

Interprétation des programmes – TP 5 :

Inférence des types

Université Paris Diderot – Master 1

(2014-2015)

Cette feuille de TP vous donne les étapes à suivre pour implémenter l'inférence de types de HOPIX.
Le code source correspondant à ces travaux pratiques se trouve sur le GIT, dont on rappelle l'URL :

<http://moule.informatique.univ-paris-diderot.fr:8080/Yann/compilation-m1>

On rappelle que vous devez faire des *commits* réguliers (à chaque modification de votre code) pour que nous puissions suivre votre avancement.

Exercice 1 Préannotation

1. À quoi sert la fonction `fresh_unification_type_variable` ?
2. Implémentez `preannotate_expression`.

□

Exercice 2 Génération des contraintes

1. Quel est le langage des contraintes de type utilisé par le moteur d'inférence de HOPIX ?
2. Donnez une définition de la satisfiabilité d'une telle contrainte.
3. Implémentez les règles de génération des contraintes dans la fonction `generate_constraint`

□

Exercice 3 Résolutions des contraintes

1. Pour mieux comprendre le processus de résolution des contraintes, essayez de résoudre sur papier les contraintes suivantes :
 - `int = X1`
 - `int = int; X1 = bool`
 - `int = bool`
 - `X1 = X2 -> X3; X2 = X3; X2 * X3 = int * int`
 - `X1 = X2 -> X3; X2 = int`
 - `X1 = X2 -> X3; X2 = int`
 - `X1 = int; X1 -> bool = bool -> bool`
2. Implémentez la fonction `unify`.
3. Pourquoi votre fonction `unify` n'introduit-elle jamais de cycle dans la substitution qu'elle calcule ?
4. Pourquoi votre fonction `unify` termine-t-elle ?

□

Exercice 4 Élaboration de l'expression explicitement typée

1. Implémentez la fonction `elaborate`.

□