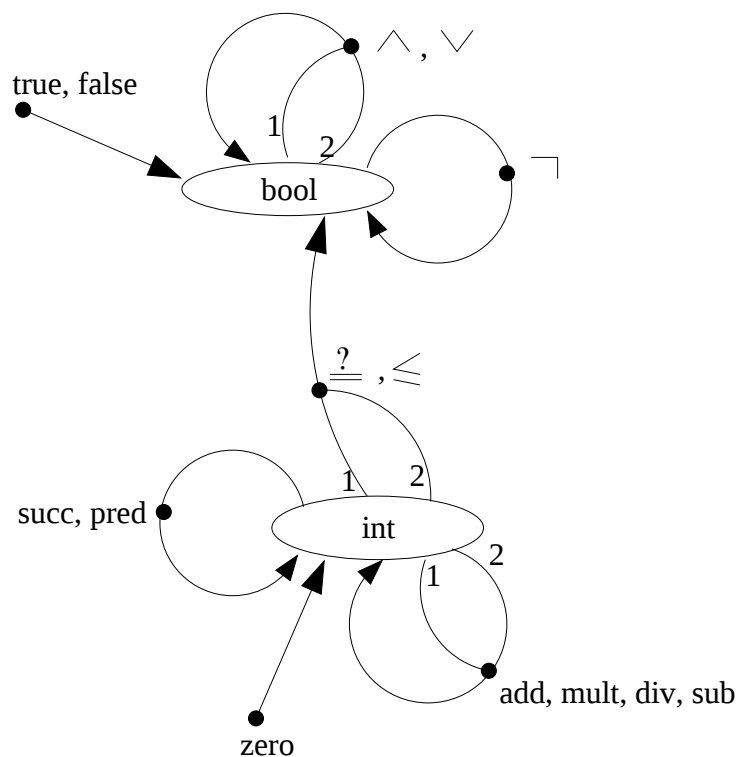


Übungen zu Einführung in die Informatik I

Aufgabe 17 Die Signatur der Rechenstruktur der ganzen Zahlen (Lösungsvorschlag)

Der Signaturgraph der ganzen Zahlen mit den Sorten $S = \{\mathbf{bool}, \mathbf{int}\}$ und den Funktionssymbolen $F = \{\text{true}, \text{false}, \wedge, \vee, \neg, \text{zero}, \text{succ}, \text{pred}, \text{add}, \text{mult}, \text{sub}, \text{div}, \overset{?}{=}, \leq\}$ hat die Form:



Durch Umbenennung der Sorte **int** in **nat** ergibt sich der Signaturgraph der Rechenstruktur der natürlichen Zahlen.

Um zu zeigen, dass die Spezifikationen der beiden Rechenstrukturen unterschiedlich sind, betrachten wir die Spezifikation des Funktionssymbols **pred**: Für die ganzen Zahlen gilt

$$\text{pred}^{\mathbf{INT}}(z) = z - 1$$

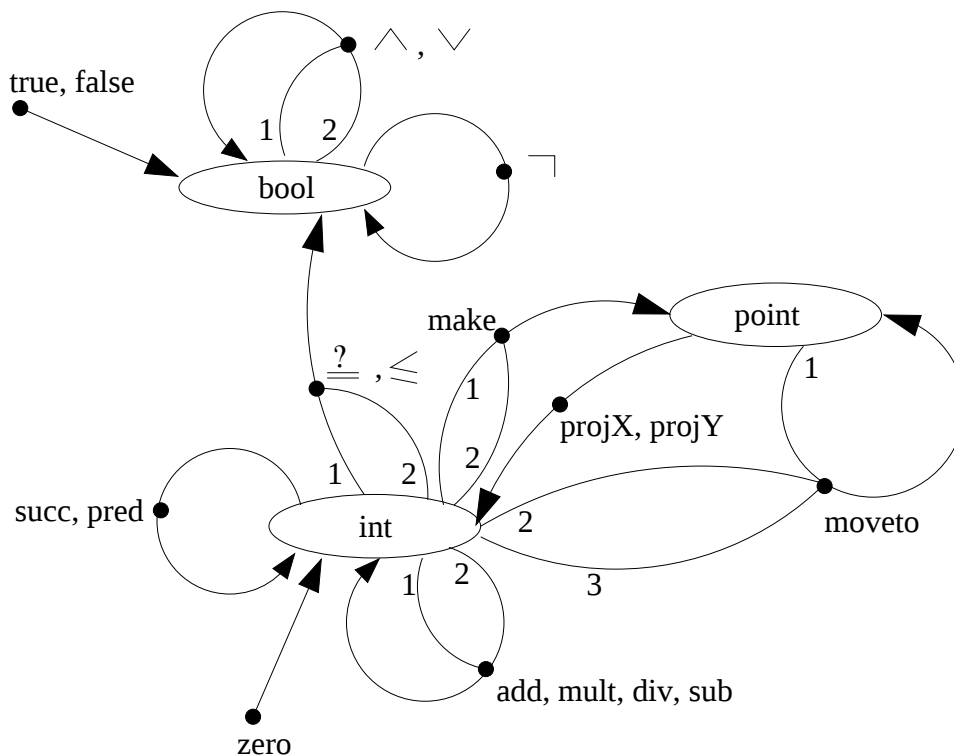
wohingegen bei der Rechenstruktur **NAT** diese Funktion durch

$$\text{pred}^{\mathbf{NAT}}(x) = \begin{cases} x - 1 & x \geq 1 \\ \perp & x = 0 \end{cases}$$

gegeben ist.

Aufgabe 18 Kartesisches Koordinatensystem (Lösungsvorschlag)

a) Der Signaturgraph der Rechenstruktur **K** besitzt die Form:



- b) (i) syntaktisch korrekt
 (ii) syntaktisch inkorrekt
 (iii) syntaktisch inkorrekt

Aufgabe 19 Termersetzungsregeln und partielle Korrektheit (Lösungsvorschlag)

a) Reduktionsregeln für die Substraktion:

Wir definieren für das Funktionssymbol `sub` mit der Funktionalität

fct `sub` = (**nat**, **nat**) **nat**

die Termersetzungsregeln

$$\text{sub}(x, \text{succ}(y)) \rightarrow \text{pred}(\text{sub}(x, y)) \quad (1)$$

$$\text{sub}(x, \text{zero}) \rightarrow x \quad (2)$$

b) Zu Vereinfachung der Terme benötigen wir die aus der Vorlesung bekannte Reduktionsregel:

$$\text{pred}(\text{succ}(x)) \rightarrow x \quad (3)$$

$$\text{pred}(\text{sub}(\text{succ}(\text{succ}(\text{succ}(\text{zero}))), \text{succ}(\text{succ}(\text{zero}))))$$

$$\xrightarrow{1} \text{pred}(\text{pred}(\text{sub}(\text{succ}(\text{succ}(\text{succ}(\text{zero}))), \text{succ}(\text{zero}))))$$

$$\xrightarrow{1} \text{pred}(\text{pred}(\text{pred}(\text{sub}(\text{succ}(\text{succ}(\text{succ}(\text{zero}))), \text{zero}))))$$

$$\xrightarrow{2} \text{pred}(\text{pred}(\text{pred}(\text{succ}(\text{succ}(\text{succ}(\text{zero}))))))$$

$$\xrightarrow{3} \text{pred}(\text{pred}(\text{succ}(\text{succ}(\text{zero}))))$$

$$\begin{aligned}
& \xrightarrow{3} \text{pred}(\text{succ}(\text{zero})) \\
& \xrightarrow{3} \text{zero} \\
& \text{sub}(\text{succ}(\text{zero}), \text{succ}(\text{succ}(\text{zero}))) \\
& \xrightarrow{1} \text{pred}(\text{sub}(\text{succ}(\text{zero}), \text{succ}(\text{zero}))) \\
& \xrightarrow{1} \text{pred}(\text{pred}(\text{sub}(\text{succ}(\text{zero}), \text{zero}))) \\
& \xrightarrow{2} \text{pred}(\text{pred}(\text{succ}(\text{zero}))) \\
& \xrightarrow{3} \text{pred}(\text{zero}).
\end{aligned}$$

- c) Wiederholung aus der Vorlesung: Eine Termersetzungsregel $t \rightarrow r$ heißt *partiell korrekt* bezüglich der Rechenstruktur A , falls für jede Belegung β in A gilt: $I_\beta^A[t] = I_\beta^A[r]$

Wir müssen also für alle Belegungen β zeigen:

- (i) $I_\beta^{\text{NAT}}[\text{sub}(x, \text{succ}(y))] = I_\beta^{\text{NAT}}[\text{pred}(\text{sub}(x, y))]$ und
- (ii) $I_\beta^{\text{NAT}}[\text{sub}(x, \text{zero})] = I_\beta^{\text{NAT}}[x]$.

zu (i): Um die partielle Korrektheit der Regel $\text{sub}(x, \text{succ}(y)) \rightarrow \text{pred}(\text{sub}(x, y))$ zu zeigen, unterscheiden wir drei Fälle:

1. Fall: β ist eine Belegung mit $\beta(x) = \perp$ oder $\beta(y) = \perp$. Dann gilt wegen der Striktheit von sub , succ und pred :

$$I_\beta^{\text{NAT}}[\text{sub}(x, \text{succ}(y))] = \perp = I_\beta^{\text{NAT}}[\text{pred}(\text{sub}(x, y))]$$

2. Fall: β ist eine Belegung mit $\beta(x) > \beta(y)$. Dann folgt $I_\beta^{\text{NAT}}[x] \geq I_\beta^{\text{NAT}}[\text{succ}(y)]$ und es gilt aufgrund der Spezifikation von sub :

$$\begin{aligned}
I_\beta^{\text{NAT}}[\text{sub}(x, \text{succ}(y))] &= I_\beta^{\text{NAT}}[x] - I_\beta^{\text{NAT}}[\text{succ}(y)] \\
&= I_\beta^{\text{NAT}}[x] - (I_\beta^{\text{NAT}}[y] + 1) \\
&= (I_\beta^{\text{NAT}}[x] - I_\beta^{\text{NAT}}[y]) - 1 \\
&= I_\beta^{\text{NAT}}[\text{sub}(x, y)] - 1 \\
&= I_\beta^{\text{NAT}}[\text{pred}(\text{sub}(x, y))]
\end{aligned}$$

3. Fall: β ist eine Belegung mit $\beta(x) \leq \beta(y)$. Dann folgt $I_\beta^{\text{NAT}}[x] < I_\beta^{\text{NAT}}[\text{succ}(y)]$ und es gilt aufgrund der Spezifikation von sub :

$$I_\beta^{\text{NAT}}[\text{sub}(x, \text{succ}(y))] = \perp$$

Ferner gilt wegen $\beta(x) \leq \beta(y)$: $I_\beta^{\text{NAT}}[\text{sub}(x, y)] = 0$ bzw. $I_\beta^{\text{NAT}}[\text{sub}(x, y)] = \perp$. Also gilt aufgrund der Spezifikation bzw. Striktheit von pred :

$$I_\beta^{\text{NAT}}[\text{pred}(\text{sub}(x, y))] = \perp$$

zu (ii): Ist β eine Belegung mit $\beta(x) = \perp$, dann folgt aus der Striktheit von sub : $I_\beta^{\text{NAT}}[\text{sub}(x, \text{zero})] = \perp = I_\beta^{\text{NAT}}[x]$.

Für alle Belegungen β mit $\beta(x) \neq \perp$ gilt $I_\beta^{\text{NAT}}[\text{sub}(x, \text{zero})] = I_\beta^{\text{NAT}}[x] - 0 = I_\beta^{\text{NAT}}[x]$.

Aufgabe 20 Termersetzung: Sequenzen (Lösungsvorschlag)

Der Einfachheit halber betrachten wir nur Grundterme, die aus `empty`, `make` und `conc` aufgebaut sind.

Betrachten wir den Term `kompl(conc(x0, x1), conc(y0, y1))`. Die Sequenzen `x0` und `x1` bilden eine beliebige Zerlegung des ersten Arguments, `y0` und `y1` eine beliebige Zerlegung des zweiten. Solche beliebige Zerlegungen reichen aber nicht, um den Funktionswert von `kompl` anzugeben.

Wir können den Wert von `kompl` jedoch angeben, wenn wir die Argumente in feste Bestandteile zerlegen, etwa jeweils in das erste Element und die Restsequenz.

Zu diesem Zweck definieren wir eine Normalform für Konstruktortermine. Ein Konstruktorterm ist in Normalform, wenn er von der Gestalt

`empty`, `make(a)`

oder

`conc(make(a), s)`

ist, wobei `s` wieder in Normalform ist und `s ≠ empty`.

Zuerst geben wir ein Termersetzungssystem an, das Konstruktortermine auf Normalform reduziert. Dies erfolgt durch Eliminieren von unnötigen `empty` Termen und durch Anwendung der Assoziativität von `conc`. Damit erhalten wir:

1. `conc(empty, y) → y`
2. `conc(x, empty) → x`
3. `conc(conc(x, y), z) → conc(x, conc(y, z))`

Jetzt brauchen wir die Regeln für `kompl` nur noch für Normalformen anzugeben:

4. `kompl(empty, y) → y  $\stackrel{?}{=}$  empty`
5. `kompl(make(a), y) → y  $\stackrel{?}{=}$  make(¬ a)`
6. `kompl(conc(make(a), x), empty) → false`
7. `kompl(conc(make(a), x), make(b)) → a  $\stackrel{?}{=}$  ¬b ∧ x  $\stackrel{?}{=}$  empty`
8. `kompl(conc(make(a), x), conc(make(b), y)) → a  $\stackrel{?}{=}$  ¬ b ∧ kompl(x, y)`

Für die Vereinfachung von Booleschen Termen mit der Vergleichsoperation $\stackrel{?}{=}$ auf Sequenzen bzw. auf Booleschen Termen werden Vereinfachungsregeln vorausgesetzt.

Aufgabe 21 Markov — geklammerte Boolesche Terme (Lösungsvorschlag)

- a) Zeichenvorrat $V = \{true, false, x, y, z, \wedge, \vee, \neg, (,), t, \otimes\}$

Mit dem Zeichen t werden bereits erkannte, Boolesche Terme gekennzeichnet. Das Zeichen $\otimes$ dient der Anzeige eines besonderen Fehlers.

- b) Regelmeng $T = \{$
- | | | | |
|----------------|---------------|--|------|
| x | $\rightarrow$ | t , | (1) |
| y | $\rightarrow$ | t , | (2) |
| z | $\rightarrow$ | t , | (3) |
| $true$ | $\rightarrow$ | t , | (4) |
| $false$ | $\rightarrow$ | t , | (5) |
| $(\neg t)$ | $\rightarrow$ | t , | (6) |
| $(t \wedge t)$ | $\rightarrow$ | t , | (7) |
| $(t \vee t)$ | $\rightarrow$ | t , | (8) |
| tt | $\rightarrow$ | $\cdot \otimes$, (Fehlerfall zwei Terme, kein Operator) | (9) |
| t | $\rightarrow$ | $\varepsilon \}$ | (10) |

- c) (i)
$$\begin{array}{llll} (((x \wedge y) \vee (\neg y)) \wedge true) & \xrightarrow{1} & (((t \wedge y) \vee (\neg y)) \wedge true) & \xrightarrow{2} \\ (((t \wedge t) \vee (\neg y)) \wedge true) & \xrightarrow{2} & (((t \wedge t) \vee (\neg t)) \wedge true) & \xrightarrow{4} \\ (((t \wedge t) \vee (\neg t)) \wedge t) & \xrightarrow{6} & (((t \wedge t) \vee t) \wedge t) & \xrightarrow{7} \\ ((t \vee t) \wedge t) & \xrightarrow{8} & (t \wedge t) & \xrightarrow{7} \\ t & \xrightarrow{10} & \varepsilon & \end{array}$$

$$(ii) \quad (z \vee (yz)) \xrightarrow{2} (z \vee (tz)) \xrightarrow{3} (t \vee (tz)) \xrightarrow{3} (t \vee (tt)) \xrightarrow{9} (t \vee (\otimes))$$

d) Lösung analog zu Aufgabe 16, Blatt 4