

Compilation – TP 2 : Langage du premier ordre

Université Paris Diderot – Master 2
(2014-2015)

Nous avons introduit deux langages :

- FOPIX : un langage du premier ordre, c'est-à-dire un langage avec variables, expressions arithmétiques et appels de fonctions primitives.
- STACKIX : un langage de bas-niveau pour une machine à pile.

Cette séance de travaux pratiques a pour objectifs :

- de vous faire découvrir le code source du projet en introduisant une extension simple du langage FOPIX – les expressions conditionnelles – dans les différents modules du compilateur de FOPIX vers STACKIX ;
- de vous confronter aux problèmes soulevés par la définition et l'appel de fonctions définies par le programmeur.

Pour travailler sur votre projet, vous devez tout d'abord *cloner* le dépôt GIT contenant le code à compléter. Il se trouve sur le GITLAB de l'UFR :

<http://moule.informatique.univ-paris-diderot.fr:8080/letouzey/compilation-m2>

Voici les instructions à suivre :

1. Pour vous logger sur le serveur GITLAB, vous devez utiliser l'onglet LDAP et vos identifiant et mot de passe de votre compte UFR.
2. Un des membres du groupes doit cloner le projet en cliquant sur *Fork repository*.
3. Cette même personne doit ensuite rajouter les autres membres du groupe à son projet par le menu *Settings > Members*. Pierre Letouzey et doit aussi être rajouté au projet.

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

Exercice 1 (Expressions conditionnelles)

1. Déterminer quelle est la construction syntaxique de l'arbre de syntaxe abstraite de FOPIX qui représente une construction « *if ... then ... else ...* ».
2. Écrire un programme FOPIX simple utilisant une expression conditionnelle.
3. Modifier l'interpréteur de FOPIX pour traiter les expressions conditionnelles.
4. Modifier la passe de traduction de FOPIX à STACKIX pour rajouter les expressions conditionnelles.

□

Exercice 2 (Définitions et appels de fonction)

1. Déterminer à quelles constructions de l'arbre de syntaxe abstraite correspondent les définitions et les appels de fonctions.
2. Écrire la fonction factorielle en FOPIX et l'expression qui calcule 5!
3. Implémenter un nouvel environnement pour l'interpréteur FOPIX dont le rôle est d'associer à chaque identificateur de fonction utilisateur les informations suivantes :
 - la liste des arguments formelles de la fonction ;
 - l'expression qui correspond au corps de cette fonction.

4. *Modifier l'interpréteur de FOPIX pour qu'il prenne en charge les définitions et les appels de fonctions. (Attention : les fonctions FOPIX sont toutes définies de façon mutuellement récursives.)*
5. *Étudier l'instruction UJUMP de la machine de STACKIX. Quelle est sa sémantique ? À quoi peut-elle servir ?*
6. *Modifier la passe de traduction de FOPIX vers STACKIX pour prendre en charge les définitions et les appels de fonctions.*
7. *Implémenter le tri rapide en FOPIX. Comparer le temps d'exécution du tri rapide sur des tableaux de taille importante si il est interprété ou si il est compilé vers STACKIX.*

□