Programozáselmélet - előadás kivonat - 7. alkalom

1. Programkonstrukciók

Ebben a fejezetben azzal foglalkozunk, hogy lehet meglévő programokból új programokat készíteni. Háromféle konstrukciót engedünk meg: szekvencia, elágazás és ciklus. Ezek definícióit úgy adjuk meg, hogy a velük képzett relációk illeszkedjenek a korábban bevezetett program fogalmához. A programkonstrukciókat struktogrammal ábrázoljuk.

2. Szekvencia

Szekvencia esetében két programot egymás után végzünk el. Amennyiben az első program végrehajtása adott állapotból indulva nem véges vagy hibásan terminál, a második program nem tudja folytatni a végrehajtást a végpontból; az első program által generált végrehajtás egy lehetséges végrehajtása lesz a szekvenciának is.
Nevezzük csatlakozási pontnak azt az állapotot egy végrehajtási sorozatban, ahol az egyik program terminál és a másik program ugyaninnen indul el. Nem szeretnénk ha például az $x := x + 1$ és $x := x + 2$ programok szekvenciája (ahol x egész típusú) az állapottér $\{x{:}5\}$ eleméhez az $< \{x{:}5\}, \{x{:}6\}, \{x{:}6\}, \{x{:}8\} >$ sorozatot rendelné. Ezért az egymás után véges sokszor ismétlődő csatlakozási pontokból egyet hagyjunk el.

Jelölés: *Véges hosszú α sorozat utolsó elemét jelölje $\tau(\alpha)$. Tehát ha α véges, $\tau(\alpha) = \alpha_{|\alpha|}$.*

Jelölés: *Legyen A tetszőleges állapottér. $\alpha \in \bar{A}^*$ és $\beta \in (\bar{A} \cup \{fail\})^{**}$ úgy hogy α és β nem üres sorozatok továbbá $\tau(\alpha) = \beta_1$. Ekkor $\alpha \otimes \beta$ jelölje az α és β sorozatok összefűzésében β első elemének elhagyásával kapott sorozatot.*
Általánosítsuk a jelölést $n \in \mathbb{N}^+$ darab vagy akár végtelen sok sorozat esetére. A sorozatok összefűzése (konkatenációja) után az egymás után véges sokszor ismétlődő csatlakozási pontokból egyet hagyjunk el.

Példa: *Legyen $A = \{1, 2, 3, 4\}$. Ekkor*

$$\otimes_4(<1,2,3,1>,<1,2,3,1>,<1,2,3,1>,<1,2,3,1>) =<1,2,3,1,2,3,1,2,3,1,2,3,1>$$

$$\otimes_4(<1>,<1>,<1>,<1,2,3,1>) =<1,1,1,2,3,1>$$

$$\otimes_\infty(<1>,<1>,<1>,<1>,\ldots) =<1,1,\ldots>$$

$$\otimes_\infty(<1,2,3,1>,<1,2,3,1>,<1,4>,<4>,<4>,<4>,\ldots) =<1,2,3,1,2,3,1,4,4,4,\ldots>$$

Definíció: *Legyen A közös alap-állapottere az S_1 és S_2 programoknak. Az $(S_1; S_2)$ relációt az S_1 és S_2 programok szekvenciájának nevezzük, ha*

$$
\begin{aligned}
(S_1; S_2)(a) = & \{\alpha \in \bar{A}^{\infty} \mid \alpha \in S_1(a)\} \cup \\
& \{\alpha \in (\bar{A} \cup \{fail\})^{*} \mid \alpha \in S_1(a) \wedge \alpha_{|\alpha|} = fail\} \cup \\
& \{\gamma \in (\bar{A} \cup \{fail\})^{**} \mid \gamma = \alpha \otimes \beta \wedge \alpha \in S_1(a) \wedge |\alpha| < \infty \wedge \alpha_{|\alpha|} \neq fail \wedge \beta \in S_2(\alpha_{|\alpha|})\}
\end{aligned}
$$

A szekvencia struktogramja:

$(S_1; S_2)$

S_1
S_2

Tétel: *Legyen A közös alap-állapottere az S_1 és S_2 programoknak. Az $(S_1; S_2)$ szekvencia program.*

Tétel: *Legyen A közös alap-állapottere az S_1 és S_2 programoknak és jelölje S az $(S_1; S_2)$ szekvenciát. Ekkor*

$$p(S) = p(S_2) \odot p(S_1)$$

3. Elágazás

Definíció: *Legyen A közös alap-állapottere az $S_1, \dots S_n$ programoknak. Legyenek továbbá $\pi_1, \dots \pi_n \in A \to \mathbb{L}$ logikai függvények. Ekkor az $IF \subseteq A \times (\bar{A} \cup \{fail\})^{**}$ relációt az S_i programokból képzett π_i feltételek által meghatározott elágazásnak nevezzük és $(\pi_1{:}S_1, \dots \pi_n{:}S_n)$-nel jelöljük, ha*

$$\forall a \in A : IF(a) = \omega_0(a) \cup \bigcup_{i=1}^{n} \omega_i(a)$$

ahol $\forall i \in [1..n]$:

$$\omega_i(a) = \begin{cases} S_i(a), & \textit{ha } a \in \mathcal{D}_{\pi_i} \wedge \pi_i(a) \\ \emptyset, & \textit{ha } a \in \mathcal{D}_{\pi_i} \wedge \neg\pi_i(a) \\ \{< a, fail >\}, & \textit{ha } a \notin \mathcal{D}_{\pi_i} \end{cases}$$

és

$$\omega_0(a) = \begin{cases} \{< a, fail >\}, & \textit{ha } \forall i \in [1..n] : (a \in \mathcal{D}_{\pi_i} \wedge \neg\pi_i(a)) \\ \emptyset, & \textit{különben} \end{cases}$$

Az elágazás struktogramja:

IF

π_1	$\cdots$	π_n
S_1	$\cdots$	S_n

Az elágazást szokás még a következő módon is leírni:

if
$\quad \pi_1 \to S_1 \;\square$
$\quad \ldots$
$\quad \pi_{n-1} \to S_{n-1} \;\square$
$\quad \pi_n \to S_n$
fi

Tétel: *Legyen A közös alap-állapottere az $S_1, \ldots S_n$ programoknak. Legyenek továbbá $\pi_1, \ldots \pi_n \in A \to \mathbb{L}$ logikai függvények. Az $IF = (\pi_1{:}S_1, \ldots \pi_n{:}S_n)$ elágazás program.*

Tétel: *Legyen A közös alap-állapottere az $S_1, \ldots S_n$ programoknak. Legyenek továbbá $\pi_1, \ldots \pi_n \in A \to \mathbb{L}$ logikai függvények. $IF = (\pi_1{:}S_1, \ldots \pi_n{:}S_n)$. Ekkor*

$$\mathcal{D}_{p(IF)} = \{a \in A \mid a \in \bigcap_{i=1}^{n} \mathcal{D}_{\pi_i} \wedge a \in \bigcup_{i=1}^{n} \lceil \pi_i \rceil \wedge \forall i \in [1..n] : a \in \lceil \pi_i \rceil \implies a \in \mathcal{D}_{p(S_i)}\}$$

és

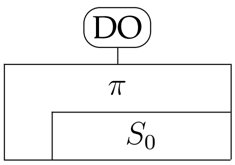
$$\forall a \in \mathcal{D}_{p(IF)} \colon p(IF)(a) = \bigcup_{i=1}^{n} p(S_i)|_{\lceil \pi_i \rceil}$$

4. Ciklus

Definíció: *Legyen $\pi \in A \to \mathbb{L}$ feltétel és $S_0 \subseteq A \times (\bar{A} \cup \{fail\})^{**}$ program. A $DO \subseteq A \times (\bar{A} \cup \{fail\})^{**}$ relációt az S_0 programból π feltétellel képzett ciklusnak nevezzük és (π, S_0)-lal jelöljük, ha $\forall a \in A$:*

$$DO(a) = \begin{cases} (S_0; DO)(a) & \textit{ha} \quad a \in \mathcal{D}_\pi \wedge \pi(a) \\ \{< a >\} & \textit{ha} \quad a \in \mathcal{D}_\pi \wedge \neg\pi(a) \\ \{< a, fail >\} & \textit{ha} \quad a \notin \mathcal{D}_\pi \end{cases}$$

A ciklus struktogramja:



A szakirodalomban a következő mód is elterjedt a ciklus leírására:

while π **do**
$\quad S_0$
od

A szekvenciához és az elágazáshoz hasonló módon is definiálhatjuk a ciklust.

Definíció: $\forall a \in A$:

$$DO(a) = \begin{cases} \{< a, fail >\}, & \textit{ha } a \notin \mathcal{D}_\pi \\ \{< a >\}, & \textit{ha } a \in \mathcal{D}_\pi \wedge \neg\pi(a) \\ \{\alpha \in (\bar{A} \cup \{fail\})^{**} \mid \exists \alpha^1, \dots, \alpha^n \in (\bar{A} \cup \{fail\})^{**} \colon \alpha = \otimes_n(\alpha_1, \dots, \alpha_n) \wedge & \\ \alpha^1 \in S_0(a) \wedge \forall i \in [1..n-1] \colon (\alpha^i \in \bar{A}^* \wedge \tau(\alpha^i) \in \lceil \pi \rceil \wedge \alpha^{i+1} \in S_0(\tau(\alpha^i))) \wedge & \\ ((\alpha^n \in \bar{A}^\infty \vee (\alpha^n \in (\bar{A} \cup \{fail\})^* \wedge \tau(\alpha^i) = fail) \vee & \\ (\alpha^n \in \bar{A}^* \wedge \tau(\alpha^i) \in \mathcal{D}_\pi \wedge \tau(\alpha^i) \notin \lceil \pi \rceil)\} & \\ \cup & \\ \{\alpha \in \bar{A}^\infty \mid \exists \alpha^1, \alpha^2, \dots \in \bar{A}^* \colon \alpha = \otimes_\infty(\alpha_1, \alpha_2, \dots) \wedge \alpha^1 \in S_0(a) \wedge \forall i \in \mathbb{N}^+ \colon & \\ (\alpha_i \in \bar{A}^* \wedge \tau(\alpha^i) \in \lceil \pi \rceil \wedge \alpha^{i+1} \in S_0(\tau(\alpha^i)))\} & \\ \cup & \\ \{\alpha \in (\bar{A} \cup \{fail\})^* \mid \exists \alpha^1, \dots, \alpha^n \in (\bar{A} \cup \{fail\})^{**} \colon \alpha = \otimes_n(\alpha_1, \dots, \alpha_n) \wedge & \\ \alpha^1 \in S_0(a) \wedge \forall i \in [1..n-2] \colon (\alpha^i \in \bar{A}^* \wedge \tau(\alpha^i) \in \lceil \pi \rceil \wedge \alpha^{i+1} \in S_0(\tau(\alpha^i))) & \\ \wedge(\alpha^{n-1} \in \bar{A}^* \wedge \tau(\alpha^{n-1}) \notin \mathcal{D}_\pi \wedge \alpha^n =< \tau(\alpha^{n-1}), fail >)\}, & \textit{ha } a \in \mathcal{D}_\pi \wedge \pi(a) \end{cases}$$

Első ránézésre a definíció kissé bonyolultnak tűnik. Amennyiben sorra vesszük hogy az állapottér egy a pontjához milyen sorozatokat rendelhet a ciklus, már nem is fogjuk bonyolultnak találni a definíciót!

- Ha a állapotban a ciklusfeltétel nem kiértékelhető, akkor a ciklus hibásan terminál.
- Ha a állapotban a ciklusfeltétel kiértékelhető de nem teljesül a-ra, akkor a ciklus semmit nem csinál, végrehajtása befejeződik az a állapotban.
- A ciklusmagot véges sokszor elvégezzük egymás után, úgy hogy a ciklusmag utolsó végrehajtásához tartozó sorozat
  - végtelen; vagy
  - véges és a $fail$ állapotban végződik; vagy
  - véges és utolsó elemében kiértékelhető a π feltétel, de az $hamis$.
- A ciklusmagot végtelen sokszor elvégezzük egymás után, mert egy végrehajtás után mindig olyan állapotba jutunk ahol a ciklusfeltétel teljesül.
- Az utolsó lehetőség az, hogy a ciklusmagot azért véges sokszor (de legalább egyszer) végezzük el egymás után, mert utoljára egy olyan állapotba jutunk ahol a ciklusfeltétel nem kiértékelhető.