



Éléments de correction Examen final réparti d'ILP

Christian Queinnec

5 janvier 2012

Voici quelques éléments de correction de l'examen de février 2012 qui complètent les éléments de janvier (cf. sujet enrichi).

Méfiez vous des imports sauvages. Des `ParseException`, il y en a dans plein de bibliothèques. De plus, vous ne devez jamais utiliser de classe dont le nom comporte des mots comme `internal` ou `com.sun`.

Ne confondez pas la définition d'une fonction (qui n'arrive qu'une seule fois au début du programme) avec ses potentiellement multiples invocations.

Ne confondez pas les temps! L'intégration est une transformation de programme source donc une analyse statique (pas dynamique!). Décider d'intégrer si la valeur à l'exécution de la fonction invoquée satisfait une quelconque propriété n'a pas de sens.

Méfiez-vous des transtypages : quand Eclipse vous signale un problème du genre `pomme = orange`, n'ayez pas le réflexe d'écrire (ce qui satisfait le compilateur mais pas le correcteur humain) : `pomme = (Pomme) orange`. Dans le même genre et en combinant avec le mélange des temps précédent, n'écrivez pas `(IAST4expression) lexenv.lookup(...)`. Comment une valeur provenant de l'exécution du programme pourrait-elle être convertie en un AST?

Ne confondez pas les temps! Si, à l'interprétation ou à la compilation ou à l'exécution, vous avez besoin d'un booléen, pourquoi l'ajouter à la grammaire? Vous invalidez ainsi tous les programmes de tests. Et pourquoi modifier l'AST d'un programme lorsque celui-ci est exécuté?