrow$ všechny možné výpočty *M* na *w* skončí v zamítajícím stavu.

Výměnou akceptujícího a zamítajícího stavu nedokážeme vyřešit ty situace, kdy pro dané slovo w bude řada zamítajících i akceptujících výpočtů.

13.4 Třída coNP je definována jako $\mathrm{coNP} = \{\mathrm{co} - L \mid L \in \mathrm{NP}\}$. Rozhodněte, které z následujících tvrzení platí. Odpovědi zdůvodněte.

a) $\mathrm{coNP} = \mathrm{co} - \mathrm{NP}$

b) $L_1, L_2 \in \mathrm{coNP} \Rightarrow L_1 \cap L_2 \in \mathrm{coNP}$

c) $L_1 \in \mathrm{NP}, L_2 \subset L_1, L_2 \in \mathrm{coNP} \Rightarrow L_1 \setminus L_2 \in \mathrm{NP}$

Řešení:

a) **Neplatí.** Protipříkladem může být jakýkoliv jazyk $L \in \mathrm{P}$. Díky uzavřenosti třídy P na operaci doplněk platí též $\mathrm{co} - L \in \mathrm{P} \subseteq \mathrm{NP}$. Jazyk $\mathrm{co} - L \in \mathrm{coNP}$, avšak $\mathrm{co} - L$ nepatří do $\mathrm{co} - \mathrm{NP}$.

b) **Platí.** Protože $L_1, L_2 \in \mathrm{coNP}$, znamená to dle definice, že $\mathrm{co} - L_1, \mathrm{co} - L_2 \in \mathrm{NP}$. V příkladu 13.3 jsme ukázali, že třída NP je uzavřena na sjednocení. Platí tedy
$\mathrm{co} - L_1 \cup \mathrm{co} - L_2 = \mathrm{co} - (L_1 \cap L_2) \in \mathrm{NP}$.
Proto $L_1 \cap L_2 \in \mathrm{coNP}$.

c) **Platí.** Protože $L_2 \in \mathrm{coNP}$, je dle definice $\mathrm{co} - L_2 \in \mathrm{NP}$. Platí také vztah
$L_1 \setminus L_2 = L_1 \cap \mathrm{co} - L_2$.
Navíc, třída NP je uzavřena na průnik (důkaz podobný jako v případě uzavřenosti rekurzivních a rekurzivně spočetných jazyků na průnik). Protože $L_1, \mathrm{co} - L_2 \in \mathrm{NP}$, je i
$L_1 \cap \mathrm{co} - L_2 = L_1 \setminus L_2 \in \mathrm{NP}$.

13.6 Dokažte, že následující problémy jsou NP-úplné.

a) Problém Hamiltonovské cesty v grafu:
HAMPATH $= \{\langle G, s, t\rangle \mid G$ je orientovaný graf obsahující Hamiltonovskou cestu z *s* do $t\}$

b) Problém *k*-kliky (*k*-klika je úplný podgraf s *k* vrcholy):
CLIQUE $= \{\langle G, k\rangle \mid G$ je neorientovaný graf s *k*-klikou$\}$

c) Problém podgrafového izomorfismu (Subgraph Isomorphism, SGI):
SGI $= \{\langle H, G\rangle \mid H = (V, E), G = (U, F)$ jsou neorientované grafy takové, že existuje injektivní zobrazení $f: V \to U$ splňující $(u, u') \in E \Rightarrow \left(f(u), f(u')\right) \in F\}$

Řešení:

a) HAMPATH – řešení viz Sipser, kap. 7, důkaz Theoremu 7.35 začínající na str. 262

b) CLIQUE

Důkaz $\mathrm{CLIQUE} \in \mathrm{NP}$: Setrojíme nedeterministický TM *M* pracující pro vstup $\langle G, k\rangle$ takto:

1. Nejprve zjistí, zda vstup je tvaru $\langle G, k\rangle$, kde G je neorientovaný graf a *k* je přirozené číslo. Pokud ne, *M* zamítne.
2. Nechť $G = (V, E)$, kde *V* je množina vrcholů, *E* množina hran. Nedeterministicky zvol množinu vrcholů $U \subseteq V, |U| = k$.
3. Ověř, zda *U* je úplný podgraf grafu *G*.
   a. Pokud ano, akceptuj.
   b. V opačném případě zamítni.

Části 1, 2 lze provést polynomiálním počtu kroků. Je-li $n = |\langle G, k\rangle|$ délka vstupu TM *M*, pak ověření korektnosti vstupu zabere $O(n)$ času. Nedeterministická volba *k* vrcholů také nezabere víc než $O(n)$ kroků. Ověření, zda *k* vrcholů je spojeno hranami, každý s každým, znamená ověřit $(k-1) \cdot \ldots \cdot 2 \cdot 1$

hran. Jde o konstantní počet kroků nezávisející na délce vstupu. Všechny části, kterých je polynomiálně mnoho, lze tedy provést v polynomiálním čase.

Důkaz $\mathrm{3SAT} \leq_{\mathrm{p}} \mathrm{CLIQUE}$: Buď $\Phi = (a_1 \vee b_1 \vee c_1) \wedge (a_2 \vee b_2 \vee c_2) \wedge \dots \wedge (a_k \vee b_k \vee c_k)$ Booleovská formule ve 3cnf. Redukce *f* vyprodukuje kód $\langle G, k \rangle$, kde *G* je neorientovaný graf. Vrcholy budou uspořádány do *k* trojic, jejich názvy budou označeny literály příslušných klauzulí. Vrcholy jedné trojice nebudou spojené hranami. Hrany povedeme napříč trojicemi (klauzulemi) dle tohoto pravidla. Libovolný vrchol x trojice t_i spojíme hranou s vrcholem y trojice t_j $(i \neq j)$ pouze tehdy, je-li $y \neq \neg x$.

Příklad takového grafu, viz Sipser, kap. 7, obr. 7.7.

Chceme ukázat, že 3cnf formule Φ je splnitelná právě tehdy, když $f(\Phi) = \langle G, k \rangle$ je neorientovaný graf, v němž existuje *k*-klika.

1. Předpokládejme, že $\Phi = (a_1 \vee b_1 \vee c_1) \wedge (a_2 \vee b_2 \vee c_2) \wedge \dots \wedge (a_k \vee b_k \vee c_k)$ je splnitelná 3cnf formule. To znamená, že v každé klauzuli je alespoň jeden z literálů pravdivý. Např. jde o množinu $S = \{a_1, b_2, c_3, \dots, b_k\}$. V orientovaném grafu *G* jistě musí být hrana mezi každou dvojicí vybranou z *S*. Pokud by např. pro dvojici $\{a_1, b_2\}$ nebyla, znamenalo by to případ, že $b_2 = \neg a_1$. To ovšem není možné, nemůže být pravdivý literál a_1 i jeho negace $\neg a_1$. Množina vrcholů *S* tedy tvoří *k*-kliku v grafu *G*.
2. Předpokládejme, že graf *G* disponuje *k*-klikou. Dle pravidel konstrukce žádné dva vrcholy kliky nejsou ve stejné trojici. Ohodnocení proměnných provedeme tak, že každý vrchol kliky (tj. literál, ať už v pozitivní či negativní formě) bude pravdivý. Protože hranou nejsou spojeny dva vrcholy s návěštím $x, \neg x$, dosáhneme ohodnocení, které pro každou klauzuli bude znamenat alespoň jeden pravdivý literál, tj. celá formule Φ sestavená z jednotlivých trojic grafu *G* tvořících klauzule bude splnitelná.

c) SGI

Důkaz $\mathrm{SGI} \in \mathrm{NP}$: Setrojíme nedeterministický TM *M* pracující pro vstup $\langle H, G \rangle$ takto:

1. Nejprve zjistí, zda vstup $\langle H, G \rangle$ kóduje dva neorientované grafy $H = (V, E)$ a $G = (U, F)$. Pokud ne, *M* zamítne.
2. Nedeterministicky zvolí zobrazení $f: V \to U$ takové, aby bylo injektivní.
3. Pro každou hranu $\{v_1, v_2\} \in E$ ověří, zda $\{f(v_1), f(v_2)\} \in F$. Pokud ne, zamítá.
4. Stroj *M* akceptuje, pokud testy v předchozím bodu byly pro všechny hrany úspěšné.

Všechny čtyři části lze provést v polynomiálním počtu kroků.

Důkaz $\mathrm{CLIQUE} \leq_{\mathrm{p}} \mathrm{SGI}$: Mějme kód $\langle G, k \rangle$. Sestrojíme funkci f, která nejprve ověří, zda

- $\langle G, k \rangle$ skutečně kóduje dvojici neorientovaný graf $G = (V, E)$ a přirozené číslo *k*;
- $|V| \geq k$

Pokud jedna z podmínek neplatí, nastavíme pro kód *x* funkci *f* takto:

$f(\langle x \rangle) = \langle G_1, G_2 \rangle$,

kde oba grafy budou mít pouze jeden vrchol, přičemž G_1 bude mít u svého vrcholu smyčku, množina hran G_2 bude prázdná. Jistě kód $\langle G_1, G_2 \rangle$ nepatří do SGI. Platí-li obě podmínky, nastavíme funkci *f* takto:

$f(\langle G, k \rangle) = \langle G, H_k \rangle$,

kde H_k je úplný neorientovaný graf s *k* vrcholy. Jistě platí, že $\langle G, k \rangle \in \mathrm{CLIQUE} \Leftrightarrow \langle G, H_k \rangle \in \mathrm{SGI}$ (ověření obou stran implikace je tentokrát zřejmé). Redukce *f* je sestrojena tak, že reaguje na všechny možné vstupy, i když není zadána dvojice (graf, číslo). Ověření obou uvedených podmínek jistě probíhá v polynomiálním čase. Sestrojení grafů G_1, G_2 je také nenáročné na čas. Vytvoření grafu H_k je také možné zvládnout polynomiálně. Následné ověření podmínky $\langle G, H_k \rangle \in \mathrm{SGI}$ sice probíhá nedeterministicky, ale v polynomiálním čase, protože jsme ověřili platnost $\mathrm{SGI} \in \mathrm{NP}$.

13.7 Určete vztahy inkluze/rovnost mezi následujícími dvojicemi složitostních tříd. Svoje tvrzení zdůvodněte.

a) $\mathrm{TIME}(n^2)$ a $\mathrm{TIME}(n^3)$

b) $\mathrm{SPACE}(2n^2)$ a $\mathrm{SPACE}(100n^2)$

c) $\mathrm{SPACE}(n^2)$ a $\mathrm{TIME}(n^2)$

d) $\mathrm{NSPACE}(n^2)$ a $\mathrm{SPACE}(n^5)$

e) P a $\mathrm{TIME}(2^n)$

__**Řešení:**__

a) $\mathrm{TIME}(n^2)$, resp. $\mathrm{TIME}(n^3)$ jsou třídy jazyků rozhodovaných nějakým deterministickým jedno- či vícepáskovým TM s časovou složitostí $O(n^2)$, resp. $O(n^3)$. Jistě platí $n^2 \in O(n^3)$. Proto jazyk $L \in \mathrm{TIME}(n^2)$ patří i do třídy $\mathrm{TIME}(n^3)$, tedy $\mathrm{TIME}(n^2) \subseteq \mathrm{TIME}(n^3)$.

b) V obou případech platí, že $O(k \cdot n^2) = O(n^2)$. Proto se třídy rovnají, tj. platí $\mathrm{SPACE}(2n^2) = \mathrm{SPACE}(100n^2)$.

c) Z přednášky (poslední přednášková sada, slajd 20) víme, že pro každou funkci $f: \boldsymbol{N_0} \to \boldsymbol{R}^+$ platí $\mathrm{TIME}\big(f(n)\big) \subseteq \mathrm{SPACE}(f(n))$. Jistě tedy platí vztah $\mathrm{TIME}(n^2) \subseteq \mathrm{SPACE}(n^2)$.

d) Podle Savitchovy věty platí vztah $\mathrm{NSPACE}(n^2) \subseteq \mathrm{SPACE}(n^4)$. Dále je zřejmé, že $n^4 \in O(n^5)$, tj. $\mathrm{SPACE}(n^4) \subseteq \mathrm{SPACE}(n^5)$. Z obou vztahů tedy vyplývá $\mathrm{NSPACE}(n^2) \subseteq \mathrm{SPACE}(n^5)$.

e) P je třída jazyků rozhodovaných nějakým deterministickým jedno- či vícepáskovým TM s polynomiální časovou složitostí $O(n^k)$. Zcela jistě $n^k \in O(2^n)$ pro každé přirozené *k.* Tedy $P \subseteq TIME(2^n)$.