

# Examen de rattrapage d'ILP

durée 2 heures (Revision: 1.6 )

Christian Queinnec

08 septembre 2006

## Conditions générales

Cet examen est formé d'un unique problème en plusieurs questions auxquelles vous pouvez répondre dans l'ordre qui vous plait.

Le barème est fixé à 20 ; la durée de l'épreuve est de 2 heures. Tous les documents sont autorisés et notamment ceux du cours<sup>1</sup>.

Votre copie sera formée de fichiers (ASCII ou ISO-latin 1 ou 15 seulement) dont les lignes contiendront au plus 78 caractères. Vous rassemblerez ces fichiers dans un répertoire nommé `ilp` placé immédiatement dans votre répertoire personnel (ou `HOME`). Le répertoire `ilp` (pas votre `HOME`) sera ramassé automatiquement en fin d'examen par le centre de calcul. Seuls les fichiers mentionnés dans les livraisons à effectuer seront ramassés. Aucun fichier des sources d'ILP n'est à modifier (sauf si demandé).

L'examen sera corrigé à la main, il n'est donc pas utile de s'acharner sur un problème de compilation. Il est beaucoup plus important de rendre aisé, voire plaisant, le travail du correcteur et, notamment, de l'aider, par de judicieux commentaires, à comprendre vos intentions.

Vous pouvez vous aider des machines pour naviguer dans la documentation ou dans le code d'ILP4 (avec Emacs (`etags`) ou Eclipse). Attention, il est peu conseillé que vous lanciez tous Eclipse en même temps : Eclipse n'est pas nécessaire pour se balader dans le code.

## 1 Vérification statique d'arité

Le but de ce problème est de vérifier au plus vite que les fonctions globales (non primitives) sont correctement invoquées : lorsqu'une fonction globale  $f$  est définie (par le programmeur) et possède une arité  $n$  alors toute invocation à  $f$  doit comporter exactement  $n$  arguments. Cette vérification est nommée « vérification d'arité ».

Rappelons que, pour cet examen, le langage de référence est ILP4. On n'oubliera pas la force de persuasion que peut revêtir un croquis (en ASCII-art) bien pensé ou un commentaire pertinent.

### Question 1

Après qu'un programme est lu comme un DOM, quelles sont, dans le cas de l'interprétation et de la compilation, les phases, transformations ou passes successivement appliquées sur ce DOM ? Dans quelles classes ou méthodes ou autres entités d'ILP4 est **effectivement** réalisée la vérification d'arité concernant les fonctions primitives ? La vérification est-elle statique ou dynamique ?

### Livraison

- un fichier textuel nommé `q1` contenant votre réponse.

### Notation sur 4 points

- 4 points

---

<sup>1</sup><http://www-master.ufr-info-p6.jussieu.fr/2005/Ext/queinnec/>

## Question 2

Dans quelles classes ou méthodes ou autres entités d'ILP4 est **effectivement** réalisée la vérification d'arité concernant les fonctions globales non primitives ? La vérification est-elle statique ou dynamique ?

### Livraison

- un fichier textuel nommé **q2** contenant votre réponse.

### Notation sur 4 points

- 4 points

## Question 3

Que ce soit pour l'interprétation ou la compilation, on souhaite détecter statiquement les invocations à des fonctions globales définies par le programmeur dont le nombre d'arguments n'est pas approprié. Dans quelle passe insèreriez-vous cette analyse ? Argumentez votre réponse en détaillant notamment la technique que vous comptez mettre en œuvre, les informations dont vous avez besoin et la disponibilité de ces informations.

### Livraison

- un fichier textuel nommé **q3** contenant votre réponse.

### Notation sur 3 points

- 3 points

## Question 4

Si vous aviez à modifier les fichiers d'ILP4, quelles modifications apporteriez-vous pour implanter l'analyse statique de vérification d'arité concernant les fonctions globales ?

### Livraison

- un fichier textuel nommé **q4** contenant votre réponse.

### Notation sur 4 points

- 4 points

## Question 5

Vous ne pouvez modifier les sources d'ILP4, écrivez le code Java implantant l'analyse statique de vérification d'arité concernant les fonctions globales. Les classes que vous aurez à écrire devront appartenir au paquetage `fr.upmc.ilp.ilp4fns`. Pensez à expliquer les nouveaux problèmes qui surgissent et comment vous les résolvez.

### Livraison

- des fichiers Java. Ces fichiers Java seront placés dans `ilp/fr/upmc/ilp/ilp4fns/`

### Notation sur 5 points

- 5 points