Obsah

Kapitola 1

Vyčíslitelné funkce

Příklad 1 Dokažte, že tyto dvě množiny nejsou spočetné:

1. Množina všech podmnožin množiny $\mathbb{N}$
2. Množina všech částečných funkcí z $\mathbb{N}$ do $\mathbb{N}$, jejichž obor hodnot je konečná množina.

***Příklad 2** Uvažujte následující funkci:

$$f(x) = \begin{cases} 1 & \text{jestliže existuje posloupnost } \textit{alespoň } x \\ & \text{sousedních 5 v desetinném rozvoji čísla } \pi; \\ 0 & \text{jinak} \end{cases}$$

Například $f(0) = f(1) = 1$, protože $\pi = 3.14159265358979\ldots$. Je funkce f vyčíslitelná? Změní se odpověď, jestliže v definici funkce f nahradíme *alespoň x* výrazem *přesně x*?

Příklad 3 Nechť f je částečná funkce, jejíž definiční obor je konečná množina. Je f vyčíslitelná? Nechť nyní f je částečná funkce, jejíž obor hodnot je konečná množina, ale definiční obor nemusí být konečný. Je f vyčíslitelná?

Příklad 4 Předpokládejme programovací jazyk $\mathcal{L}$, jehož programy vždy *končí*. Zdůvodněte, že existují vyčíslitelné funkce, které nejsou vyčíslitelné v tomto programovacím jazyku.

Příklad 5 Napište **while**-program, který počítá "$x_i := x_j - 1$" a používá z přiřazovacích příkazů jen $x := 0$ a $x_i := x_j + 1$.

Příklad 6 Napište **while**-programy pro následující makro-příkazy:

1. $Z := X + Y$
2. $Z := X * Y$
3. $Z := X \mathbf{div} Y$
4. $Z := X \mathbf{mod} Y$
5. $Z := 2^X$

Příklad 7 Napište makro definice pro konstrukce **if-then-else** a **repeat-until**.

****Příklad 8** **While**-programy, které neobsahují příkaz cyklu **while-do** se nazývají "přímé" programy.

1. Ukažte, že ke každému **while**-programu s přesně jednou proměnnou, existuje přímý program, který počítá tutéž funkci.

2. Ukažte, že existuje funkce, kterou lze počítat **while**-programem se dvěma proměnnými, ale není vyčíslitelná žádným přímým programem.

****Příklad 9** Dokažte, že náhradou příkazu **while** $x_i \neq x_j$ **do** δ příkazem **while** $x_i \neq 0$ **do** δ získáme ekvivalentní třídu **while**-programů.

***Příklad 10** Ukažte, že neexistuje **while**-program s *jednou* proměnnou, který počítá funkci $f(x) = 2 * x$. Nepoužívejte makro-příkazy!

Kapitola 2

Numerace vyčíslitelných funkcí

Příklad 11 Nechť $\Phi : \mathbb{N}^2 \to \mathbb{N}$ je univerzální funkce pro množinu všech unárních vyčíslitelných funkcí. Kolik indexů má Φ? Vaši odpověď zdůvodněte!

Příklad 12 Nechť P_e je program, který počítá universální funkci $\Phi : \mathbb{N}^2 \to \mathbb{N}$. Ukažte, že výsledek výpočtu programu P_e pro vstupní vektor (e, a) je stejný jako výsledek výpočtu P_e pro vstupní vektor $(a, 0)$ pro libovolné $a \in \mathbb{N}$.

***Příklad 13** Nechť $f : \mathbb{N} \to \mathbb{N}$ je vyčíslitelná bijekce. Uvažujme numeraci unárních vyčíslitelných funkcí

$$\psi_0, \psi_1, \dots, \psi_n, \dots$$

kde $\psi_n = \varphi_{f(n)}$. Dokažte, že pro tuto numeraci existuje vyčíslitelná univerzální funkce $\Psi : \mathbb{N}^2 \to \mathbb{N}$. Ukažte, že existuje a takové, že funkce Ψ je počítána programem $P_{f(a)}$.

Budou požadovaná tvrzení platit i v případě, že f je totálně vyčíslitelná surjekce, ale není prostá a v případě, že f je totálně vyčíslitelná a prostá, ale není surjektivní?

Příklad 14 Definujme obecněji pojem *univerzální* funkce následovně. Nechť $\mathcal{F}$ je třída $j-$árních funkcí, ne nutně vyčíslitelných. Univerzální funkce $F : \mathbb{N}^{j+1} \to \mathbb{N}$ pro $\mathcal{F}$ je funkce splňující následující dvě podmínky:

1. Pro každé pevné $e \in \mathbb{N}$ patří $j-$ární funkce $F(e, x_1, \dots, x_j)$ do $\mathcal{F}$.
2. Ke každé funkci $f(x_1, \dots, x_j) \in \mathcal{F}$ existuje $e \in \mathbb{N}$ tak, že pro všechna $x_1, \dots, x_n \in \mathbb{N}$:

$$F(e, x_1, \dots, x_j) = f(x_1, \dots, x_j)$$

Pro jednoduchost předpokládejme $j = 1$.

1. Definujte univerzální funkci pro třídu funkcí $\mathcal{F} = \{1, x, x^2, x^3, \dots\}$.
2. Definujte univerzální funkci pro třídu funkcí $\mathcal{F} = \{x, x^3, x^5, \dots\}$.
3. Ukažte, že každá konečná třída funkcí $\mathcal{F}$ má univerzální funkci F.

***Příklad 15** Použijeme definici univerzální funkce z cvičení 14.

1. Nechť $\mathcal{F}$ je třída totálních unárních funkcí nad $\mathbb{N}$, která má univerzální funkci $F : \mathbb{N}^2 \to \mathbb{N}$. Dokažte, že pak existuje totální unární funkce, která nepatří do $\mathcal{F}$.
2. Dokažte, že třída *všech* totálně vyčíslitelných unárních funkcí nad $\mathbb{N}$, nemá *vyčíslitelnou* univerzální funkci.
3. Definujte nekonečnou třídu totálně vyčíslitelných funkcí, která má vyčíslitelnou univerzální funkci.

Příklad 16 Ukažte, že existuje totálně vyčíslitelná funkce $g : \mathbb{N}^2 \to \mathbb{N}$ taková, že

$$\varphi_{g(i,j)}(x) = \varphi_i(x) + \varphi_j(x)$$

přičemž součet je definován právě pro ta x, pro která jsou definovány obě funkce φ_i, φ_j.

***Příklad 17** Dokažte, že existují totálně vyčíslitelné funkce f, g takové, že

1. $\varphi_{f(x)}(y) = \begin{cases} \varphi_x^{(2)}(y, \varphi_x^{(2)}(y,y)) & \text{je-li } \varphi_x^{(2)}(y,y) \text{ definováno;} \\ \perp & \text{jinak} \end{cases}$

2. $\varphi_{g(x)}^{(2)}(y,z) = \begin{cases} \varphi_x^{(2)}(y,y) & \text{jestliže } y = z; \\ \varphi_x^{(2)}(z,z) & \text{jestliže } y = z+1; \\ \varphi_x^{(2)}(y,z) & \text{jestliže } y = z+2; \\ \varphi_x^{(2)}(z,y) & \text{jinak} \end{cases}$

V obou případech dokažte existenci funkce jednak konstrukcí odpovídajícího programu, jednak s využitím věty o parametrizaci (bez konstrukce programu).

****Příklad 18** Nechť $\psi_0, \psi_1, \ldots, \psi_n, \ldots$ je libovolná numerace vyčíslitelných funkcí, která nemusí být efektivní a která nemusí obsahovat všechny vyčíslitelné funkce, t.j. jestliže položíme $\psi_n = \varphi_{f(n)}$, pak f je totální funkce, která nemusí být ani vyčíslitelná ani surjektivní. Takovouto numeraci nazveme *kvazi-efektivní* jestliže existuje totálně vyčíslitelná funkce $g : \mathbb{N} \to \mathbb{N}$ taková, že:

$$\varphi_{f(0)} = \varphi_{g(0)}, \varphi_{f(1)} = \varphi_{g(1)}, \ldots, \varphi_{f(n)} = \varphi_{g(n)}, \ldots$$

Poznamenejme, že to nemusí nutně znamenat, že $f = g$.

Dokažte, že numerace ψ_n je kvazi-efektivní právě když její univerzální funkce $\Psi(n,x) = \psi_n(x)$ je vyčíslitelná.

Příklad 19 Nechť $\Phi : \mathbb{N}^2 \to \mathbb{N}$ je univerzální funkce. Ukažte, že funkce $\psi(x) = \Phi(x,x)$ nemůže být rozšířena na totální vyčíslitelnou funkci.
Řekneme, že f je *rozšířením* g, píšeme $g \leq f$, jestliže kdykoliv je $g(x)$ definováno, pak je definováno i $f(x)$ a $f(x) = g(x)$.

Příklad 20 Zjistěte, která z následujících funkcí je vyčíslitelná a výsledek zdůvodněte:

1. $\psi_1(x,y) = \begin{cases} y & \text{jestliže } \varphi_x(x) \text{ je definováno;} \\ \perp & \text{jinak} \end{cases}$

2. $\psi_2(x,y) = \begin{cases} y & \text{jestliže } \varphi_x(x) \text{ je definováno;} \\ 0 & \text{jinak} \end{cases}$

3. Nechť $\mu, \sigma : \mathbb{N} \to \mathbb{N}$ jsou vyčíslitelné funkce takové, že $\sigma \leq \mu$.
   $\psi_3(x,y) = \begin{cases} \mu(y) & \text{jestliže } \varphi_x(x) \text{ je definováno;} \\ \sigma(y) & \text{jinak} \end{cases}$

Kapitola 3

Vyčíslitelné vlastnosti množin

***Příklad 21** Nechť A je nekonečná r.e. množina, pro jejíž numerující funkci f platí:

$$\text{pro všechna } n \geq 0 : f(2n+3) > f(2n+1) \text{ a } f(2n+2) > f(2n)$$

Dokažte, že A musí být rekurzívní.

***Příklad 22** Charakteristická funkce χ_A množiny $A \subseteq \mathbb{N}$ je definována takto:

$$\chi_A(x) = \begin{cases} 1 & \text{je-li } x \in A \\ 0 & \text{je-li } x \notin A. \end{cases}$$

Dokažte, že A je rekurzívní množina právě když χ_A je totálně vyčíslitelná funkce.

Příklad 23 Použijte techniku "paralelního zpracování" k důkazu toho, že následující množiny jsou rekurzívně spočetné:

1. $\{i \mid \varphi_i \neq \epsilon\}$, kde ϵ je prázná funkce,
2. $\{i \mid \varphi_i$ není prostá$\}$,
3. $\{i \mid \varphi_i$ není konstantní funkce$\}$,
4. $\{n \mid a \in \mathrm{dom}(\varphi_n)\}$, pro pevné $a \in \mathbb{N}$,
5. $\{n \mid a \in \mathrm{range}(\varphi_n)\}$, pro pevné $a \in \mathbb{N}$.

****Příklad 24** Dokažte, že každá nekonečná rekurzívní množina má jak nerekurzívní r.e. podmnožinu, tak i podmnožinu, která není r.e.

****Příklad 25** Dokažte, že každá nekonečná r.e. množina má jak nerekurzívní r.e. podmnožinu, tak i podmnožinu, která není r.e.

***Příklad 26** Uveďte příklad množiny, která není r.e. a která má rekurzívní podmnožinu a nerekurzívní r.e. podmnožinu.

Příklad 27 Dokažte, že řezy a kartézské součiny rekurzívních relací jsou rekurzívní.

Příklad 28 Dokažte, že každá nekonečná r.e. množina má prostou numerující funkci.

Příklad 29 Nechť A je nekonečná r.e. množina, jejíž numerující funkce f splňuje tuto podmínku: Existuje $c \in \mathbb{N}$ tak, že pro všechna $i, j \geq c$ platí

$$i < j \Rightarrow f(i) < f(j).$$

Dokažte, že A je rekurzívní.

Příklad 30 Najděte vyčíslitelnou funkci $\theta : \mathbb{N} \to \mathbb{N}$ takovou, že pro každou totálně vyčíslitelnou funkci $f : \mathbb{N} \to \mathbb{N}$ existuje $n \in \mathbb{N}$ tak, že $\theta(n)$ je definováno a $\theta(n) \neq f(n)$.

****Příklad 31** Nechť θ je funkce zadaná ve cvičení 30 a nechť f je *daná* totálně vyčíslitelná funkce. Ukažte, že množina

$$A = \{n \mid \theta(n) \text{ je definováno a } f(n) \neq \theta(n)\}$$

je rekurzívně spočetná, ale není rekurzívní.

Kapitola 4

Uzávěrové vlastnosti

***Příklad 32** Nechť A je rekurzívní množina a f je totálně vyčíslitelná funkce. Jsou množiny $f(A)$ a $f^{-1}(A)$ rekurzívní? Jsou r.e.? Odpověď zdůvodněte!

Příklad 33 Nechť $A, B \subseteq \mathbb{N}$. Nechť $A \oplus B = \{2a \mid a \in A\} \cup \{2b + 1 \mid b \in B\}$. Dokažte, že

1. A je rekurzívní a B je rekurzívní právě když $A \oplus B$ je rekurzívní,
2. A je rekurzívně spočetná a B je rekurzívně spočetná právě když $A \oplus B$ je rekurzívně spočetná.

Kapitola 5

Riceovy věty

Příklad 34 Pomocí Riceovy věty dokažte, že následující množiny nejsou rekurzívní.

1. $A_2 = \{i \mid \varphi_i = f\}$, kde f je pevná totálně vyčíslitelná funkce
2. $A_3 = \{i \mid \varphi_i = g\}$, kde g je pevná vyčíslitelná funkce
3. $A_4 = \{i \mid a \in \mathrm{dom}(\varphi_i)\}$, kde $a \in \mathbb{N}$ je pevné
4. $A_5 = \{i \mid \mathrm{dom}(\varphi_i) = \emptyset\}$
5. $A_6 = \{i \mid \mathrm{dom}(\varphi_i)$ je konečná množina$\}$
6. $A_7 = \{i \mid a \in \mathrm{range}(\varphi_i)\}$, kde $a \in \mathbb{N}$ je pevné
7. $A_8 = \{i \mid \mathrm{range}(\varphi_i)$ je konečná množina$\}$
8. $A_9 = \{i \mid \mathrm{dom}(\varphi_i) = \mathbb{N}\}$
9. $A_{10} = \{i \mid \varphi_i$ je prostá$\}$
10. $A_{11} = \{i \mid \varphi_i$ je bijekce$\}$

***Příklad 35** Funkce $\xi : \mathbb{N} \to \mathbb{N}$ se nazývá *konečná*, je-li $\mathrm{dom}(\xi)$ konečná množina. Nechť

$$I = \{i \mid \xi \leq \varphi_i\}$$

kde ξ je konečná funkce. Respektuje I funkce? Je I rekurzívní? Je I r.e? Je její doplněk $\overline{I}$ r.e? Odpovědi zdůvodněte.

***Příklad 36** Je množina

$$A = \{i \mid P_i \text{ s proměnnými inicializovanými na 0 cyklí}\}$$

r.e.? Je rekurzívní? Lze použít Riceovu větu?

Příklad 37 Dokažte, že množina $B = \{i \mid \varphi_i(i^2)$ je definováno$\}$ není rekurzívní. Lze použít Riceovu větu? Je tato množina r.e.?

Příklad 38 Uvažujme následující množiny:
(a) $\{i \mid W_i$ je konečná$\}$
(b) $\{i \mid W_i$ je rekurzívní$\}$
(c) $\{i \mid W_i$ je rekurzívní, ale ani W_i ani $\overline{W}_i$ není konečná$\}$

U každé z nich zjistěte, zda množina respektuje funkce, zda je rekurzívní, r.e. nebo není r.e. a zdůvodněte.

Příklad 39 Pomocí Riceovy věty dokažte, že následující množiny nejsou rekurzívně spočetné.

1. $\overline{A_2} = \{i \mid \varphi_i \neq f\}$, kde f je pevná totálně vyčíslitelná funkce
2. $A_3 = \{i \mid \varphi_i = g\}$, kde g je pevná vyčíslitelná funkce
3. $\overline{A_3} = \{i \mid \varphi_i \neq g\}$, kde g je pevná vyčíslitelná funkce
4. $A_4 = \{i \mid a \in \mathrm{dom}(\varphi_i)\}$, kde $a \in \mathbb{N}$ je pevné
5. $A_5 = \{i \mid \mathrm{dom}(\varphi_i) = \emptyset\}$
6. $A_6 = \{i \mid \mathrm{dom}(\varphi_i)$ je konečná množina $\}$
7. $\overline{A_7} = \{i \mid a \notin \mathrm{range}(\varphi_i)\}$, kde $a \in \mathbb{N}$ je pevné
8. $A_8 = \{i \mid \mathrm{range}(\varphi_i)$ je konečná množina$\}$
9. $\overline{A_9} = \{i \mid \mathrm{dom}(\varphi_i) \neq \mathbb{N}\}$
10. $A_{10} = \{i \mid \varphi_i$ je prostá$\}$
11. $\overline{A_{11}} = \{i \mid \varphi_i$ není bijekce$\}$

Kapitola 6

Složitost – Rozcvička

Příklad 40 Které z následujících dvojic čísel jsou nesoudělné? Podrobně napište výpočty, které vedly k Vašemu závěru.

a. 1274 and 10505

b. 7289 and 8029

Příklad 41 Které z následujících dvojic čísel jsou nesoudělné? Podrobně napište výpočty, které vedly k Vašemu závěru.

a. 1276 and 10505

b. 7289 and 8129

Příklad 42 Které z následujících dvojic čísel jsou nesoudělné? Podrobně napište výpočty, které vedly k Vašemu závěru.

a. 1620 and 1989

b. 1322 and 8129

Příklad 43 Je následující formule splnitelná? Pokud ano, napište nějaké splňující přiřazení proměnným!

$$(x \vee y) \wedge (x \vee \neg y) \wedge (\neg x \vee y) \wedge (\neg x \vee \neg y)$$

Příklad 44 Je následující formule splnitelná? Pokud ano, napište nějaké splňující přiřazení proměnným!

$$(x \vee \neg y) \wedge (x \vee y \vee z) \wedge (\neg x \vee \neg y) \wedge (\neg x \vee y) \wedge (x \vee \neg z)$$

Příklad 45 Je následující formule splnitelná? Pokud ano, napište nějaké splňující přiřazení proměnným!

$$(x \vee \neg y) \wedge (x \vee \neg y \vee z) \wedge (\neg x \vee \neg y) \wedge (\neg x \vee y) \wedge (x \vee \neg z)$$

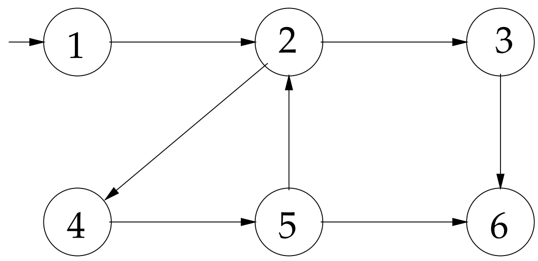
Příklad 46 Je následující formule splnitelná? Pokud ano, napište nějaké splňující přiřazení proměnným!

$$(x \vee y \vee z) \wedge (\neg x \vee y \vee z) \wedge (x \vee \neg y \vee z) \wedge (x \vee y \vee \neg z) \wedge (\neg x \vee \neg y \vee z) \wedge (x \vee \neg y \vee \neg z) \wedge (\neg x \vee \neg y \vee \neg z)$$

Příklad 47 Hra *tic-tac-toe* je variace piškvorek. Využívá hrací pole tři krát tři, na které dva hráči postupně umisťují své symboly, stejně jako u piškvorek. Účelem hry je vytvořit přímku ze svých tří symbolů. Pokud se zaplní hrací pole bez vytvoření sledu tří stejných symbolů, hra končí remízou. Má hráč který začíná výherní strategii? Pokud ano, popište ji, stačí první tah. Pokud ne, popište obrannou strategii druhého hráče, stačí protitah proti každému možnému prvnímu tahu prvního hráče.

Příklad 48 Uvažujme následující hru dvou hráčů na orientovaném grafu $G = (V, H)$. První hráč začíná ve vyznačeném vrcholu $v \in V$ a vybere nějaký vrchol $u \in V$, do kterého vede z v hrana, tedy $(v, u) \in H$. Druhý hráč potom vybere nějaký nenavštívený vrchol, do kterého vede hrana z u, atd. Hráči se střídají. Hra pokračuje dokud existuje tah do ještě nenavštíveného vrcholu. Hráč, který nemá kam táhnout, prohrává.

Má první hráč výherní strategii v následující hře? Má ji druhý hráč? Startovní vrchol je vyznačen šipkou směřující odnikud.



Příklad 49 Je dán semi-Thueův systém $\mathcal{T} = (\{a, b\}, \{ba \to bbb, bbb \to b\})$. Rozhodněte, zda:

a. $abaaba \Rightarrow^*_{\mathcal{T}} abb$

b. $babababa \Rightarrow^*_{\mathcal{T}} b$

Příklad 50 Je dán semi-Thueův systém $\mathcal{T} = (\{a, b, c\}, \{ab \to ba, abc \to acb, ac \to ca, cba \to ab\})$. Rozhodněte, zda:

a. $abcabc \Rightarrow^*_{\mathcal{T}} baab$

b. $acbccc \Rightarrow^*_{\mathcal{T}} ba$

Příklad 51 Je dán semi-Thueův systém $\mathcal{T} = (\{a, b, c\}, \{ab \to ba, ba \to ab, bb \to aa\})$. Rozhodněte, zda:

a. $abbbbbaab \Rightarrow^*_{\mathcal{T}} baabbbbba$

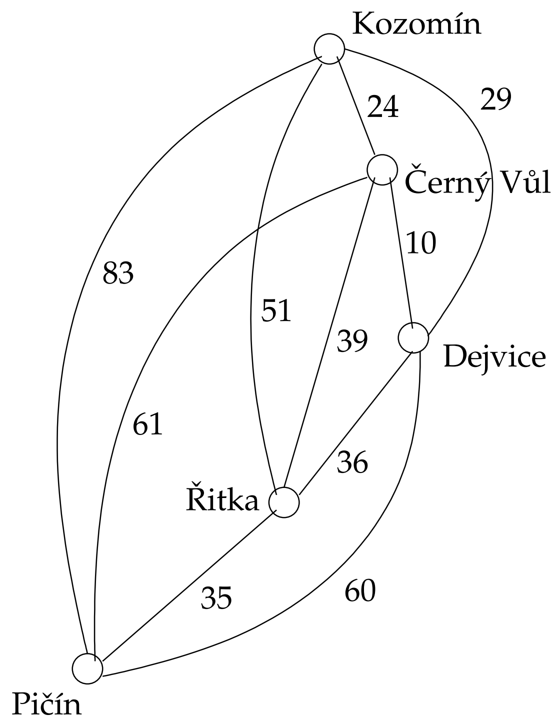
b. $abbabababab \Rightarrow^*_{\mathcal{T}} aaaaaaaaa$

Příklad 52 Je dán Postův systém $S = \{(aba, a), (bbb, aaa), (aab, abab), (bb, babba)\}$. Má tento systém řešení? Zdůvodněte!

Příklad 53 Je dán Postův systém $S = \{(abc, ab), (ca, a), (acc, ba)\}$. Má tento systém řešení? Zdůvodněte!

Příklad 54 Je dán Postův systém $S = \{(ab, a), (bbaaba, a), (b, bbbb), (bb, ab)\}$. Má tento systém řešení? Zdůvodněte!

Příklad 55 Vyřešte TSP problém s následujícím zadáním.



Příklad 56 Zkonstruujte jednopáskový Turingův stroj, který rozhoduje jazyk $L = \{0^k 1^k \mid k \geq 0\}$ v čase $\mathcal{O}(n \log n)$. Přesná definice Vámi navrženého stroje není nutná, stačí dostatečně podrobný popis funkce.

Příklad 57 Zkonstruujte dvoupáskový Turingův stroj, který rozhoduje jazyk $L = \{0^k 1^k \mid k \geq 0\}$ v čase $\mathcal{O}(n)$. Přesná definice Vámi navrženého stroje není nutná, stačí dostatečně podrobný popis funkce.

Příklad 58 Zkonstruujte jednopáskový Turingův stroj, který rozhoduje jazyk $L = \{0^k 1^k \mid k \geq 0\}$ v čase $\mathcal{O}(n^2)$. Stroj má povoleno zapisovat na pásku pouze symboly z abecedy $\Sigma = \{0, 1\}$. Přesná definice Vámi navrženého stroje není nutná, stačí dostatečně podrobný popis funkce.

Příklad 59 Nechť M je deterministický jednopáskový Turingův stroj, který rozhoduje jazyk L v čase $\mathcal{O}(n^{\frac{3}{2}})$. Je možné z toho vyvodit závěr, že existuje konstanta $k \in \mathbb{N}$ taková, že hlava M se při výpočtu nad libovolným slovem z L nikdy nedostane dál než k políček od začátku pásky? Pokud ano, určete konstantu k.

Příklad 60 Nechť M je deterministický jednopáskový Turingův stroj, který rozhoduje konečný jazyk L v čase $\mathcal{O}(n^{\frac{3}{2}})$. Je možné z toho vyvodit závěr, že existuje konstanta $k \in \mathbb{N}$ taková, že hlava M se při výpočtu nad libovolným slovem z L nikdy nedostane dál než k políček od začátku pásky? Pokud ano, určete konstantu k.

Kapitola 7

Redukce a úplnost

Příklad 61 Dokažte, že relace $\leq_m$ (many-to-one redukce) je tranzitivní.

Příklad 62 Dokažte, že relace $\leq_p$ (polynomiální many-to-one redukce) je tranzitivní.

Příklad 63 Rozhodněte, zda relace $\leq_p$ (polynomiální many-to-one redukce) je reflexivní a symetrická.

Příklad 64 Nechť $A, B \subseteq \{a,b\}^*$, $A \leq_m B$ a jazyk B je regulární. Platí, že pak i jazyk A je regulární? Zdůvodněte!

Příklad 65 Nechť množina $A \subseteq \mathbb{N}$ je rekursivně spočetná a $A \leq_m \mathsf{co}\text{–}A$. Pak A je rekursivní. Dokažte.

Příklad 66 Nechť množina $A \subseteq \mathbb{N}$ je rekursivně spočetná a $\mathsf{co}\text{–}A \leq_m A$. Pak A je rekursivní. Dokažte.

Příklad 67 Nechť $A = \{i \in \mathbb{N} \mid \varphi_i(i^4 + 2i^2 + i + 8)\ \textit{je definováno}\}$. Dokažte, že množina A není rekurzivní.

Příklad 68 Nechť $A = \{i \in \mathbb{N} \mid \varphi_i(5i)\ \textit{není definováno}\}$. Dokažte, že množina A není rekurzivně spočetná.

Příklad 69 Nechť $A = \{i \in \mathbb{N} \mid \varphi_i = \epsilon\}$, kde ϵ je prázdná funkce. Dokažte pomocí redukce, že A není rekurzivně spočetná.

Příklad 70 Nechť $A = \{i \in \mathbb{N} \mid \varphi_i = g\}$, kde g je pevně daná vyčíslitelná funkce různá od prázdné funkce. Dokažte pomocí redukce, že A není rekurzivní.

Příklad 71 Nechť L je jazyk, $L \subseteq \Sigma^*$. Jestliže $L \neq \emptyset$ a současně $L \neq \Sigma^*$, pak pro každý jazyk $L_0 \in \mathsf{P}$ platí $L_0 \leq_p L$. Dokažte.

Příklad 72 Za předpokladu $\mathsf{P} \neq \mathsf{NP}$ existují dva jazyky A, B patřící do třídy NP takové, že

- A,B jsou neprázdné a i jejich komplementy jsou neprázdné, a současně
- A není redukovatelné na B.

Dokažte!

Příklad 73 Ukažte, že za předpokladu $\mathsf{P} = \mathsf{NP}$ je každý jazyk $L \in \mathsf{P}$, kromě $L = \emptyset$ a $L = \Sigma^*$, NP-úplný.

Příklad 74 Rozhodněte, zda je problém zastavení Turingova stroje NP–těžký? Svou odpověď dokažte!

****Příklad 75** Problém *SUBSET-SUM* je definován následovně.

$$\textit{SUBSET-SUM} = \{S \subset \mathbb{Z} \mid S \text{ je konečná a existuje } A \subseteq S \text{ taková, že } A \neq \emptyset \wedge \sum_{a \in A} a = 0\}$$

Najděte polynomiální redukci problému *SUBSET-SUM* na problém odvození v semi-Thueově systému.

***Příklad 76** Dokažte, že problém jednoznačnosti bezkontextové gramatiky je nerozhodnutelný. Můžete k tomu využít fakt, že Postův problém přiřazení je nerozhodnutelný.

***Příklad 77** Dokažte, že problém zániku matic je nerozhodnutelný. Můžete k tomu využít fakt, že Postův problém přiřazení je nerozhodnutelný.

***Příklad 78** *Hamiltonovská cesta* v orientovaném grafu je cesta, která obsahuje každý vrchol grafu právě jednou. *Hamiltonovský cyklus* v orientovaném grafu je cyklus, který obsahuje každý vrchol grafu právě jednou. Problémy *HAMPATH* a *HAMCYCLE* jsou definovány následovně:

$$\textit{HAMPATH} = \{< G, s, t >\mid \textit{V orientovaném grafu G existuje hamiltonovská cesta z } s \textit{ do } t\}$$

$$\textit{HAMCYCLE} = \{< G >\mid \textit{V orientovaném grafu G existuje hamiltonovský cyklus}\}$$

Dokažte následující tvrzení:

a. *HAMPATH* $\leq_p$ *HAMCYCLE*

b. *HAMCYCLE* $\leq_p$ *HAMPATH*

Příklad 79 *Hamiltonovský cyklus* v neorientovaném grafu je cyklus, který obsahuje každý vrchol grafu právě jednou. *Hamiltonovský cyklus* v orientovaném grafu je cyklus, který obsahuje každý vrchol grafu právě jednou, přitom cyklus musí respekrovat orientaci hran. Problémy *UHAMCYCLE* a *HAMCYCLE* jsou definovány následovně:

$$\textit{UHAMCYCLE} = \{< G >\mid \textit{V neorientovaném grafu G existuje hamiltonovský cyklus}\}$$

$$\textit{HAMCYCLE} = \{< G >\mid \textit{V orientovaném grafu G existuje hamiltonovský cyklus}\}$$

Dokažte, že problém *UHAMCYCLE* je NP-úplný. Můžete k tomu využít fakt, že problém *HAMCYCLE* je NP-úplný.

Návod: HAMCYCLE $\leq_p$ *UHAMCYCLE*. Redukce přiřadí orientovanému grafu graf neorientovaný. Každému vrcholu v orientovaného grafu budou odpovídat tři vrcholy – v_{in}, v_{mid}, v_{out} – grafu neorientovaného. Mezi v_{in} a v_{mid} a mezi v_{mid} a v_{out} budou neorientované hrany. Pokud v původním grafu existovala orientovaná hrana (u, v), bude v novém grafu neorientovaná hrana (u_{out}, v_{in}).

Příklad 80 *Hamiltonovský cyklus* v neorientovaném grafu je cyklus, který obsahuje každý vrchol grafu právě jednou, na orientaci hran nezáleží. Problém *UHAMCYCLE* je definován následovně:

$$\textit{UHAMCYCLE} = \{< G >\mid \textit{V neorientovaném grafu G existuje hamiltonovský cyklus}\}$$

Problém *TSP* je definovaný následovně.

$$\begin{aligned}\textit{TSP} = \{ < G, k > \mid\ & \text{V neorientovaném ohodnoceném grafu } G \\ & \text{existuje trasa obchodního cestujícího s hodnotou nejvýše } k\}\end{aligned}$$

Dokažte, že problém *TSP* je NP-úplný. Můžete k tomu využít fakt, že problém *UHAMCYCLE* je NP-úplný.

Příklad 81 Dokažte, že problém *3SAT* definovaný následujícím způsobem je NP-úplný.

$$\mathit{3SAT} = \{< \varphi >| \quad \varphi \textit{ je splnitelná formule v konjunktivní normální formě a každá její klauzule má právě tři literály } \}$$

Použijte redukci z problému *SAT*!

Příklad 82 Dokažte NP-úplnost problému *CNF*(8)

Formulace problému: Je daná booleovská formule Φ v konjunktivním normálním tvaru. Existuje alespoň 8 různých splňujících přiřazení hodnot proměnným formule Φ?

***Příklad 83** Řekneme, že neorientovaný graf G má k-kliku, pokud v něm existuje úplný podgraf s k vrcholy. Dokažte, že problém *KLIKA* definovaný následujícím způsobem je NP-úplný.

$$\mathit{KLIKA} = \{< G, k >| \textit{ G je neorientovaný graf s k-klikou}\}$$

***Příklad 84** Dokažte, že jazyk *1–3SAT* je NP–úplný.

$$\mathit{1\text{–}3SAT} = \{ F \mid F \text{ je formule v konjunktivním normálním tvaru a taková, že každá její klauzule obsahuje právě 3 literály, a existuje takové přirazení hodnot proměnným, že v každé klauzuli formuly } F \text{ je splněn právě jeden literál.}\}$$

Návod: Důkaz redukcí z *3SAT*. Klauzuli $(x \vee y \vee z)$ nahradíme klazulemi $(x \vee a \vee d)$, $(y \vee b \vee d)$, $(a \vee b \vee e)$, $(c \vee d \vee f)$, $(z \vee c \vee g)$, $(g \vee h \vee \neg h)$, kde a, b, c, d, e, f, g, h jsou nové proměnné.

Příklad 85 Dokažte, že jazyk *01–3SAT* je NP–úplný.

$$\mathit{01\text{–}3SAT} = \{ F \mid F \text{ je formule v konjunktivním normálním tvaru a taková, že každá její klunzule obsahuje právě 3 literály a existuje takové přirazení hodnot proměnným, že v žádné klauzuli formule } F \text{ nejsou splněny všechny tři literály.}\}$$

Návod: Důkaz redukcí z *3SAT*. Klauzuli $(x \vee \neg y \vee z)$ nahradíme klazulí $(\neg x \vee y \vee \neg z)$

Příklad 86 Dokažte, že jazyk *IZO* je NP-úplný.

$$\mathit{IZO} = \{< G, \overline{G} >| \textit{ graf G obsahuje podgraf izomorfní s grafem } \overline{G}\}$$

Definice: Nechť $G_1 = (V_1, H_1)$ a $G_2 = (V_2, H_2)$ jsou grafy. Graf G_1 je izomorfní s grafem G_2 právě když existuje bijekce $g : V_1 \to V_2$ taková, že hrana $(u, v) \in H_1$ právě když hrana $(g(u), g(v)) \in H_2$. Graf G_1 je podgrafem grafu G_2 když $V_1 \subseteq V_2$ a $H_1 \subseteq H_2$.

Návod: KLIKA $\leq$ IZO

Příklad 87 Dokažte NP-úplnost problému *NON-EKVIVALENCE*.

Formulace problému: Jdou dány dvě booleovské formule Φ_1 a Φ_2 proměnných $x_1, \ldots, x_n$. Jsou formule Φ_1 a Φ_2 různé? (Tj. existuje takové přiřazení hodnot $a_1, \ldots, a_n$ proměnným $x_1, \ldots, x_n$, pro které $\Phi_1(a_1, \ldots, a_n) \neq \Phi_2(a_1, \ldots, a_n)$?)

Návod: SAT $\leq$ NON-EKVIVALENCE

Příklad 88 Řekneme, že neorientovaný graf $G = (V, H)$ má k-vrcholové pokrytí, pokud existuje množina $X \subseteq V$ taková, že $|X| = k$ a $\{(x, y) \in H \mid x \in X \vee y \in X\} = H$. Dokažte, že jazyk *VRCHOLOVÉ POKRYTÍ* je NP-úplný.

$$\mathit{VRCHOLOVÉ\ POKRYTÍ} = \{< G, k >| \textit{ graf G obsahuje } k\textit{-vrcholové pokrytí}\}$$

Návod: SAT $\leq$ VRCHOLOVÉ POKRYTÍ. Nechť F je formule proměnných $x_1, \ldots, x_n$ s m klauzulemi.

1. Pro každou proměnnou x_i formule F obsahuje graf G dvojici vrcholů x_i^t a x_i^f a hranu (x_i^t, x_i^f). Intuitivně, vybraný vrchol bude odpovídat přiřazení hodnoty proměnné x_i.
2. Pro každou klauzuli C_j formule F obsahuje G kompletní podgraf G_j s n_j vrcholy, ze kterých každý odpovídá jednomu literálu klauzule. Intuitivně, na pokrytí hran této kliky je potřeba $n_j - 1$ vrcholů, zbylý vrchol bude odpovídat tomu literálu, který bude mít přiřazenu pravdivostní hodnotu $true$.
3. Pokud x_i resp. $\neg x_i$ je literál formule F, tak G obsahuje hranu spojující vrchol odpovídající literálu x_i resp. $\neg x_i$ s vrcholem x_i^t resp. x_i^f.

Číslo k definujeme jako

$$k = n + (n_1 - 1) + (n_2 - 1) + \cdots + (n_m - 1)$$

Příklad 89 Nechť $G = (V, H)$ je neorientovaný graf. Množina $X \subseteq V$ je nezávislá, pokud $\{(x, y) \in H \mid x \in X \wedge y \in X\} = \emptyset$. Dokažte, že jazyk *NEZÁVISLÁ MNOŽINA* je NP-úplný.

$$\textit{NEZÁVISLÁ MNOŽINA} = \{< G, k >\mid \textit{graf } G \textit{ obsahuje nezávislou množinu o } k \textit{ prvcích}\}$$

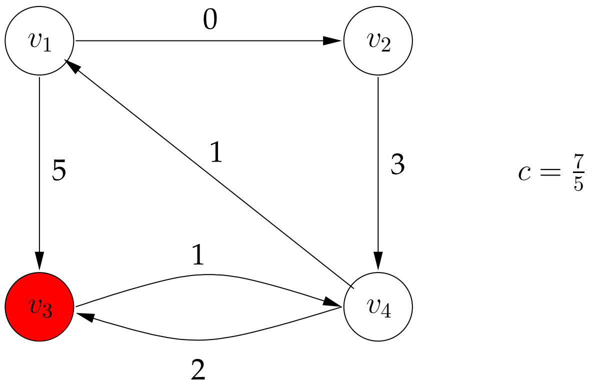
Návod: *KLIKA* $\leq$ *NEZÁVISLÁ MNOŽINA*

Příklad 90 Dokažte, že problém *NM* je NP-úplný.

Formulace problému: Je daný neorientovaný graf $G = (V, H)$, přičemž $|V| = 2n$. Existuje $V' \subset V$ taková, že $|V'| = n$ a žádné dva vrcholy z V' nejsou v G spojeny hranou?

Návod: *NEZÁVISLÁ MNOŽINA* $\leq$ *NM*

Příklad 91 Uvažujme následující problém. Je zadán ohodnocený orientovaný graf $G = (V, H, w)$, kde $w : H \to \mathbb{Z}$ je funkce ohodnocení hran. Dále je zadán vrchol $v \in V$ a konstanta $c \in \mathbb{Q}$. Úkolem je zjistit, zda graf G obsahuje cyklus jehož součástí je vrchol v a průměrná hodnota hrany na tomto cyklu je menší než c. Příklad:



Uvedený graf obsahuje cyklus s průměrnou délkou hrany menší než $\frac{7}{5}$, totiž cyklus (v_1, v_2, v_4, v_1), jehož průměrná hodnota hrany je $\frac{4}{3}$. Nicméně součástí tohoto cyklu není vyznačený vrchol v_3 a cyklus s minimální průměrnou délkou hrany, jehož součástí je vrchol v_3, je cyklus (v_3, v_4, v_3), jeho průměrná délka hrany je $\frac{3}{2}$, což je více než $\frac{7}{5}$. Řešením této konkrétní instance problému je tedy odpověď NE.

Dokažte, že v obecnosti je uvedený problém NP-úplný. Můžete k tomu využít fakt, že problém existence Hamiltonovského cyklu v orientovaném grafu je NP-úplný.

Příklad 92 Uveďte příklad co–NP-úplného problému a dokažte jeho co–NP-úplnost.

Příklad 93 Ukažte, že každý PSPACE-těžký problém je také NP-težký.

Příklad 94 Nechť $A \oplus B = \{x0 \mid x \in A\} \cup \{x1 \mid x \in B\}$. Dokažte, že pro libovolné jazyky A, B platí $A \leq_p A \oplus B$.

Příklad 95 Nechť C_1, C_2 jsou složitostní třídy uzavřené vzhledem k redukci. Dále nechť L je jazyk, který je současně C_1-úplný a C_2-úplný. Plyne z těchto faktů rovnost tříd C_1 a C_2?

Příklad 96 Nechť jazyky L_1 a L_2 jsou NP–úplné. Dokažte, že za předpokladu $L_1 \setminus L_2 \in \mathsf{P}$ anebo $L_2 \setminus L_1 \in \mathsf{P}$ je i jazyk $L_1 \cup L_2$ NP–úplný.

Příklad 97 Definujme jazyky B_1 a B_2 následovně:

$$B_1 = \{\ (F_1, F_2) \mid F_1 \in \mathit{CNF} \quad \wedge \quad F_2 \notin \mathit{CNF}\ \}$$

$$B_2 = \{\ (F_1, F_2) \mid F_1 \in \mathit{CNF} \quad \vee \quad F_2 \notin \mathit{CNF}\ \}$$

Dokažte:

a) $B_1 \leq_p B_2$

b) $B_2 \leq_p B_1$.

Příklad 98 Dokažte, že pokud existuje NP-úplný problém L takový, že $L \in \mathsf{NP} \cap \text{co–NP}$, tak $\mathsf{NP} = \text{co–NP}$.

Kapitola 8

Asymptotické chování funkcí

V této sekci pracujeme s přirozenými čísly bez nuly, $\mathbb{N} = \{1, 2, \ldots\}$, a s funkcemi typu $f, g : \mathbb{N} \to \mathbb{R}^+$.

Příklad 99 Dokažte, že pokud $\lim_{n\to\infty} \frac{f(n)}{g(n)} < \infty$, tak $f(n) \in \mathcal{O}(g(n))$.

Příklad 100 Dokažte, že pokud $\lim_{n\to\infty} \frac{f(n)}{g(n)} > 0$, tak $f(n) \in \Omega(g(n))$.

Příklad 101 Rozhodněte, které z následujících vztahů platí. Odpovědi zdůvodněte.

a. $2n \in \mathcal{O}(n)$

b. $n^2 \in \mathcal{O}(n)$

c. $n \log_2 n \in \mathcal{O}(n^2)$

d. $n \log_2 n \in \mathcal{O}(n)$

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

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

Příklad 102 Rozhodněte, které z následujících vztahů platí. Odpovědi zdůvodněte.

a. $n \in o(2n)$

b. $2n \in o(n^2)$

c. $2^n \in o(3^n)$

d. $1 \in o(n)$

e. $1 \in o(\log n)$

f. $1 \in o(\frac{1}{n})$

g. $n^3 \in o(3^n)$

Příklad 103 Třída $\mathcal{O}(g(n))$ se někdy definuje následujícím způsobem. Nechť $f, g : \mathbb{N} \to \mathbb{R}^+$ jsou funkce. Píšeme $f(n) \in \mathcal{O}(g(n))$, pokud existují $c_1, c_2 \in \mathbb{N}$ takové, že pro všechna $n \in \mathbb{N}$ platí:

$$f(n) \leq c_1 \cdot g(n) + c_2$$

Je tato definice ekvivalentní standardní definici z přednášek? Změní se odpověď, pokud uvažujeme pouze funkce $g(n)$ splňující $g(n) \geq k$ pro nějakou konstantu $k \in \mathbb{R}^+$ a pro všechna $n \in \mathbb{N}$?

Příklad 104 Existuje funkce $f : \mathbb{N} \to \mathbb{R}^+$ taková, že platí $f(n) \in o(f(n))$?

Příklad 105 Dokažte, že platí následující vztahy:

$$
\begin{array}{rccc}
\text{(a)} & f(n) \in \mathcal{O}(g(n)) & \iff & g(n) \in \Omega(f(n)) \\
\text{(b)} & f(n) \in o(g(n)) & \implies & f(n) \in \mathcal{O}(g(n)) \\
\text{(c)} & f(n) \in o(g(n)) & \implies & f(n) \notin \Omega(g(n)) \\
\text{(d)} & f(n) \in o(g(n)) & \implies & g(n) \notin \mathcal{O}(f(n))
\end{array}
$$

Příklad 106 Rozhodněte, zda platí následující vztah. Odpověď zdůvodněte.

$$f(n) \in o(g(n)) \iff g(n) \notin \mathcal{O}(f(n))$$

Kapitola 9

Složitostní třídy

9.1 Lehké

Příklad 107 Dokažte, že každý bezkontextový jazyk je v P.

Příklad 108 Dokažte, že třída Pje uzavřená na operace sjednocení, komplement a zřetězení. Rozhodněte, na které z těchto operací je uzavřena třída NP. Odpověď zdůvodněte.

Příklad 109 Dokažte, že třída PSPACEje uzavřená na operace sjednocení, komplement a iteraci.

Příklad 110 Určete vztah mezi následujícími dvojicemi složitostních tříd. Svoje tvrzení zdůvodněte. Rozlišujte mezi rovností, inkluzí a ostrou inkluzí.

DTIME(n^2) a DTIME(n^3)

DSPACE$(2n^2)$ a DSPACE$(100n^2)$

DTIME(n^2) a DSPACE(n^2)

NSPACE(n^2) a DSPACE(n^5)

NSPACE(n^2) a DSPACE(n^3)

P a DTIME(2^n)

Příklad 111 Dokažte:

DSPACE$(2^n) =$ DSPACE(2^{n+1})

Příklad 112 Najděte chybu v následujícím důkazu tvrzení P $\neq$ NP.

Predpokládejme, že P= NP. Pak pro nějaké $k \in \mathbb{N}$ platí $SAT \in$ DTIME(n^k). Protože každý jazyk z třídy NP je polynomiálně redukovatelný na jazyk SAT, tak NP $\subseteq$ DTIME(n^k). Podle předpokladu je P= NP, a tedy i P $\subseteq$ DTIME(n^k). Lze dokázat (opravdu lze, zde chybu nehledejte), že DTIME$(n^k) \subset$ DTIME(n^{k+1}). To je spor s P $\subseteq$ DTIME(n^k). Proto P $\neq$ NP.

Příklad 113 Najděte chybu v následujícím důkazu tvrzení P $\neq$ NP.

Uvažujme následující algoritmus pro problém CNF. Pro vstupní formuli F prověříme všechna možná přiřazení hodnot proměnným. Když některé z nich splňuje F, tak akceptujeme. Zřejmě tento algoritmus má exponenciální složitost. Proto problém CNF má exponenciální složitost. Problém CNF tedy nepatří do třídy P. Protože CNF je NP–úplný, tak musí být P $\neq$ NP.

Příklad 114 Najděte chybu v následujícím důkazu tvrzení, že jazyk

$$\forall \textit{KLIKA} = \{\langle G, k\rangle \mid \text{každá klika v grafu } G \text{ má velikost menší než } k\}$$

patří do třídy NP.

> Víme, že jazyk
>
> $$\exists \textit{KLIKA} = \{\langle G, k\rangle \mid \text{v grafu } G \text{ existuje klika velikosti alespoň } k\}$$
>
> patří do třídy NP. Nechť $\mathcal{M}$ je nedeterministický Turingův stroj polynomiální časové složitosti, akceptující jazyk $\exists$*KLIKA*. Pak záměnou akceptujícího a zamítajícího stavu stroje $\mathcal{M}$ dostaneme nedeterministický TS stroj polynomiální časové složitosti a rozhodující jazyk $\forall$*KLIKA*. Proto jazyk $\forall$*KLIKA* patří do třídy NP.

Příklad 115 Popište chování nedeterministického Turingova stroje s výstupní páskou (speciální páska, na které se na konci výpočtu nachází výstup výpočtu) řešícího rozklad čísla n na prvočinitele v polynomiálním čase vzhledem k n. Nevyžaduje se formální zápis Turingova stroje, stačí popis činnosti od načtení vstupu po zápis rozkladu na výstupní pásku.

Příklad 116 Nechť $T, S : \mathbb{N} \to \mathbb{N}$ jsou funkce. Definujeme složitostní třídu $\mathsf{DTISP}(T(n), S(n))$ takto: jazyk L patří do $\mathsf{DTISP}(T(n), S(n))$ právě když existuje k-páskový Turingův stroj M akceptují jazyk L v čase $T(n)$ a v prostoru $S(n)$. Poznamenáváme, že stroj musí *současně* splňovat časové i prostorové ohraničení. Ukažte, že pro každý jazyk $L \in \mathsf{DTISP}(T(n), S(n))$ existuje *jednopáskový* Turingův stroj akceptující L v čase $\mathcal{O}(T(n)S(n))$ a *současně* v prostoru $S(n)$.

Příklad 117 Nechť $L \subseteq \Sigma^*$ je jazyk. *Polynomiální prodloužení* jazyka L je jazyk L_k definovaný předpisem

$$L_k = \{x0^r \mid x \in L, r = |x|^k\}$$

kde k je fixní přirozené číslo a symbol 0 nepatří do abecedy Σ.

(a) Dokažte, že třída NP je uzavřená vůči operaci polynomiálního prodloužení, tj. pro každé dva jazyky L a $\overline{L}$ takové, že $\overline{L}$ je prodloužením jazyka L platí současně

- $L \in \mathsf{NP} \Rightarrow \overline{L} \in \mathsf{NP}$
- $\overline{L} \in \mathsf{NP} \Rightarrow L \in \mathsf{NP}$

(b) Zdůvodněte, proč $\mathsf{DSPACE}(n) \neq \mathsf{NP}$

Příklad 118 Dokažte, že problém *DNF-SAT* definovaný následujícím způsobem je v P.

$$\textit{DNF-SAT} = \{<\varphi>\mid \varphi \textit{ je splnitelná formule v disjunktivní normální formě}\}$$

Příklad 119 *Trojúhelník* v neorientovaném grafu je 3-klika. Ukažte, že jazyk *TRIANGLE* je v P.

$$\textit{TRIANGLE} = \{< G >\mid G \textit{ obsahuje trojúhelník}\}$$

Příklad 120 Ukažte, že testování prvočíselnosti je řešitelné v polynomiálním čase, pokud použijeme unární kódování čísel. Jinými slovy, ukažte, že jazyk *UNARY-PRIMES* je v P.

$$\textit{UNARY-PRIMES} = \{1^n \mid n \textit{ je prvočíslo}\}$$

Poznámka: Poznamenejme, že testování prvočíselnosti je řešitelné v polynomiálním čase i pro binární kódování čísel. Důkaz je ale v tomto případě poněkud složitější.

Příklad 121 Dokažte, že problém *CONNECTED* definovaný následujícím způsobem je v P.

$$\textit{CONNECTED} = \{< G >\mid G \textit{ je spojitý neorientovaný graf}\}$$

Příklad 122 Pro dvě přirozená čísla $a, b > 1$ definujeme $\mathrm{lcm}(a, b)$ jako nejmenší společný násobek čísel a a b. Dokažte, že problém *LCM* definovaný následujícím způsobem je v P.

$$LCM = \{< a, b, c >| \ c = \mathrm{lcm}(a, b)\}$$

Příklad 123 Řekneme, že grafy G_1 a G_2 jsou izomorfní, pokud vrcholy grafu G_2 mohou být přeuspořádány tak, aby byl identický s G_1. Dokažte, že problém *ISO* definovaný následujícím způsobem je v NP.

$$ISO = \{< G_1, G_2 >| \ G_1 \textit{ a } G_2 \textit{ jsou izomorfní grafy}\}$$

Příklad 124 Dokažte, že následující jazyk je v NP.

$$FACTOR = \{< m, n >| \ m \textit{ je dělitelné číslem r takovým, že } 1 < r < n\}$$

Příklad 125 Označme *DNF-SAT* problém splnitelnosti booleovské formule v disjunktivní normální formě. Najděte chybu v následující úvaze.

> Použitím distributivního zákona můžeme každou CNF formuli transformovat na formuli v DNF tvaru. Například, $(x_1 \vee x_2) \wedge (\neg x_1 \vee x_3)$ můžeme přepsat do tvaru $(x_1 \wedge \neg x_1) \vee (x_1 \wedge x_3) \vee (x_2 \wedge \neg x_1) \vee (x_2 \wedge x_3)$. To znamená, že *SAT* $\leq$*DNF-SAT*. Problém *DNF-SAT* se dá řešit v polynomiálním čase. Proto i problém *SAT* sa dá řešit v polynomiálním čase, a proto P= NP.

Příklad 126 Je daný nedeterministický konečný automat $\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$ a slovo $w \in \Sigma^*$. O slovu w řekneme, že je *víceznačné*, právě když existují dva různé akceptující výpočty $\mathcal{M}$ na w. Automat $\mathcal{M}$ nazveme víceznačný, právě když nějaké slovo w z $L(\mathcal{M})$ je víceznačné. Prozkoumejte pojem víceznačnosti. Konkrétně: je pro daný automat $\mathcal{M}$ rozhodnutelné, jestli je víceznačný? Když ano, jaká je zložitost tohoto problému? Patří do P, NP, je NP–úplný?

Příklad 127 Dokažte anebo vyvraťte:

$$\mathsf{NP} \neq \mathsf{co\text{–}NP} \implies \mathsf{P} \neq \mathsf{NP}.$$

9.2 Průměrně obtížné

***Příklad 128** Dokažte, že problém *2SAT* definovaný následujícím způsobem je v P.

$$2SAT = \{< \varphi >| \ \begin{array}{l}\varphi \textit{ je splnitelná formule v konjunktivní normální} \\ \textit{formě a každá její klauzule má právě dva literály }\}\end{array}$$

***Příklad 129** Ukažte, že libovolná booleovská formule se dá transformovat v polynomiálním čase na formuli v CNF, která je splnitelná tehdy a jen tehdy, když byla původní formule splnitelná.

***Příklad 130** Booleovská formule se nazývá *tautologie*, právě když je splněna pro každé přiřazení hodnot proměnným. Existuje polynomiální algoritmus, který rozhodne, jestli je daná formule tautologie?

***Příklad 131** Booleovská formule se nazývá *kontradikce*, právě když není splněna pro žádné přiřazení hodnot proměnným. Existuje polynomiální algoritmus, který rozhodne, jestli je daná formule kontradikce?

***Příklad 132** Předpokládejme, že existuje polynomiální algoritmus pro jazyk *SAT*. Navrhněte polynomiální algoritmus, který pro danou formuli najde splňující přiřazení hodnot proměnným (pokud existuje).

***Příklad 133** Dokažte, že pro deterministický Turingův stroj $\mathcal{M}$, který vždy zastaví, existuje ekvivalentní deterministický Turingův stroj $\mathcal{N}$, který s prostorovou složitostí:

$$S_{\mathcal{N}} \leq \lceil \frac{S_{\mathcal{M}}}{2} \rceil$$

Návod: Modifikujte důkaz věty o prostorové kompresi – ve výše uvedené nerovnosti oproti větě o kompresi chybí konstantní člen

***Příklad 134** Jaký je vztah mezi třídami DTIME($f(n)$) a DSPACE($f(n)$) ($\subset, \subseteq, =, \supseteq, \supset$)? *Zdůvodněte své tvrzení.*

Uveďte příklad $f(n)$ a Turingova stroje $\mathcal{M}$ takového, že $S_{\mathcal{M}} = o(f(n))$ a $T_{\mathcal{M}} = \Omega(f(n))$.

***Příklad 135** Určete co nejpřesněji jednotkovou a logaritmickou paměťovou složitost algoritmu:

```
n := read();
i := 0;
r := 1;
for i:=1 to n do r := 3*r;
```

***Příklad 136** Ukažte, že když jazyk L patří do třídy P, pak i jazyk L^* patří do P. Platí analogické tvrzení i pro třídu NP?

***Příklad 137** Pro každý z níže definovaných jazyků rozhodněte, do které z následujícíh zložitostných tříd patří: P, NP, NP–úplné jazyky, NP–těžké jazyky. Jestliže některý z jazyků neumíte zařadit do žádné z uvedených tříd, tak uveďte, jestli jeho přesná klasifikace je otevřeným problémem. V každém případě odůvodněte svou odpověď.

a) $A = \{\langle M, w\rangle \mid M$ je NTS akceptující $w\}$,

b) $B = \{\langle M, w, 0^k\rangle \mid M$ je DTS akceptující slovo w v nanejvýš k krocích$\}$,

c) $C = \{\langle M, w, 0^k\rangle \mid M$ je NTS akceptující slovo w v nanejvýš k krocích$\}$.

DTS — deterministický Turingův stroj, NTS — nedeterminstický Turingův stroj

***Příklad 138** Rozhodněte, zda následující argumenty jsou důkazem toho, že problém neekvivalence bezkontextových gramatik je ve třídě NP.

> Nechť $\mathcal{G}_1$ a $\mathcal{G}_2$ jsou bezkontextové gramatiky nad abecedou Σ. Víme, že problém rozhodnout, zda dané slovo je generováno danou bezkontextovou gramatikou, je v P. Nedeterministický stroj se vstupem $< \mathcal{G}_1, \mathcal{G}_2 >$ bude proto pracovat následovně. Nedeterministicky vybere nějaké slovo $w \in \Sigma^*$ (cílem je uhodnout slovo, na kterém se $\mathcal{G}_1$ a $\mathcal{G}_2$ liší) a v polynomiálním čase ověří, zda $w \in \mathcal{L}(\mathcal{G}_1)$ a $w \in \mathcal{L}(\mathcal{G}_2)$. Pokud dostane různé výsledky, tak akceptuje. Pokud obě gramatiky slovo w generují, nebo naopak obě gramatiky slovo w negenerují, tak zamítá.

9.3 Obtížné

****Příklad 139** Dokažte, že pro deterministický k-páskový Turingův stroj $\mathcal{M}$, který vždy zastaví, existuje ekvivalentní deterministický $(k+1)$-páskový Turingův stroj $\mathcal{N}$, který s časovou složitostí:

$$T_{\mathcal{N}} \leq \lceil \frac{T_{\mathcal{M}}}{2} \rceil + n$$

Návod: Modifikujte důkaz věty o prostorové kompresi.

****Příklad 140** Dokažte, že následující jazyk je v co-NP.

$$\mathit{FACTOR} = \{< m, n > \mid m \text{ je dělitelné číslem } r \text{ takovým, že } 1 < r < n\}$$