

Sémantique opérationnelle

Réalisation d'un interprète du langage PCF

Introduction

Le but de ce TP est de programmer la partie sémantique d'un interprète du langage PCF du TP précédent. Les principaux composants du programme, qui ont formé le corrigé du TP précédent, vous sont donnés : analyseurs lexical et syntaxique, boucle principale, ainsi que quelques éléments de la partie sémantique.

Les fichiers desquels vous devez partir sont les suivants :

- `pcflex.mll` : l'analyseur lexical
- `pcfasm.ml` : le type des arbres de syntaxe abstraite, et leur imprimeur (*pretty-printer*)
- `pcfparse.mly` : l'analyseur syntaxique
- `pcfsem.ml` : (nouveau) des éléments de sémantique opérationnelle du langage :
 - le type `pcfval` des valeurs sémantiques
 - des indications relatives aux différents éléments à produire (environnements, fonction sémantique).
- `pcfloop.ml` : la boucle principale.

Compiler et recompiler

Dans un terminal, exécuter dans votre dossier de travail :

```
make
```

Quelquefois, il peut être nécessaire d'exécuter :

```
make clean
```

au préalable, afin d'effacer les traces des compilations précédentes.

1 Traitement des constantes et des opérateurs

Q1 Complétez `pcfsem.ml` de sorte que le traitement des constantes entières (`PCFast.EInt _`), booléennes (`EBool _`), de la conditionnelle (`PCFast.EIf(_, _, _)`), et des opérateurs unaires et binaires soit effectif.

La fonction sémantique `eval` devra garder son paramètre `rho` même s'il n'est jamais utilisé à ce stade (il le sera à partir de la question suivante).

À chaque ajout, recompilez et testez.

2 Traitement des identificateurs

On traitera les environnements (`rho`) comme des listes des couples (*identificateur*, *valeur*). L'environnement initial (vide) est la liste vide, et la fonction `lookup` produit la valeur associée à un identificateur, lorsque c'est possible.

Q2 Testez les fonctions `lookup` et `extend` sur de petits exemples. (Demandez à OCaml quel est leur type.)

Q3 Complétez le traitement des identificateurs et du `let` dans la fonction `eval`.

3 Fonctions et applications

Q4 Les valeurs sémantiques des fonctions de PCF sont essentiellement celles décrites durant le cours. Complétez le traitement des fonctions et des applications dans la fonction `eval`.

4 Déclarations récursives

La syntaxe des définitions récursives permet de définir des **fonctions** récursives locales. Vous avez vu, durant le cours, qu'une fonction récursive disposait de son propre nom qu'elle associait à sa propre valeur lors de l'évaluation de chacun de ses appels.

Q5 Complétez votre fonction sémantique en donnant la sémantique de ces fonctions récursives locales.

5 S'il vous reste du temps ...

... enrichissez votre langage à votre gré : traitement des chaînes de caractères, environnement primitif plus riche, définitions globales, *etc.*