Université Pierre & Marie Curie

DESS – GLA

Programmation – Spécifications formelles
II. Spécification ensemblistes – Z

P. Manoury

Nov. - Déc. *2001*

Le langage des ensembles
Rapide rappels

Appartenance (abstraite) : $x \in y$

Égalité (axiome) : $x = y \Leftrightarrow (\forall z. z \in x \Leftrightarrow z \in y)$

Inclusion (définie) : $x \subseteq y \mathrel{\widehat{=}} (\forall z. z \in x \Rightarrow z \in y)$

Constructions de base et axiomes :

	notation	axiome
Ensemble vide :	$\emptyset$	$\forall x. x \notin \emptyset$
Couples	(x, y)	$(x, y) = (x', y') \Leftrightarrow x = x' \wedge y = y'$
Produit	$X \times Y$	$z \in X \times Y \Leftrightarrow \exists x \in X. \exists y \in Y. z = (x, y)$
Union	$X \cup Y$	$z \in X \cup Y \Leftrightarrow z \in X \vee z \in Y$
Ensemble des parties	$\mathcal{P}(X)$	$z \in \mathcal{P}(X) \Leftrightarrow z \subseteq X$
Schéma de compréhension	$\{x \in X \mid \varphi\}$	$y \in \{x \in X \mid \varphi\} \Leftrightarrow y \in X \wedge \varphi[y/x]$

Construtions définies :

	notation	définition
Intersection	$X \cap Y$	$\{z \in X \mid z \in Y\}$
Différence	$X \backslash Y$	$\{z \in X \mid z \notin Y\}$

etc.

Notions fondamentales
Relations – fonctions

Ensemble des relations binaires entre X et Y (définition) :

$$\boxed{X \leftrightarrow Y \mathrel{\widehat{=}} \mathcal{P}(X \times Y)}$$

Relations entre éléments (notation) :

$$\boxed{\text{si } R \in X \leftrightarrow Y,\ x \,\underline{R}\, y \mathrel{\widehat{=}} (x, y) \in R}$$

Domaine et codomaine de $R \in X \leftrightarrow Y$ (définitions) :

$$\boxed{\begin{array}{c} \mathrm{dom}(R) \mathrel{\widehat{=}} \{x \in X \mid \exists y \in Y . x \,\underline{R}\, y\} \\ \mathrm{ran}(R) \mathrel{\widehat{=}} \{y \in Y \mid \exists x \in X . x \,\underline{R}\, y\} \end{array}}$$

Ensemble des fonctions partielles de X vers Y (définition) :

$$\boxed{X \nrightarrow Y \mathrel{\widehat{=}} \{f \in X \leftrightarrow Y \mid \forall x \in \mathrm{dom}(f) . \exists! y \in Y . x \,\underline{f}\, y\}}$$

Ensemble des fonctions totales de X vers Y (définition) :

$$\boxed{X \rightarrow Y \mathrel{\widehat{=}} \{f \in X \nrightarrow Y \mid \mathrm{dom} f = X\}}$$

Application d'une fonction $f \in X \nrightarrow Y$ (axiome) :

$$\boxed{f(x) = y \Leftrightarrow (x, y) \in f}$$

Opération de «mise à jour» :

$$\boxed{\begin{array}{ll} R \oplus S \mathrel{\widehat{=}} \{\ (x, y) \in R \cup S \mid & (x \in \mathrm{dom}(R) \backslash \mathrm{dom}(S) \Rightarrow (x, y) \in R) \wedge \\ & (x \in \mathrm{dom}(S) \Rightarrow (x, y) \in S) \quad \} \end{array}}$$

Exemple : en notant $x \mapsto y$ le couple (x, y)

$$\left\{ \begin{array}{lcl} f \oplus \{x \mapsto y\}(x) & = & y \\ f \oplus \{x \mapsto y\}(z) & = & f(z) \quad \text{si } z \neq x \end{array} \right.$$

Notions fondamentales (suite)
Arithmétique

L'ensemble des entiers naturels (définition) :

$$\boxed{\mathit{IN} \mathrel{\widehat{=}} \textit{ un truc un peu compliqué ...}}$$

Constantes, opérations, relations :

$$\boxed{0\ 1\ \ldots\ +\ \times\ \leq\ \ldots}$$

Entiers non nuls (définition) :

$$\boxed{\mathit{IN}_1 \mathrel{\widehat{=}} \{n \in \mathit{IN} \mid n \neq 0\}}$$

Intervales d'entiers (définition) :

$$\boxed{n..m \mathrel{\widehat{=}} \{k \in \mathit{IN} \mid n \leq k \wedge k \leq m\}}$$

etc.

Structure linéaire générique
Les suites

Ensemble des suites d'éléments de X (définition) :

$$\boxed{\text{seq } X \mathrel{\widehat{=}} \{s \in I\!N_1 \twoheadrightarrow X \mid \exists n \in I\!N . \text{dom}(s) = 1..n\}}$$

Accés aux éléments de $s \in \text{seq } X$:

$$\boxed{s(i)}$$

avec $i \in \text{dom}(s)$

Suite vide :

$$\boxed{\langle\rangle \mathrel{\widehat{=}} \emptyset}$$

remarque : $\emptyset \in 1..0 \twoheadrightarrow X \in \text{seq } X$

Longueur d'une suite :

$$\boxed{\#s}$$

cardinal de l'ensemble s

Suites non vides (définition) :

$$\boxed{\text{seq}_1 \ X \mathrel{\widehat{=}} \{s \in \text{seq } X \mid s \neq \langle\rangle\}}$$

Fonctions sur les suites

Définitions axiomatiques

Format des définitions :
$$\begin{array}{|l}
nom : type \\ \hline
formule
\end{array}$$

Concaténation :
$$\begin{array}{|l}
\frown : \mathrm{seq}\ X \times \mathrm{seq}\ X \rightarrow \mathrm{seq}\ X \\ \hline
\forall s_1, s_2 \in \mathrm{seq}\ X . \forall i \in I\!N_1 . \\
\quad (i \leq \#s_1 \Rightarrow s_1 \frown s_2(i) = s_1(i)) \wedge \\
\quad (i > \#s_1 \Rightarrow s_1 \frown s_2(i) = s_2(i - \#s_1))
\end{array}$$

Sous suite :
$$\begin{array}{|l}
sub : \mathrm{seq}\ X \times I\!N \times I\!N \rightarrow seqX \\ \hline
\forall s \in \mathrm{seq}\ X . \forall i, j \in I\!N . \\
\quad \mathrm{dom}(sub(s,i,j)) = 1..j - i + 1 \\
\quad \forall k \in \mathrm{dom}(sub(s,i,j)) . sub(s,i,j)(k) = s(k + i - 1)
\end{array}$$

Comme des listes :

Constructeurs

$$\begin{array}{|c}
nil : \mathrm{seq}\ X \\ \hline
nil = \emptyset
\end{array}
\qquad
\begin{array}{|l}
cons : X \times \mathrm{seq}\ X \rightarrow \mathrm{seq}_1\ X \\ \hline
\forall s \in \mathrm{seq}\ X . \\
\quad cons(x,s) = \{1 \mapsto x\} \frown s
\end{array}$$

Accesseurs

$$\begin{array}{|l}
car : \mathrm{seq}_1\ X \rightarrow X \\ \hline
\forall s \in \mathrm{seq}_1\ X . \\
\quad car(s) = s(1)
\end{array}
\qquad
\begin{array}{|l}
cdr : \mathrm{seq}_1\ X \rightarrow \mathrm{seq}\ X \\ \hline
\forall s \in \mathrm{seq}_1\ X . \\
\quad cdr(s) = sub(s, 2, \#s)
\end{array}$$

Comme des files d'attentes :

Constructeurs

$$\begin{array}{|c}
new : \mathrm{seq}\ X \\ \hline
new = \emptyset
\end{array}
\qquad
\begin{array}{|l}
add : \mathrm{seq}\ X \times X \rightarrow \mathrm{seq}_1\ X \\ \hline
\forall s \in \mathrm{seq}\ X . \\
\quad add(s,x) = s \frown \{1 \mapsto x\}
\end{array}$$

Accesseurs

$$\begin{array}{|l}
last : \mathrm{seq}_1\ X \rightarrow X \\ \hline
\forall s \in \mathrm{seq}_1\ X . \\
\quad last(s) = s(\#s)
\end{array}
\qquad
\begin{array}{|l}
front : \mathrm{seq}_1\ X \rightarrow \mathrm{seq}\ X \\ \hline
\forall s \in \mathrm{seq}_1\ X . \\
\quad front(s) = sub(s, 1, \#s - 1)
\end{array}$$

Retournement d'une suite
Schéma d'opérations

Relation (prédicat) *avant-aprés* :

$$Rev \in \text{seq } X \leftrightarrow \text{seq } X$$

Intuitivement, procédure *vs* fonction :

$$s \ \underline{Rev} \ s' \ \widehat{=} \ \text{«}s' \textit{ est l'état de } s \textit{ aprés l'opération»}$$

Format des schémas d'opération :

$$\begin{array}{|l}
\multicolumn{1}{l}{Nom} \\
\hline
\\
déclarations \\
\\
\hline
\\
formule \\
\\
\hline
\end{array}$$

L'opération de retournement de suite :

$$\begin{array}{|l}
\multicolumn{1}{l}{Rev} \\
\hline
\\
s, s' : \text{seq } X \\
\\
\hline
\\
\text{dom}(s) = \text{dom}(s') \wedge \\
\forall i \in \text{dom}(s'), s'(i) = s(\#s - i + 1) \\
\\
\hline
\end{array}$$

Schémas d'opérations
Quelque manipulations

Apostrophe :

$$\begin{array}{|l} \multicolumn{1}{l}{S} \\ \hline x : T \\ \hline \varphi \\ \hline \end{array} \quad \leadsto \quad \begin{array}{|l} \multicolumn{1}{l}{S'} \\ \hline x' : T \\ \hline \varphi[x'/x] \\ \hline \end{array}$$

Conjonction :

$$\begin{array}{|l} \multicolumn{1}{l}{S_1} \\ \hline x_1 : T_1 \\ \hline \varphi_1 \\ \hline \end{array} \quad \begin{array}{|l} \multicolumn{1}{l}{S_2} \\ \hline x_2 : T_2 \\ \hline \varphi_2 \\ \hline \end{array} \quad \leadsto \quad \begin{array}{|l} \multicolumn{1}{l}{S_1 \wedge S_2} \\ \hline x_1 : T_1 \\ x_2 : T_2 \\ \hline \varphi_1 \wedge \varphi_2 \\ \hline \end{array}$$

Inclusion :

$$\begin{array}{|l} \multicolumn{1}{l}{S_2} \\ \hline S_1 \\ x_2 : T_2 \\ \hline \varphi_2 \\ \hline \end{array} \quad \widehat{=} \quad \begin{array}{|l} \multicolumn{1}{l}{S_2} \\ \hline x_1 : T_1 \\ x_2 : T_2 \\ \hline \varphi_1 \wedge \varphi_2 \\ \hline \end{array}$$

Identificateurs :

$$\begin{array}{|l} \multicolumn{1}{l}{S} \\ \hline x_1 : T_1 \\ \ldots \\ x_n : T_n \\ \hline \varphi \\ \hline \end{array} \quad \leadsto \quad \theta S \mathrel{\widehat{=}} (x_1, \ldots, x_n)$$

IBM CICS API
Temporary Storage Queue

Ensembles de base :

$$\begin{array}{rcl} BYTE & \widehat{=} & 0..255 \\ TSElem & \widehat{=} & \text{seq } BYTE \end{array}$$

Une suite et un pointeur (sur le dernier objet modifié) :

$$\begin{array}{|l} \multicolumn{1}{l}{TSQ} \\ \hline ar : \text{seq } TSElem \\ p : IN \\ \hline p \leq \#ar \\ \hline \end{array}$$

Initialisation :

$$\begin{array}{|l} \multicolumn{1}{l}{TSQInit} \\ \hline TSQ \\ \hline ar = \langle\rangle \wedge p = 0 \\ \hline \end{array}$$

Schéma «avant-aprés» :

$$\begin{array}{|l} \multicolumn{1}{l}{\Delta TSQ} \\ \hline TSQ \wedge TSQ' \\ \hline \\ \hline \end{array}$$

Temporary Storage Queue
Opérations fournies

Entrées-sortie, notations conventionnelles :

S

$$\begin{array}{|l}
\hline
in? : T_1 \\
out! : T_2 \\
\ldots \\
\hline
\varphi \\
\hline
\end{array}$$

Ajout et retrait :

$Append0$

$$\begin{array}{|l}
\hline
\Delta TSQ \\
from? : TSElem \\
item! : \mathit{IN} \\
\hline
ar' = ar \frown \langle from? \rangle \land \\
item! = \# ar' \land \\
p' = p \\
\hline
\end{array}$$

$Remove0$

$$\begin{array}{|l}
\hline
\Delta TSQ \\
item! : TSElem \\
\hline
p < \# ar \land \\
p' = p + 1 \land \\
into! = ar(p') \land \\
ar' = ar \\
\hline
\end{array}$$

Lecture et écriture :

$Write0$

$$\begin{array}{|l}
\hline
\Delta TSQ \\
item? : \mathit{IN} \\
from? : TSElem \\
\hline
item? \in 1..\# ar \land \\
ar' = ar \oplus \{ item? \mapsto from? \} \land \\
p' = p \\
\hline
\end{array}$$

$Read0$

$$\begin{array}{|l}
\hline
\Delta TSQ \\
item? : \mathit{IN} \\
into! : TSElem \\
\hline
item? \in 1..\# ar \land \\
into! = ar(item?) \land \\
p' = item? \land \\
ar' = ar \\
\hline
\end{array}$$

Temporary Storage Queue
Gestion d'erreurs

Statuts d'exécution :

$$OpStatus ::= Success \mid ItemErr \mid NoSpace$$

(macro syntaxique pour $OpStatus = \{Success, ItemErr, NoSpace\}$)

Schéma générique d'erreur :

$$
\begin{array}{|l}
\multicolumn{1}{l}{ERROR} \\ \hline
\Delta TSQ \\
report! : OpStatus \\ \hline
\theta TSQ' = \theta TSQ \\ \hline
\end{array}
$$

Rapports d'erreurs :

$$
\begin{array}{|l}
\multicolumn{1}{l}{NoneLeft} \\ \hline
ERROR \\ \hline
p = \#ar \wedge \\
report! = ItemErr \\ \hline
\end{array}
$$

$$
\begin{array}{|l}
\multicolumn{1}{l}{OutOfBounds} \\ \hline
ERROR \\
item? : \mathbb{N} \\ \hline
item? \notin 1..\#ar \wedge \\
report! = ItemErr \\ \hline
\end{array}
$$

$$
\begin{array}{|l}
\multicolumn{1}{l}{OutOfSpace} \\ \hline
ERROR \\ \hline
report! = NoSpace \\ \hline
\end{array}
$$

Tout va bien :

$$
\begin{array}{|l}
\multicolumn{1}{l}{Successful} \\ \hline
report! : OpStatus \\ \hline
report! = Success \\ \hline
\end{array}
$$

Temporary Storage Queue
Intégration

$$
\begin{aligned}
Append &\mathrel{\widehat{=}} (Append0 \land Successful) \lor OutOfSpace \\
Remove &\mathrel{\widehat{=}} (Remove0 \land Successful) \lor NoneLeft \\
Write &\mathrel{\widehat{=}} (Write0 \land Successful) \lor OutOfBound \lor OutOfSpace \\
Read &\mathrel{\widehat{=}} (Read0 \land Successful) \lor OutOfBound
\end{aligned}
$$