Příklad: Dokažte následující tvrzení:

$$\forall\, a \in \mathbf{Aexp}, \forall\, \sigma \in \Sigma, \forall\, n \in \mathbb{N} : \mathcal{A}[a](\sigma) = n \Rightarrow \langle a, \sigma \rangle \to n$$

Řešení: dukcí ke struktuře a:

1. Pokud $a \equiv n$, potom $\mathcal{A}[n]\sigma = n$ a $\langle n, \sigma \rangle \to n$ z definice.
2. Pokud $a \equiv X$, potom $\mathcal{A}[X]\sigma = \sigma(X)$ a $\langle X, \sigma \rangle \to \sigma(X)$
3. Pokud $a \equiv a_1 \odot a_2$ a máme tvrzení dokázané pro a_1 i a_2, potom pokud $\mathcal{A}[a]\sigma = n$, potom $\mathcal{A}[a_1]\sigma \odot \mathcal{A}[a_2]\sigma = n$. Podle indukčního předpokladu $\langle a_1, \sigma \rangle \to n_1$ a $\langle a_2, \sigma \rangle \to n_2$ takové, že $n_1 \odot n_2 = n$, potom:
$$\frac{\langle a_1, \sigma \rangle \to n_1 \quad \langle a_2, \sigma \rangle \to n_2}{\langle a_1 \odot a_2, \sigma \rangle \to n} \quad n = n_1 \odot n_2$$

Příklad: Sestrojte Büchiho automaty pro formule $p\mathcal{U}(p \wedge q \wedge \mathcal{X}q)$ a $\mathcal{GF}(p \vee q)$.

Řešení: Neumím teXovat automaty.

Příklad: Definujte denotační sémantiku následujících programů:

1. $X := X + 1; Y := 3; Z := X + Y$
2. **if** $X = 1$ **then** $Y := 2$ **else** $Z := 1$
3. **while** $X = 2$ **do** $X := 3; Y := Y - 1$

Řešení:

1. Označme $c \equiv X := X + 1; Y := 3; Z := X + Y$, potom:
$$\begin{aligned}\mathcal{C}[c]\sigma &= (\mathcal{C}[Z := X + Y] \circ \mathcal{C}[Y := 3] \circ \mathcal{C}[X := X + 1])\sigma \\ &= (\mathcal{C}[Z := X + Y] \circ \mathcal{C}[Y := 3])\sigma[X/\sigma(X) + 1] \\ &= \mathcal{C}[Z := X + Y]\sigma[X/\sigma(X) + 1, Y/3] \\ &= \sigma[X/\sigma(X) + 1, Y/3, Z/\sigma[X/\sigma(X) + 1, Y/3](X) + \sigma[X/\sigma(X) + 1, Y/3](Y)] \\ &= \sigma(X/\sigma(X) + 1, Y/3, Z/\sigma(X) + 4)\end{aligned}$$

2. $$\mathcal{C}[\textbf{if } X = 1 \textbf{ then } Y := 2 \textbf{ else } Z := 1]\sigma = \begin{cases} \sigma(Y/2) & \text{pokud } \mathcal{B}[X = 1]\sigma = \textbf{true} \\ \sigma(Z/1) & \text{jinak} \end{cases}$$

3. $$\mathcal{C}[\textbf{while } X = 2 \textbf{ do } X := 3; Y := Y - 1] = \{(\sigma, \sigma) \in \Sigma \times \Sigma \mid \sigma(X) \neq 2\}$$

Příklad: Buď $\Gamma : A \to A$ monotónní funkce, kde A je konečná množina. Je Γ nutně spojitá?

Řešení: Ano, platí. Musíme ověřit podmínku zachovávání suprema řetězců. Pokud je $a_1 \leq a_2 \leq \ldots$ nekonečný řetězec v A, potom $\Gamma(a_1) \leq \Gamma(a_2) \leq \ldots$ je opět nekonečný řetězec v A. Protože je A konečná, existuje $n_0 \in \mathbb{N}$ takové, že $a_n = a_{n_0}$ a $\Gamma(a_n) = \Gamma(a_{n_0})$ pro všechna $n \geq n_0$. Potom zřejmě:

$$\Gamma(\bigvee_{n\in\mathbb{N}} a_n) = \Gamma(a_{n_0}) = \bigvee_{n\in\mathbb{N}} \Gamma(a_n)$$

Příklad: Vyjádřete v jazyce **Assn** nejslabší vstupní podmínku pro následující dvojice (program, podmínka):

1. $X := 3; Y := X$, $Y = 6$
2. **while** $X = 1$ **do** $X := 1$, $X \neq 2$
3. **if** $X = 1$ **then** $Y := Y - 1$ **else** $Y := Y + 1$, $Y = 6$

Řešení:

1. **false**
2. $X \neq 2$
3. $(X = 1 \wedge Y = 7) \vee (X \neq 1 \wedge Y = 5)$

Příklad: Dokažte nebo uveďte protipříklad pro následující tvrzení: Jsou-li příkazy **while** $X = 1$ **do** c a **while** $X = 1$ **do** c' ekvivalentní z hlediska operační sémantiky 1. typu, pak jsou c a c' ekvivalentní z hlediska denotační sémantiky.

Řešení: Neplatí. Např. **while** $X = 1$ **do** **skip** a **while** $X = 1$ **do** $X := 1$ jsou ekvivalentní z hlediska operační sémantiky I. typu, ale **skip** a $X := 1$ nejsou ekvivalentní z hlediska žádné sémantiky.

Příklad: Definujte denotační sémantiku a operační sémantiku I. typu pro operátor $b\,?\,a_1 : a_2$, kde $b \in$ **Bexp** a $a_1, a_2 \in$ **Aexp**. Neformálně řečeno, $b\,?\,a_1 : a_2$ je aritmetický výraz, který vrací hodnotu a_1 v případě, že b je **true**, jinak vrací hodnotu a_2.

Řešení: Denotační sémantika:

$$\mathcal{A}[b\,?\,a_1 : a_2]\sigma = \begin{cases} \mathcal{A}[a_1]\sigma & \text{pokud } \mathcal{B}[b]\sigma = \textbf{true} \\ \mathcal{A}[a_2]\sigma & \text{jinak} \end{cases}$$

Operační sémantika I. typu:

$$\frac{\langle b, \sigma\rangle \to \textbf{true} \quad \langle a_1, \sigma\rangle \to n}{\langle b\,?\,a_1 : a_2, \sigma\rangle \to n}$$

$$\frac{\langle b, \sigma\rangle \to \textbf{false} \quad \langle a_2, \sigma\rangle \to n}{\langle b\,?\,a_1 : a_2, \sigma\rangle \to n}$$

Příklad: Uvažme následující predikát jazyka **Assn** s volnými proměnnými k, ℓ, kde β je Gödelův predikát:

$$\exists\, m, n : \beta(m, n, 0, 0) \wedge \beta(m, n, k, \ell) \wedge \forall\, i : [(0 \leq i < k) \Rightarrow (\beta(m, n, i, j) \Rightarrow \beta(m, n, i + 1, j + 4))]$$

Uveďte, jaký vztah mezi proměnnými k, ℓ tento výraz popisuje (tj. vyjádřete ℓ jako funkci k). Uveďte, jak jste k vašemu výsledku dospěli.

Řešení: Platí $\ell = 4k$, což by se ukázalo indukcí.

Příklad: Uveďte příklad (případně dokažte neexistenci) programu $c \in \mathbf{Com}$, kde pro každé $I \in \mathcal{I}$ platí:

1. $wp^I[c, X = 3] = \{\perp\} \cup \{\sigma \in \Sigma \mid \sigma(Y) = 2\}$
2. $wp^I[c, \mathbf{false}] = \{\perp\} \cup \{\sigma \in \Sigma \mid \sigma(X) = 1 \wedge \sigma(Y) < 1\}$
3. $wp^I[c, \mathbf{true}] = \{\perp\} \cup \{\sigma \in \Sigma \mid \sigma(X) > 5\}$

Řešení:

1. $c \equiv X := Y + 1$
2. $c \equiv$ **while** $X = 1 \wedge Y < 1$ **do skip**
3. Neexistuje, neboť **true** je splněna v každém stavu.

Příklad: V Hoareově odvozovacím systému dokažte tvrzení:

$$\{X = 1\}\, \mathbf{while}\ \neg\,(X = 1)\ \mathbf{do}\ Y := 5\, \{Y = 2\}$$

Řešení: Neplatí, a proto nelze ani dokázat.

Příklad: Dokažte nebo uveďte protipříklad pro následující tvrzení: Je-li A invariant cyklu **while** b **do** c pak také $\neg A$ je invariantem téhož cyklu.

Řešení: Neplatí. Invariantem cyklu **while** b **do** c je taková podmínka A, že platí $\{A \wedge b\}\, c\, \{A\}$. Zvolíme-li za $b \equiv \mathbf{true}$, $c \equiv X := 1$ a $A \equiv X = 1$, potom $\{X = 1\}\, X := 1 \{X = 1\}$ platí, ale zřejmě neplatí $\{X \neq 1\}\, X := 1\, \{X \neq 1\}$.

Příklad: Uvažme cyklus **while** $\neg\,(X = Y)$ **do** $X := Y$. Doplňte následující definice:

1. $\Gamma(\emptyset) = \{(\sigma, \sigma') \mid \ldots\}$
2. $\Gamma^2(\emptyset) = \Gamma(\emptyset) \cup \{(\sigma, \sigma') \mid \ldots\}$
3. $\Gamma^3(\emptyset) = \Gamma^2(\emptyset) \cup \{(\sigma, \sigma') \mid \ldots\}$

Řešení:

1. $\sigma = \sigma' \wedge \sigma(X) = \sigma(Y)$
2. $\sigma' = \sigma[X/\sigma(Y)] \wedge \sigma(X) \neq \sigma(Y) \wedge \sigma'(X) = \sigma'(Y)$
3. $\Gamma^3(\emptyset) = \Gamma^2(\emptyset)$

Příklad: Uveďte příklad konečného CPO $(D, \leq)$ s nejmenším prvkem $*$ a spojité funkce $f : D \to D$, která největší pevný bod.

Řešení: $D = \{*, 1, -1\}$, kde $* \leq 1$ a $* \leq -1$, ale 1 a -1 jsou neporovnatelné. Pokud za f zvolíme identitu, každý prvek D budem pevným bodem f, ale protože D nemá největší prvek, f nemůže mít největší pevný bod.

Příklad: Uveďte příklad programu $c \in \mathbf{PCom}$ tak, aby pro každé $\sigma \in \Sigma$ a $I \in \mathcal{I}$ platilo $\langle c, \sigma \rangle \models^I \mathcal{GF}(X = 3)$ a současně $\langle c, \sigma \rangle \not\models^I \mathcal{FG}(X = 3)$.

Řešení:

$$\textbf{while true do } (X := 2; X := 3)$$

Příklad: Uvažme variantu jazyka IMP, kde místo **while**-cyklu je cyklus **from-to** se syntaxí:

$$\textbf{from } X \textbf{ to } n \textbf{ do } C$$

kde $n \in \mathbb{N}$ a v příkazu C se proměnná X nevyskytuje na levé straně přiřazovacího příkazu. Výpočet jedné iterace **from-to**-cyklu se provádí takto: nejprve se otestuje, zda $X \leq n$. Pokud tato podmínka není splněna, cyklus se ukončí. Jinak se vypočítá tělo cyklu C, hodnota X se zvýší o jedna a pokračuje se další iterací. Definujte operační sémantiku **from-to**-cyklu prvního typu (tj. big step), která je v souladu s touto neformální definicí.

Řešení:

$$\frac{\langle X \leq n, \sigma \rangle \to \textbf{false}}{\langle \textbf{from } X \textbf{ to } n \textbf{ do } c, \sigma \rangle \to \sigma}$$

$$\frac{\langle X \leq n, \sigma \rangle \to \textbf{true} \quad \langle c; X := X + 1, \sigma \rangle \to \sigma'' \quad \langle \textbf{from } X \textbf{ to } n \textbf{ do } c, \sigma'' \rangle \to \sigma'}{\langle \textbf{from } X \textbf{ to } n \textbf{ do } c, \sigma \rangle \to \sigma'}$$

Příklad: Čemu je rovno $\mathcal{A}[3*X](\mathcal{C}[\textbf{while false do } X := X+1]\sigma)$, kde $\sigma(X) = 2$? Zejména uveďte, jakého typu je výsledek (zda je to funkce, pravdivostní hodnota nebo něco jiného; pokud jde o funkci, uveďte, co je definiční obor a co obor hodnot).

Řešení: Protože $\mathcal{A}$ vyhodnocuje aritmetické výrazy, výsledek bude typu $\mathbb{Z}$.

$$\mathcal{A}[3 * X](\mathcal{C}[\textbf{while false do } X := X + 1]\sigma) = \mathcal{A}[3 * X]\sigma = \mathcal{A}[3]\sigma * \mathcal{A}[X]\sigma = 3 * \sigma(X) = 6$$

Příklad: Dokažte nebo uveďte protipříklad pro následující tvrzení: Pro každé $a, a_1, a_2 \in \mathbf{Aexp}$ platí, že jestliže $\mathcal{A}[a_1 + a] = \mathcal{A}[a_2 + a]$, pak také $\mathcal{A}[a_1] = \mathcal{A}[a_2]$. (Protipříklad nesmí obsahovat výrazy s více jak jedním aritmetickým operátorem, důkaz nesmí být veden strukturální indukcí.)

Řešení: Buď $\sigma \in \Sigma$ libovolné. Potom platí:

$$\mathcal{A}[a_1]\sigma + \mathcal{A}[a]\sigma = \mathcal{A}[a_1 + a]\sigma = \mathcal{A}[a_2 + a]\sigma = \mathcal{A}[a_2]\sigma + \mathcal{A}[a]\sigma$$

a odečtením dostaneme rovnost:

$$\mathcal{A}[a_1]\sigma = \mathcal{A}[a_2]\sigma$$

pro každé $\sigma \in \Sigma$.

Příklad: Uveďte příklad (nebo dokažte neexistenci) výrazu $b \in \mathbf{Bexp}$ a příkazu $c \in \mathbf{Com}$ takových, že pro cyklus **while** b **do** c současně platí:

1. Pro každé $I \in \mathcal{I}$ platí:

$$wp^I[\textbf{while } b \textbf{ do } c, \textbf{false}] = \{\perp\} \cup \{\sigma \in \Sigma \mid \sigma(X) < 2\}$$

2. $\Gamma(\emptyset) = \Gamma^2(\emptyset)$

Řešení:

$$\textbf{while } X < 2 \textbf{ do skip}$$

Příklad: V Hoareově odvozovacím systému dokažte tvrzení:

$$\{X = 3\}\, \textbf{while } X = 3 \textbf{ do } Y := X + 1\, \{2 = 3\}$$

Řešení:

$$\dfrac{\dfrac{\{X = 3\}\, Y := X + 1\, \{X = 3\}}{\{X = 3\}\, \textbf{while } X = 3 \textbf{ do } Y := X + 1\, \{X = 3 \land X \neq 3\}} \quad \models (X = 3 \land X \neq 3) \Rightarrow 2 = 3}{\{X = 3\}\, \textbf{while } X = 3 \textbf{ do } Y := X + 1\, \{2 = 3\}}$$

kde $\{X = 3\}\, Y := X + 1\, \{X = 3\}$ je axiomem pro přiřazení.

Příklad: Dokařte nebo uveďte protipříklad pro následující tvrzení: Por každé $c \in \textbf{Com}$, $\sigma \in \Sigma$, $I \in \mathcal{I}$ a LTL formuli φ platí, že jestliže $\langle c, \sigma\rangle \models^I \varphi$, pak také $\langle c \,||\, c, \sigma\rangle \models^I \varphi$.

Řešení: Protipříkladem je konfigurace: $c \equiv X := X + 1$, $\sigma[X/0]$, libovolné $I \in \mathcal{I}$ a formule $\varphi \equiv \mathcal{F}\mathcal{G}(X = 1)$ nebo $\mathcal{G}(X \leq 1)$.

Příklad: Uveďte příklad konečného CPO $(C, \sqsubseteq)$ s nejmenším prvkem $*$ a spojité funkce $f : C \to C$ tak, aby platilo $\mu f = f^3(*) \neq f^2(*)$.

Řešení: Zvolíme množinu $\{*, 1, 2, 3\}$ a relaci $\sqsubseteq$ jako uspořádání na této množině generované relací:

$$\{(*, 1), (1, 2), (2, 3)\}$$

Funkci zvolíme předpisy $f(*) = 1$, $f(1) = 2$, $f(2) = 3$ a $f(3) = 3$. Potom:

$$\mu f = f^3(*) = 3 \neq 2 = f^2(*)$$

Příklad: Mějme dva Büchiho automaty $\mathcal{A}_1 = (Q_1, \{a\}, \to_1, q_1, F_1)$ a $\mathcal{A}_2 = (Q_2, \{a\}, \to_2, q_2, F_2)$. Definujme automat:

$$\mathcal{A}_1 \ominus \mathcal{A}_2 = (Q_1 \times Q_2, \{a\}, \to, (q_1, q_2), F_1 \times (Q_2 \setminus F_2))$$

Rozhodněte, zda platí:

$$\mathcal{L}(\mathcal{A}_1 \ominus \mathcal{A}_2) = \mathcal{L}(\mathcal{A}_1) \setminus \mathcal{L}(\mathcal{A}_2)$$

Řešení: Neplatí. Uvažme automaty $\mathcal{A}_1 = (\{q_1, q_2\}, \{a\}, \to_1, q_1, \{q_1\})$ a $\mathcal{A}_1 = (\{q_1, q_2\}, \{a\}, \to_2, q_1, \{q_2\})$, kde $q_1 \xrightarrow{a}_i q_2$ a $q_2 \xrightarrow{a}_i q_1$ pro $i \in \{1, 2\}$. Tedy tyto automaty jsou identické až na koncový stav. Potom:

$$\mathcal{L}(\mathcal{A}_1 \ominus \mathcal{A}_2) = \{a^\omega\} \neq \emptyset = \{a^\omega\} \setminus \{a^\omega\} = \mathcal{L}(\mathcal{A}_1) \setminus \mathcal{L}(\mathcal{A}_2)$$

Příklad: Nakreslete část přechodového systému určeného SOS 2. typu obsahující všechny konfigurace dosažitelné z $\langle (X := 2 \,||\, X := 1); X := 3, \sigma\rangle$ a přechody mezi nimi.

Řešení: Přechodový systém má dvě větve:

$$\begin{aligned}\langle (X := 2 \,||\, X := 1); X := 3, \sigma \rangle &\rightarrow \langle (\mathbf{skip} \,||X := 1); X := 3, \sigma[X/2] \rangle \\ &\rightarrow \langle (\mathbf{skip} \,||\, \mathbf{skip}); X := 3, \sigma[X/1] \rangle \\ &\rightarrow \langle X := 3, \sigma[X/1] \rangle \\ &\rightarrow \langle \mathbf{skip}, \sigma[X/3] \rangle\end{aligned}$$

a

$$\begin{aligned}\langle (X := 2 \,||\, X := 1); X := 3, \sigma \rangle &\rightarrow \langle (X := 2 \,||\, \mathbf{skip}); X := 3, \sigma[X/1] \rangle \\ &\rightarrow \langle (\mathbf{skip} \,||\, \mathbf{skip}); X := 3, \sigma[X/2] \rangle \\ &\rightarrow \langle X := 3, \sigma[X/2] \rangle \\ &\rightarrow \langle \mathbf{skip}, \sigma[X/3] \rangle\end{aligned}$$

Příklad: Dokažte, že následující dva programy jsou ekvivalentní ve smyslu denotační sémantiky (nejprve napište, co vlastně dokazujete):

1. $\mathbf{if}\ b\ \mathbf{then}\ c_1\ \mathbf{else}\ c_2$
2. $\mathbf{if}\ \neg b\ \mathbf{then}\ c_2\ \mathbf{else}\ c_1$

Řešení: Buď $\sigma \in \Sigma$ libovolný. Potom:

$$\begin{aligned}\mathcal{C}[\mathbf{if}\ b\ \mathbf{then}\ c_1\ \mathbf{else}\ c_2]\sigma &= \begin{cases}\mathcal{C}[c_1]\sigma & \text{pokud } \mathcal{B}[b]\sigma = \mathbf{true} \\ \mathcal{C}[c_2]\sigma & \text{jinak}\end{cases} \\ &= \begin{cases}\mathcal{C}[c_2]\sigma & \text{pokud } \mathcal{B}[\neg b]\sigma = \mathbf{true} \\ \mathcal{C}[c_1]\sigma & \text{jinak}\end{cases} \\ &= \mathcal{C}[\mathbf{if}\ \neg b\ \mathbf{then}\ c_2\ \mathbf{else}\ c_1]\sigma\end{aligned}$$

A tedy:

$$\mathcal{C}[\mathbf{if}\ b\ \mathbf{then}\ c_1\ \mathbf{else}\ c_2] = \mathcal{C}[\mathbf{if}\ \neg b\ \mathbf{then}\ c_2\ \mathbf{else}\ c_1]$$

a tyto programy jsou opravdu ekvivalentní.