

EXAMEN du cours SEMANTIQUE DES LANGAGES INFORMATIQUES,  
Semestre Printemps 2017,  
Examen du 13 juin 2017.

NOM :

PRENOM :

Points : /120

Note :

## 1 Typage dans les langages de programmation

Question 1.1 : Définition / 30 points

Donner la terminologie, un exemple et la définition de :

a) Ce qu'est un type ?

b) Comment déterminer le type d'une expression ?

c) Que veut dire compatibilité de type ?

d) Que veut dire le terme polymorphisme ?

e) Quel sont les principes des mécanismes de visibilité statique ?

f) Quel sont les principes des mécanismes de visibilité dynamique ?

## 2 Sémantique d'un langage pour Tetris

Le jeu de Tetris consiste en des pièces tombant les unes sur les autres, le joueur peut faire des rotations et des déplacements lorsqu'elles tombent. Une ligne complète disparaît lorsqu'elle est obtenue par la chute des pièces.

La zone de chute est une surface de 10 de largeur et 20 de hauteur.

Les sept pièces possibles sont constituées de 4 carrés élémentaires connexes :

- carré = C
- L gauche = LL
- L droit = LR
- T = T
- Z gauche = ZL
- Z droit = ZR
- Barre = B

Chaque pièce a quatre positions de rotation possibles, les positions 1,2,3,4. (en fait pour le carré elles sont toutes équivalentes) et une position latérale (1 à 10). Attention certaines positions sont impossibles, mais cela sera vérifié par la traduction dans le damier et la position correcte la plus proche sera adoptée.

Le langage est celui des successions de chutes de pièces et des mouvements latéraux ou des rotations. finalement move fait descendre la pièce d'un cran :

*Left, Right, RotL, RotR, move*

Pour simplifier, une réserve des pièces, sous la forme d'une liste, devant tomber est fournie comme état initial. Ils chuteront dans l'ordre toutes les fois où il n'y a plus que des pièces immobiles.

Exemple : [C,LL,LL,ZR,B,LR,C,ZR]

### Sémantique des pièces

Pour simplifier, nous supposons une série de fonctions traitant de la géométrie des pièces. Il n'est pas nécessaire de définir formellement ces fonctions qui sont supposées exister. Elles seront nécessaires pour définir la sémantique du langage.

L'opération 'Translate' traduit en position sur le damier une pièce d'une certaine forme et d'une certaine rotation et position.

*Translate : Piece, Pos, X, Y → ZoneChute*

Un damier défini comme un type abstrait *ZoneChute* a certaines opérations :

*DamierVide =  $\emptyset_{ZC} : - \rightarrow ZoneChute$*

*DamierUnion =  $\cup_{ZC} : ZoneChute, ZoneChute \rightarrow ZoneChute$*

*DamierIntersection =  $\cap_{ZC} : ZoneChute, ZoneChute \rightarrow ZoneChute$*

*DamierEffacerLine =  $clean_{ZC} : ZoneChute \rightarrow ZoneChute$*

## Comment donner une sémantique à ce langage ?

- il faut fixer un domaine sémantique et
- définir des règles pour définir une sémantique d'évaluation.

## Langage des mouvements

Nous désirons définir la sémantique du langage pour que nous puissions décrire pour chaque joueur les actions de celui-ci et si le jeu doit se terminer et quel est le nombre de lignes détruites.

Question 2.2 : Quel doit être le langage pour les programmes et autres domaines syntaxiques ? /10

$\text{Instr} = \{Left, Right, RotL, RotR, move\}, \text{Prog} = \text{Instr}^*, \text{Pieces} = \{C, LL, LR, T, ZL, ZR, B\},$   
Variables  
 $X, Y \in \mathbb{N}, LP \in \text{Pieces}^*, ZC \in \text{ZoneChute}, P \in [1, \dots, 4], Pi, Piece \in \text{Pieces}$

Question 2.3 : Quel doit être le domaine sémantique pour tetris ? /10

Semantic Domain=  
 $\text{SD} = \text{ZoneChute} \times \text{Pieces}^* \times \mathbb{N} \times \mathbb{N} \times [1, \dots, 4] \times \text{Pieces} \cup \{\perp\}$

Question 2.4 : Donner les relations de transitions nécessaires ? /10

Relation de transition =  $\Rightarrow \subseteq \text{SD} \times \text{Prog} \times \text{SD}$

Question 2.5 : Donner les règles sémantique pour les programmes ? /30

$$\begin{array}{l}
\text{Consommation Piece} = \frac{ZC' = \text{clean}_{ZC}(ZC)}{\langle ZC, Pi.LP, X, Y, P, \perp \rangle \text{prog} \Rightarrow \langle ZC', LP, 5, 20, 1, Pi \rangle} \\
\text{Instruction pas touché} \\
\frac{\langle X, Y, P \rangle, \text{instr} \rightarrow \langle X', Y', P' \rangle, \text{Translate}(Pi, P', X', Y') \cap ZC = \emptyset_{ZC}}{\langle ZC, LP, X, Y, P, Pi \rangle \text{instr.prog} \Rightarrow \langle ZC, LP, X', Y', P', Pi \rangle} \\
\text{Instruction touché} \\
\frac{\langle X, Y, P \rangle, \text{instr} \rightarrow \langle X', Y', P' \rangle, \text{Translate}(Pi, P', X', Y') \cap ZC \neq \emptyset_{ZC}, ZC' = ZC \cup_{ZC} \text{Translate}(Pi, P', X', Y')}{\langle ZC, LP, X, Y, P, Pi \rangle \text{instr.prog} \Rightarrow \langle ZC', LP, X', Y', P', \perp \rangle} \\
\text{Instructions} \frac{X' = X - 1}{\langle X, Y, P \rangle, \text{Left} \rightarrow \langle X', Y, P \rangle} \\
\text{Instructions} \frac{X' = X + 1}{\langle X, Y, P \rangle, \text{Right} \rightarrow \langle X', Y, P \rangle} \\
\text{Instructions} \frac{P' = P - 1}{\langle X, Y, P \rangle, \text{RotL} \rightarrow \langle X, Y, P' \rangle} \\
\text{Instructions} \frac{P' = P + 1}{\langle X, Y, P \rangle, \text{RotR} \rightarrow \langle X, Y, P' \rangle} \\
\text{Instructions} \frac{Y = Y + 1}{\langle X, Y, P \rangle, \text{move} \rightarrow \langle X, Y', P \rangle}
\end{array}$$

Question 2.6 : Définir un état initial du jeu non vide qui puisse terminer par une zone de chute vide DamierVide?(Current)=true? /10  
LP= CCCCC

Question 2.7 : Définir un programme pour atteindre DamierVide?(Current)=true avec l'état initial précédent ? /10  
Prog= Left\*4, move\*20, Left\*2, move\*20, move\*20, Right\*2, move\*20, Right\*4, move\*20

Question 2.8 : Prouver que le programme précédent amène a  $\text{DamierVide} \wedge (\text{Current}) = \text{true}$ /10