Fundamentele limbajelor de programare

Mulțimi definite de reguli[1]

Traian Florin Șerbănuță și Andrei Sipoș

Facultatea de Matematică și Informatică, DL Info

Anul II, Semestrul II, 2024/2025

[1] Inspirat de prezentarea dl. prof. V.E. Căzănescu

Motivație

În descrierea formală a diverse concepte informatice, întâlnim deseori definiții (recursive) prin reguli.

Exemplu: descrierea sintaxei unui limbaj

```
AExp ::= Int | AExp + AExp | AExp - AExp | AExp * AExp
```

AExp e o submulțime a limbajului ce poate fi obținut din întregi și simbolurile '+', '-', '*'.

Exemplu: Definirea unei relații

De exemplu, regulile pentru definirea deducției în logica propozițională:

$$\frac{\vdash p \qquad \vdash p \to q}{\vdash q}$$

$\vdash p \to (q \to p)$

$\vdash (p \to (q \to r)) \to ((p \to q) \to (p \to r))$

$\vdash (\neg p \to \neg q) \to (q \to p)$

Exemplu: Definirea relației de evaluare a expresiilor aritmetice

$$\frac{}{\langle i, \sigma \rangle \Downarrow \langle i \rangle}$$

$$\frac{\langle a_1, \sigma \rangle \Downarrow \langle i_1 \rangle \qquad \langle a_2, \sigma \rangle \Downarrow \langle i_2 \rangle}{\langle a_1 + a_2, \sigma \rangle \Downarrow \langle i \rangle} \text{ dacă } i = i_1 + i_2$$

$$\frac{\langle a_1, \sigma \rangle \Downarrow \langle i_1 \rangle \qquad \langle a_2, \sigma \rangle \Downarrow \langle i_2 \rangle}{\langle a_1 - a_2, \sigma \rangle \Downarrow \langle i \rangle} \text{ dacă } i = i_1 - i_2$$

$$\frac{\langle a_1, \sigma \rangle \Downarrow \langle i_1 \rangle \qquad \langle a_2, \sigma \rangle \Downarrow \langle i_2 \rangle}{\langle a_1 * a_2, \sigma \rangle \Downarrow \langle i \rangle} \text{ dacă } i = i_1 * i_2$$

Plan

- Ce au în comun aceste definiții? Ce definesc ele, mai exact?
- Inducție pe reguli de definire / Inducție structurală
- Neambiguitate și recursie pe reguli de definire
- Aplicații la semanticile big-step și small-step

Secțiunea 1

Definiții (recursive) bazate pe reguli

Mulțimi Moore

Definiție

Fie X o mulțime. Numim o mulțime $A \subseteq \mathcal{P}(X)$ **mulțime Moore** pe X dacă $X \in A$, iar, pentru orice $B \subseteq A$ nevidă, avem $\bigcap B \in A$.

Teoremă

Fie X o mulțime și A o mulțime Moore pe X. Atunci există și este unic $C \in A$, numit **minimul** lui A, astfel încât pentru orice $D \in A$, avem $C \subseteq D$.

Demonstrație

Unicitatea rezultă imediat: dacă avem C_1, $C_2 \in A$ cu acea proprietate, atunci $C_1 \subseteq C_2$ și $C_2 \subseteq C_1$, deci $C_1 = C_2$. Pentru existență, cum $X \in A$, avem $A \neq \emptyset$, deci $\bigcap A \in A$. Luăm $C := \bigcap A$ și verificăm că are proprietatea căutată.

Reguli de definire. Proprietatea de închidere

Definiție

Fie o mulțime A fixată. Numim **regulă de definire** peste mulțimea A, o pereche $(H, a) \in A^* \times A$ (A^* este monoidul cuvintelor peste A).
Dată fiind o regulă de definire $r = (a_1 a_2 \ldots a_n, a)$, definim:

ipotezele lui r ca fiind $hyp(r) = \{a_1, a_2, \ldots a_n\}$

concluzia lui r ca fiind $conc(r) = a$

O mulțime de reguli de definire se numește **sistem**.

Proprietatea de închidere

Fie B o submulțime a lui A.
B este închisă la o regulă de definire r dacă $hyp(r) \subseteq B$ implică $conc(r) \in B$.
B este inchisă la sistemul de reguli de definire R dacă este închisă la toate regulile din R

Submulțimile închise sunt o mulțime Moore

Teoremă

Fie R un sistem de reguli de definire peste o mulțime A. Mulțimea submulțimilor lui A închise la R este o mulțime Moore.

Demonstrație

1. Fie $\mathcal{N}$ o mulțime de submulțimi ale lui A închise la R arbitrar aleasă
2. ; Fie r o regulă din R arbitrar aleasă
3. ; ; Presupunem că $hyp(r) \subseteq \bigcap \mathcal{N}$
4. ; ; ; Fie $N \in \mathcal{N}$ arbitrar aleasă
5. ; ; ; ; Avem $hyp(r) \subseteq N$ (din 3 și 4)
6. ; ; ; ; Avem $conc(r) \in N$ (din 1, 2 și 5)
7. ; ; ; Avem $conc(r) \in \mathcal{N}$ (din 4 –arbitrar aleasă– și 6)
8. ; ; Avem $\mathcal{N}$ închisă la r (din 3 am dedus 7)
9. ; Avem $\mathcal{N}$ închisă la R (din 2 –arbitrar aleasă– și 8)

Mulțimea definită de un sistem de reguli

Definiție

Mulțimea definită de sistemul de reguli R peste mulțimea suport A este cea mai mică submulțime a lui A închisă la R, adică intersecția tuturor submulțimilor lui A închise la R.

Teoremă (inducție)

Fie M mulțimea definită de sistemul R și fie $N \subseteq M$. Dacă N închisă la R, atunci $N = M$.

Demonstrație (trivială)

M este cea mai mică mulțime închisă la R, deci $M \subseteq N$. Dar din ipoteză $N \subseteq M$. Deci $N = M$.

Inducție deductivă (pe reguli)

Teoremă (principiul inducției deductive)

Fie M mulțimea definită de sistemul R și fie P o proprietate peste A. Dacă pentru orice regulă $r \in R$, putem deduce $P(conc(r))$ presupunând că $P(h)$ pentru orice $h \in hyp(r)$, atunci P este adevărată pentru orice element din M.

Demonstrație

1. Fie $N = \{m \in M \mid P(m)\}$
2. Fie $r \in R$ arbitrar aleasă
3. ; Presupun $hyp(r) \subseteq N$
4. ; ; Avem $P(h)$ pentru orice $h \in hyp(r)$ (din 3 și 1)
5. ; ; Avem $P(conc(r))$ din ipoteza teoremei
6. ; Avem N închisă la r (din 3 am dedus 5)
7. Avem N închisă la R (din 2 –arbitrar aleasă– și 6)
8. Avem $N = M$ (din 7 și teorema de mai sus)

Exemplu: Accesibilitate / Definiție alternativă

Teoremă

Fie M mulțimea definită de sistemul de reguli R. Definim lanțul crescător $(M_n)_{n\in\mathbb{N}}$ astfel: $M_0 = \emptyset$; $\quad M_{n+1} = M_n \cup \{conc(r) \mid r \in R, hyp(r) \subseteq M_n\}$.
Atunci $M = \bigcup_{n\in\mathbb{N}} M_n$

Demonstrație (prin inducție deductivă)

$\bigcup_{n\in\mathbb{N}} M_n \subseteq M$: demonstrez că pentru orice $n \in \mathbb{N}$, $M_n \subseteq M$ prin inducție după n. Cazul de bază $M_0 \subseteq M$ e trivial
Pas de inducție. Presupun $M_n \subseteq M$. Fie $a \in M_{n+1}$ arbitrar.

- dacă $a \in M_n$, gata
- dacă $a = conc(r)$ unde $r \in R$ și $hyp(r) \subseteq M_n$, atunci $hyp(r) \subseteq M$, deci $a \in M$ (m închisă).

$M \subseteq \bigcup_{n\in\mathbb{N}} M_n$: inducție pe regulile R.
Fie $r = (h_1 h_2 \cdots h_k, a)$ astfel încât pentru orice $i \in \{1, \ldots, k\}$, $h_i \in \bigcup_{n\in\mathbb{N}} M_n$. Pentru orice i, fie n_i astfel încât $h_i \in M_{n_i}$.
Fie $n = \max\{n_1, \ldots, n_k\}$. Atunci $hyp(r) \subseteq M_n$, deci $a \in M_{n+1}$.

Ambiguitate / neambiguitate

O definiție este ambiguă dacă pot obține o concluzie în mai multe feluri

Exemplu

Pentru sintaxa
`AExp ::= Int | AExp + AExp | AExp - AExp | AExp * AExp`
expresia `3 + 5 * 7` este ambiguă

Definiție (Neambiguitate / citire unică)

Sistemul de reguli R are proprietatea de **neambiguitate** dacă, notând cu M mulțimea definită de R, pentru orice $m \in M$, există o singură regulă $r \in R$ astfel încât $conc(r) = m$ și $hyp(r) \subseteq M$.

Exemplu

`AExp ::= Int | (AExp + AExp) | (AExp - AExp) | (AExp * AExp)`

Definiție recursivă (pe reguli)

Teoremă (metateorema definițiilor recursive)

Fie M mulțimea definită de sistemul de reguli R cu proprietatea de neambiguitate. Fie o mulțime B, și pentru orice regulă $r = (a_1 a_2 \ldots a_n, a) \in R$, fie o funcție $g_r : B^n \to B$. Atunci există o unică funcție $f : M \to B$ cu proprietatea că pentru orice regulă $r = (a_1 a_2 \ldots a_n, a) \in R$ astfel încât $hyp(r) \subseteq M$,
$f(a) = g_r(f(a_1), f(a_2), \ldots, f(a_n))$.

Demonstrație

Fie $G_f = \bigcup_{n \in \mathbb{N}} G_n$, unde $(G_n)_{n \in \mathbb{N}}$ este definită recursiv prin:

- $G_0 = \emptyset$
- $G_{n+1} = G_n \cup \{(a, g_r(b_1, b_2, \ldots, b_k)) \mid r = (a_1 a_2 \ldots a_k, a) \in R \text{ și } (a_i, b_i) \in G_n, i \in \{1, \ldots, k\}\}$

Atunci G_f este graficul unei funcții $f : A \to B$ cu proprietatea din enunț.

Metateorema definițiilor recursive

Demonstrație (cont.)

- Demonstrăm că G_f e totală, adică pentru orice $a \in A$ există $b \in B$ a.î. $(a, b) \in G_f$ prin inducție deductivă.
  fie $r = (h_1 h_2 \ldots h_k, a) \in R$ astfel încât pentru orice $i \in \{1, \ldots, k\}$, există $b_i \in B$ a.î. $(h_i, b_i) \in G_f$.
  Pentru orice $i \in \{1, \ldots, k\}$ există un n_i astfel încât $(h_i, b_i) \in G_{n_i}$. Fie $m = max\{n_1, \ldots n_k\}$
  Atunci $(a, g_r(b_1, \ldots, b_k)) \in G_{m+1} \subseteq G_f$
- Demonstrăm că G_f e funcțională
  Fie $(a, b), (a, b') \in G_f$. Atunci:
  - există $r = (h_1 \cdots h_k, a), r' = (h'_1 \cdots h'_{k'}, a) \in R$
  - există b_i a.î. $(h_i, b_i) \in G_f$ pentru orice $i \in \{1, \ldots, k\}$
  - există b'_j a.î. $(h'_j, b'_j) \in G_f$ pentru orice $j \in \{1, \ldots, k'\}$
  - $b = g_r(b_1, \ldots, b_k)$ și $b' = g_{r'}(b'_1, \ldots, b'_{k'})$

  Deoarece $\mathrm{Dom}\, G_f = A$, avem $hyp(r) \subseteq A$ și $hyp(r') \subseteq A$.
  Din proprietatea de neambiguitate, trebuie ca $r = r'$, deci $b = b'$.

Secțiunea 2

Exemple: Aplicații la big-step și small-step

Semantica big-step este deterministă

- Pentru orice a expresie aritmetică și σ stare există un unic întreg i astfel încât $\langle a, \sigma\rangle \Downarrow \langle i\rangle$

  Demonstrație: prin inducție pe structura expresiilor aritmetice.

- Pentru orice b expresie booleană și σ stare există o unică valoare de adevăr t astfel încât $\langle b, \sigma\rangle \Downarrow \langle t\rangle$

  Demonstrație: prin inducție pe structura expresiilor booleene.

  Folosind proprietatea de mai sus.

- Pentru orice s instrucțiune și σ stare există cel mult o stare σ' astfel încât $\langle s, \sigma\rangle \Downarrow \langle \sigma'\rangle$

  Demonstrație: prin inducție pe structura instrucțiunilor.

  Folosind proprietățile de mai sus.

Echivalența între programe

Pe instrucțiuni, putem defini relația de echivalență $\sim$ în felul următor: pentru orice s_1, s_2, avem $s_1 \sim s_2$ exact atunci când, pentru orice σ, $\sigma' \in \Sigma$, $\langle s_1, \sigma \rangle \Downarrow \langle \sigma' \rangle$ dacă și numai dacă $\langle s_2, \sigma \rangle \Downarrow \langle \sigma' \rangle$.

Propoziție

Fie b o expresie booleană și s o instrucțiune. Notăm $w := \texttt{while}\, b\, \texttt{do}\, s$. Atunci $w \sim (\texttt{if}\, b\, \texttt{then}\, (s; w) \texttt{else skip})$.

Demonstrație

Fie σ, $\sigma' \in \Sigma$. Vrem să arătăm că

$$\langle w, \sigma \rangle \Downarrow \langle \sigma' \rangle \Leftrightarrow \langle \texttt{if}\, b\, \texttt{then}\, (s; w) \texttt{else skip}, \sigma \rangle \Downarrow \langle \sigma' \rangle.$$

Demonstrăm implicația "$\Rightarrow$". Presupunem că avem $\langle w, \sigma \rangle \Downarrow \langle \sigma' \rangle$. În primul caz, avem $\langle b, \sigma \rangle \Downarrow \langle \mathit{false} \rangle$ și $\sigma = \sigma'$, iar, folosind faptul că $\langle \texttt{skip}, \sigma \rangle \Downarrow \langle \sigma \rangle$, deducem că $\langle \texttt{if}\, b\, \texttt{then}\, (s; w) \texttt{else skip}, \sigma \rangle \Downarrow \langle \sigma \rangle$.

Echivalența între programe

Demonstrație (cont.)

În al doilea caz, avem că $\langle b, \sigma \rangle \Downarrow \langle true \rangle$ și există σ'' cu $\langle s, \sigma \rangle \Downarrow \langle \sigma'' \rangle$ și $\langle w, \sigma'' \rangle \Downarrow \langle \sigma' \rangle$. Putem deduce că $\langle s; w, \sigma \rangle \Downarrow \langle \sigma' \rangle$ și apoi că $\langle \texttt{if}\ b\ \texttt{then}\ (s; w) \texttt{else skip}, \sigma \rangle \Downarrow \langle \sigma' \rangle$.

Demonstrăm acum implicația "$\Leftarrow$". Presupunem că avem $\langle \texttt{if}\ b\ \texttt{then}\ (s; w) \texttt{else skip}, \sigma \rangle \Downarrow \langle \sigma' \rangle$.

În primul caz, avem $\langle b, \sigma \rangle \Downarrow \langle false \rangle$ și $\langle \texttt{skip}, \sigma \rangle \Downarrow \langle \sigma' \rangle$, de unde deducem $\sigma = \sigma'$, apoi imediat $\langle w, \sigma \rangle \Downarrow \langle \sigma \rangle$.

În al doilea caz, avem $\langle b, \sigma \rangle \Downarrow \langle true \rangle$ și $\langle s; w, \sigma \rangle \Downarrow \langle \sigma' \rangle$, de unde scoatem că există σ'' cu $\langle s, \sigma \rangle \Downarrow \langle \sigma'' \rangle$ și $\langle w, \sigma'' \rangle \Downarrow \langle \sigma' \rangle$. Putem apoi deduce imediat că $\langle w, \sigma \rangle \Downarrow \langle \sigma' \rangle$.

Semantica small-step este simulată de cea big-step

Expresii aritmetice

Dacă $\langle a, \sigma\rangle \to \langle a', \sigma\rangle$, și $\langle a', \sigma\rangle \Downarrow \langle i\rangle$, atunci $\langle a, \sigma\rangle \Downarrow \langle i\rangle$.

Demonstrație

Inducție deductivă pe regulile de definire a relației într-un pas pentru expresii aritmetice. Tratăm doar câteva cazuri.

- $(\textsc{Id})\ \langle x, \sigma\rangle \to \langle i, \sigma\rangle$ *dacă* $i = \sigma(x)$
  $\langle i, \sigma\rangle \Downarrow \langle i\rangle$ și $\langle x, \sigma\rangle \Downarrow \langle i\rangle$
- $\dfrac{\langle a_1, \sigma\rangle \to \langle a_1', \sigma\rangle}{\langle a_1 + a_2, \sigma\rangle \to \langle a_1' + a_2, \sigma\rangle}$
  Dacă $\langle a_1' + a_2, \sigma\rangle \Downarrow \langle i\rangle$, trebuie ca $\langle a_1', \sigma\rangle \Downarrow \langle i_1\rangle$, $\langle a_2, \sigma\rangle \Downarrow \langle i_2\rangle$ și $i = i_1 + i_2$.
  Din ipoteza de inducție, $\langle a_1, \sigma\rangle \Downarrow \langle i_1\rangle$, deci $\langle a_1 + a_2, \sigma\rangle \Downarrow \langle i\rangle$
- $(\textsc{Add})\ \langle i_1 + i_2, \sigma\rangle \to \langle i, \sigma\rangle$ *dacă* $i = i_1 + i_2$
  Avem $\langle i, \sigma\rangle \Downarrow \langle i\rangle$ și $\langle i_1, \sigma\rangle \Downarrow \langle i_1\rangle$, $\langle i_2, \sigma\rangle \Downarrow \langle i_2\rangle$, de unde $\langle i_1 + i_2, \sigma\rangle \Downarrow \langle i\rangle$

Semantica small-step este simulată de cea big-step

Expresii booleene

Dacă $\langle b, \sigma \rangle \to \langle b', \sigma \rangle$, și $\langle b', \sigma \rangle \Downarrow \langle t \rangle$, atunci $\langle b, \sigma \rangle \Downarrow \langle t \rangle$.

Demonstrație

Asemănător ca pentru expresii aritmetice, folosind și rezultatul deja demonstrat.

Semantica small-step este simulată de cea big-step

Instrucțiuni

Dacă $\langle s, \sigma\rangle \to \langle s', \sigma'\rangle$, și $\langle s', \sigma'\rangle \Downarrow \langle \sigma''\rangle$, atunci $\langle s, \sigma\rangle \Downarrow \langle \sigma''\rangle$.

Demonstrație

Inducție deductivă pe regulile de definire a relației într-un pas pentru instrucțiuni.

- $\dfrac{\langle s_1, \sigma\rangle \to \langle s_1', \sigma'\rangle}{\langle s_1 \ ; \ s_2, \sigma\rangle \to \langle s_1' \ ; \ s_2, \sigma'\rangle}$
  Dacă $\langle s_1' \ ; \ s_2, \sigma'\rangle \Downarrow \langle \sigma''\rangle$, trebuie ca $\langle s_1', \sigma'\rangle \Downarrow \langle \sigma_1\rangle$, $\langle s_2, \sigma_1\rangle \Downarrow \langle \sigma''\rangle$.
  Din ipoteza de inducție, $\langle s_1, \sigma\rangle \Downarrow \langle \sigma_1\rangle$, deci $\langle s_1 \ ; \ s_2, \sigma\rangle \Downarrow \langle \sigma''\rangle$
- (Seq) $\langle \texttt{skip}; \, s_2, \sigma\rangle \to \langle s_2, \sigma\rangle$ direct
- (While) $\langle \texttt{while}\ b\ \texttt{do}\ bl, \sigma\rangle \to$
  $\langle \texttt{if}\ b\ \texttt{then(}\ bl\ ;\ \texttt{while}\ b\ \texttt{do}\ bl\ \texttt{)else skip}, \sigma\rangle$
  Aplicăm proprietatea de echivalență demonstrată anterior.