Tentamen Functioneel Programmeren (2IA05) met uitwerkingen

0. De operatoren $\lceil$ ("take") en $\lfloor$ ("drop") hebben de volgende eigenschappen, voor alle $b \in B$, $s \in \mathcal{L}_*(B)$ en $n \in \mathsf{Nat}$ –voor alle datatypen B –:

$$
\begin{array}{lcllcl}
[\,]\lceil n & = & [\,] & [\,]\lfloor n & = & [\,] \\
s\lceil 0 & = & [\,] & s\lfloor 0 & = & s \\
(b \triangleright s)\lceil (n{+}1) & = & b \triangleright s\lceil n & (b \triangleright s)\lfloor (n{+}1) & = & s\lfloor n
\end{array}
$$

Bewijs dat $s\lceil(n{+}m)\lfloor n \;=\; s\lfloor n\lceil m$, voor alle $s \in \mathcal{L}_*(B)$ en $n, m \in \mathsf{Nat}$. [1 punt]

Oplossing: Een bewijs door middel van Volledige Inductie lijkt hier aangewezen, maar omdat er *twee* gebonden natuurlijke variabelen in het spel zijn moeten we kiezen: inductie over m of over n. Gelet op de voorkomens van m en n in de formule lijkt inductie over n het meest kansrijk. Dat wil zeggen, we bewijzen: $(\forall n :: (\forall m, s :: s\lceil(n{+}m)\lfloor n \;=\; s\lfloor n\lceil m\,)\,)$ door middel van Volledige Inductie over n.

basis: Voor alle $m \in \mathsf{Nat}$ en $s \in \mathcal{L}_*(B)$:

$$
\begin{array}{ll}
 & s\lceil(0{+}m)\lfloor 0 \\
= & \quad \{ \text{ definitie } \lfloor\,;\ 0{+}m = m \ \} \\
 & s\lceil m \\
= & \quad \{ \text{ definitie } \lfloor \ \} \\
 & s\lfloor 0\lceil m \quad .
\end{array}
$$

stap: Voor alle $s \in \mathcal{L}_*(B)$ en $n, m \in \mathsf{Nat}$: Dit betreft het geval $n{+}1$; om hier de definities van $\lceil$ en $\lfloor$ te kunnen toepassen moeten we gevalsonderscheid plegen op s:

geval $s := [\,]$:

$$
\begin{array}{ll}
 & [\,]\lceil(n{+}1{+}m)\lfloor(n{+}1) \\
= & \quad \{ \text{ definitie } \lceil \ \} \\
 & [\,]\lfloor(n{+}1) \\
= & \quad \{ \text{ definitie } \lfloor \ \} \\
 & [\,] \\
= & \quad \{ \text{ definitie } \lceil \ \} \\
 & [\,]\lceil m \\
= & \quad \{ \text{ definitie } \lfloor \ \} \\
 & [\,]\lfloor(n{+}1)\lceil m \quad .
\end{array}
$$

geval $s := b \triangleright s$:

$$
\begin{array}{ll}
 & (b \triangleright s)\lceil(n{+}1{+}m)\lfloor(n{+}1) \\
= & \quad \{ \text{ definitie } \lceil \ \} \\
 & (\,b \triangleright s\lceil(n{+}m)\,)\lfloor(n{+}1) \\
= & \quad \{ \text{ definitie } \lfloor \ \}
\end{array}
$$

$$
\begin{array}{ll}
& s\lceil(n{+}m)\lfloor n \\
= & \quad\{\text{ Inductie Hypothese }\} \\
& s\lfloor n\lceil m \\
= & \quad\{\text{ definitie }\lfloor\ \} \\
& (b\triangleright s)\lfloor(n{+}1)\lceil m \quad .
\end{array}
$$

1. Gegeven zijn datatypen X en Z; we beschouwen een functie F, met, voor alle $n\in\mathsf{Nat}$, type $\mathcal{L}_n(X\to Z)\to X\to\mathcal{L}_n(Z)$, en met de volgende specificatie, voor alle $fs\in\mathcal{L}_n(X\to Z)$ en $x\in X$:

$$(\forall i : 0\le i<n : F\cdot fs\cdot x\cdot i \;=\; fs\cdot i\cdot x) \quad .$$

(a) Leid een recursieve declaratie voor F af. [2 punten]

Oplossing: Met inductie over n. Er is maar maar één lijst ter lengte 0, namelijk $[\,]$, dus alleen al uit het vereiste type van F voor dit geval, namelijk $\mathcal{L}_0(X\to Z)\to X\to\mathcal{L}_0(Z)$, volgt dat de enige keuze is: $F\cdot[\,]\cdot x \;=\; [\,]$. Deze keuze voldoet aan F's specificatie, want voor $n:=0$ krijgen we:

$$
\begin{array}{ll}
& (\forall i : 0\le i<0 : F\cdot[\,]\cdot x\cdot i \;=\; [\,]\cdot i\cdot x) \\
\equiv & \quad\{\text{ leeg domein }\} \\
& \mathsf{true} \quad ,
\end{array}
$$

zelfs ongeacht hoe we F definiëren.

Voor het geval $fs := f\triangleright fs$ nemen F's specificatie als uitgangspunt, en met het oog op de eigenschappen van $\triangleright$ onderscheiden we de gevallen $i:=0$ en $i:=i{+}1$:

$$
\begin{array}{ll}
& F\cdot(f\triangleright fs)\cdot x\cdot 0 \\
= & \quad\{\text{ specificatie } F\ \} \\
& (f\triangleright fs)\cdot 0\cdot x \\
= & \quad\{\text{ eigenschap } \triangleright\ \} \\
& f\cdot x \\
= & \quad\{\ \triangleright\text{-trick, om 0 terug te introduceren }\} \\
& (f\cdot x\triangleright ?)\cdot 0 \quad ,
\end{array}
$$

en, voor $i : 0\le i<n$ $-$met $n = \#fs-$:

$$
\begin{array}{ll}
& F\cdot(f\triangleright fs)\cdot x\cdot(i{+}1) \\
= & \quad\{\text{ specificatie } F\ \} \\
& (f\triangleright fs)\cdot(i{+}1)\cdot x \\
= & \quad\{\text{ eigenschap } \triangleright\ \} \\
& fs\cdot i\cdot x \\
= & \quad\{\text{ specificatie } F\text{, per Inductie Hypothese }\} \\
& F\cdot fs\cdot x\cdot i \\
= & \quad\{\ \triangleright\text{-trick, om } i{+}1 \text{ terug te krijgen }\} \\
& (?\triangleright F\cdot fs\cdot x)\cdot(i{+}1) \quad .
\end{array}
$$

Combineren we de resultaten van deze afleidingen dan vinden we dat de keuze $F \cdot (f \triangleright fs) \cdot x \; = \; f \cdot x \triangleright F \cdot fs \cdot x$ aan F's specificatie voldoet; bovendien is eenvoudig na te gaan dat deze keuze het gewenste type $\mathcal{L}_{n+1}(Z)$ heeft.

Aldus verkrijgen we de volgende recursieve declaratie voor F:

$$\begin{array}{lll} & F \cdot [\,] \cdot x & = \; [\,] \\ \& & F \cdot (f \triangleright fs) \cdot x & = \; f \cdot x \triangleright F \cdot fs \cdot x \end{array}$$

(b) Er bestaat een functie g zodanig dat $F \cdot fs \cdot x \; = \; g \cdot x \bullet fs$. Formuleer het type van zo'n g en formuleer een specificatie voor g. [1 punt]

Oplossing: We leiden af, uitgaande van F's specificatie:

$$\begin{array}{ll} & F \cdot fs \cdot x \cdot i \\ = & \quad \{ \text{ specificatie } F \; \} \\ & fs \cdot i \cdot x \\ = & \quad \{ \text{ sectioning } \} \\ & (\cdot\, x) \cdot (fs \cdot i) \\ = & \quad \{ \text{ specificatie } \bullet \; (\text{“map”}) \; \} \\ & ((\cdot\, x) \bullet fs) \cdot i \quad . \end{array}$$

Hieruit volgt dat ook $F \cdot fs \cdot x \; = \; (\cdot\, x) \bullet fs$ aan F's specificatie voldoet. Het rechterlid is van dezelfde vorm als $g \cdot x \bullet fs$, mits we als specificatie voor g kiezen, voor alle $x \in X$:

$$g \cdot x \; = \; (\cdot\, x) \quad ,$$

en g zal dan type $X \to (X \to Z) \to Z$ moeten hebben (om bij bovenstaande afleiding te passen).

(c) Leid een declaratie af voor g, zoals gespecificeerd in (b). [1 punt]

Oplossing: Dit is eigenlijk een flauwe vraag want er valt niet veel meer af te leiden. Na weglaten van de universele quantificatie (over x) is de specificatie:

$$g \cdot x \; = \; (\cdot\, x)$$

immers al meteen een toegelaten declaratie in FuN; desgewenst kan deze, minder puntvrij, ook worden geschreven als – dat had bij (b) ook al gekund –:

$$g \cdot x \cdot f \; = \; f \cdot x \quad .$$

2. We beschouwen het recursieve datatype $\mathcal{T}(\mathsf{Int})$ van integer binary trees, aldus gedefinieerd:

$$\begin{array}{rll} \langle b \rangle \in \mathcal{T}(\mathsf{Int}) & , \text{voor alle } b \in \mathsf{Int} & , \\ \langle s, c, t \rangle \in \mathcal{T}(\mathsf{Int}) & , \text{voor alle } c \in \mathsf{Int} \text{ en } s, t \in \mathcal{T}(\mathsf{Int}) & . \end{array}$$

Verder is een functie sum, van type $\mathcal{L}_*(\mathsf{Int}) \to \mathsf{Int}$, gegeven, voor alle $b \in \mathsf{Int}$ en $s \in \mathcal{L}_*(\mathsf{Int})$:

$$\begin{array}{lll} sum \cdot [\,] & = & 0 \\ sum \cdot (b \triangleright s) & = & b + sum \cdot s \end{array}$$

Functie tsm, van type $\mathcal{T}(\mathsf{Int}) \to \mathsf{Int}$, is als volgt gedefinieerd, voor alle $b, c \in \mathsf{Int}$ en $s, t \in \mathcal{T}(\mathsf{Int})$:

$$\begin{array}{lll} tsm \cdot \langle b \rangle & = & b \\ tsm \cdot \langle s, c, t \rangle & = & tsm \cdot s + c + tsm \cdot t \end{array}$$

(a) Geef een afleiding van een (recursieve) declaratie voor een functie F, van type $\mathcal{L}_*(\mathcal{T}(\mathsf{Int})) \to \mathsf{Int}$, en die voldoet aan, voor alle $ss \in \mathcal{L}_*(\mathcal{T}(\mathsf{Int}))$: [2 punten]

$$F\cdot ss \quad = \quad sum\cdot(tsm \bullet ss) \quad .$$

In uw oplossing mogen functies sum en tsm niet voorkomen.

Oplossing: Met inductie over het "gewicht"[0] van de lijst ss leiden we af:

$$\begin{array}{ll}
 & F\cdot[\,] \\
= & \quad \{ \text{ specificatie } F \text{ } \} \\
 & sum\cdot(tsm \bullet [\,]) \\
= & \quad \{ \text{ definitie } \bullet \text{ ("map") } \} \\
 & sum\cdot[\,] \\
= & \quad \{ \text{ definitie } sum \text{ } \} \\
 & 0 \quad .
\end{array}$$

Voor het geval van de niet-lege lijsten is het eerste element een tree in $\mathcal{T}(\mathsf{Int})$; daarom is gevalsonderscheid op de twee soorten trees onvermijdelijk:

$$\begin{array}{ll}
 & F\cdot(\langle b\rangle \triangleright ss) \\
= & \quad \{ \text{ specificatie } F \text{ } \} \\
 & sum\cdot(tsm \bullet (\langle b\rangle \triangleright ss)) \\
= & \quad \{ \text{ definitie } \bullet \text{ ("map") } \} \\
 & sum\cdot(tsm\cdot\langle b\rangle \triangleright tsm \bullet ss) \\
= & \quad \{ \text{ definitie } sum \text{ } \} \\
 & tsm\cdot\langle b\rangle + sum\cdot(tsm \bullet ss) \\
= & \quad \{ \text{ definitie } tsm \text{ } \} \\
 & b + sum\cdot(tsm \bullet ss) \\
= & \quad \{ \text{ specificatie } F\text{, per Inductie Hypothese } \} \\
 & b + F\cdot ss \quad ,
\end{array}$$

en:

$$\begin{array}{ll}
 & F\cdot(\langle s,c,t\rangle \triangleright ss) \\
= & \quad \{ \text{ specificatie } F \text{ } \} \\
 & sum\cdot(tsm \bullet (\langle s,c,t\rangle \triangleright ss)) \\
= & \quad \{ \text{ definitie } \bullet \text{ ("map") } \} \\
 & sum\cdot(tsm\cdot\langle s,c,t\rangle \triangleright tsm \bullet ss) \\
= & \quad \{ \text{ definitie } sum \text{ } \} \\
 & tsm\cdot\langle s,c,t\rangle + sum\cdot(tsm \bullet ss) \\
= & \quad \{ \text{ definitie } tsm \text{ } \} \\
 & tsm\cdot s + c + tsm\cdot t + sum\cdot(tsm \bullet ss) \\
= & \quad \{ \text{ algebra } \}
\end{array}$$

[0] Het gewicht van een lijst bomen is de som van de aantallen knopen van alle bomen in de lijst.

$$
\begin{array}{ll}
& c + tsm \cdot s + tsm \cdot t + sum \cdot (\, tsm \bullet ss \,) \\
= & \quad \{ \text{ definitie } sum \text{ (2x) } \} \\
& c + sum \cdot (\, tsm \cdot s \triangleright tsm \cdot t \triangleright tsm \bullet ss \,) \\
= & \quad \{ \text{ definitie } \bullet \text{ ("map") (2x) } \} \\
& c + sum \cdot (\, tsm \bullet (s \triangleright t \triangleright ss) \,) \\
= & \quad \{ \text{ specificatie } F\text{, per Inductie Hypothese } \} \\
& c + F \cdot (s \triangleright t \triangleright ss) \quad .
\end{array}
$$

Aldus verkrijgen we de volgende declaratie voor F:

$$
\begin{array}{lll}
& F \cdot [\,] & = \; 0 \\
\& & F \cdot (\, \langle b \rangle \triangleright ss \,) & = \; b + F \cdot ss \\
\& & F \cdot (\, \langle s, c, t \rangle \triangleright ss \,) & = \; c + F \cdot (s \triangleright t \triangleright ss)
\end{array}
$$

(b) Laat zien dat functie F (min of meer) een generalisatie is van functie tsm. [1 punt]

Oplossing: Om tsm uit te drukken in F leiden we af:

$$
\begin{array}{ll}
& tsm \cdot s \\
= & \quad \{ \text{ definitie } sum \; \} \\
& tsm \cdot s + sum \cdot [\,] \\
= & \quad \{ \text{ definitie } sum \; \} \\
& sum \cdot (\, tsm \cdot s \triangleright [\,]\,) \\
= & \quad \{ \text{ definitie } \bullet \; \} \\
& sum \cdot (\, tsm \cdot s \triangleright tsm \bullet [\,]\,) \\
= & \quad \{ \text{ definitie } \bullet \; \} \\
& sum \cdot (\, tsm \bullet (s \triangleright [\,]\,) \,) \\
= & \quad \{ \text{ lijstnotatie } \} \\
& sum \cdot (\, tsm \bullet [\, s \,]\,) \\
= & \quad \{ \text{ specificatie } F \; \} \\
& F \cdot [\, s \,] \quad .
\end{array}
$$

(c) Leid uit de onder (a) verkregen oplossing een, gelijkwaardige en even efficiënte, staartrecursieve declaratie voor dezelfde functie af. [2 punten]

Oplossing: Gelet op de patronen in de definiërende expressies voor F generaliseren we F tot een nieuwe functie G, met als specificatie, voor alle $x \in \mathsf{Int}$ en $ss \in \mathcal{L}_*(\mathcal{T}(\mathsf{Int}))$:

$$G \cdot x \cdot ss \;=\; x + F \cdot ss \quad .$$

Dit is een echte generalisatie want F kan eenvoudig in G worden uitgedrukt:
$F \cdot ss = G \cdot 0 \cdot ss$. Nu leiden we af, volgens dezelfde patronen als voor F:

$$
\begin{array}{ll}
& G \cdot x \cdot [\,] \\
= & \quad \{ \text{ specificatie } G \text{ } \} \\
& x + F \cdot [\,] \\
= & \quad \{ \text{ definitie } F \; \}
\end{array}
$$

$$x \quad ,$$

en:

$$
\begin{array}{ll}
 & G\cdot x\cdot (\langle b\rangle \triangleright ss) \\
= & \quad \{ \text{ specificatie } G \text{ } \} \\
 & x + F\cdot (\langle b\rangle \triangleright ss) \\
= & \quad \{ \text{ definitie } F \text{ } \} \\
 & x + b + F\cdot ss \\
= & \quad \{ \text{ specificatie } G\text{, per Inductie Hypothese } \} \\
 & G\cdot (x{+}b)\cdot ss \quad ,
\end{array}
$$

en:

$$
\begin{array}{ll}
 & G\cdot x\cdot (\langle s,c,t\rangle \triangleright ss) \\
= & \quad \{ \text{ specificatie } G \text{ } \} \\
 & x + F\cdot (\langle s,c,t\rangle \triangleright ss) \\
= & \quad \{ \text{ definitie } F \text{ } \} \\
 & x + c + F\cdot (s \triangleright t \triangleright ss) \\
= & \quad \{ \text{ specificatie } G\text{, per Inductie Hypothese } \} \\
 & G\cdot (x{+}c)\, (s \triangleright t \triangleright ss) \quad .
\end{array}
$$

Aldus verkrijgen we de volgende (staartrecursieve) declaraties voor F en G:

$$
\begin{array}{llcl}
 & F\cdot ss & = & G\cdot 0\cdot ss \\
\& & G\cdot x\cdot [\,] & = & x \\
\& & G\cdot x\cdot (\langle b\rangle \triangleright ss) & = & G\cdot (x{+}b)\cdot ss \\
\& & G\cdot x\cdot (\langle s,c,t\rangle \triangleright ss) & = & G\cdot (x{+}c)\cdot (s \triangleright t \triangleright ss)
\end{array}
$$

Eindhoven, 2 mei 2012

Rob R. Hoogerwoord
department of mathematics and computing science
Eindhoven University of Technology
postbus 513
5600 MB Eindhoven