

Projet Programmation 1

Compilation d'expressions arithmétiques

Marianne Déglise

1 Lexer

L'analyseur syntaxique transforme l'expression lue (chaîne de caractères) en une liste de lexemes. On profite aussi de cette étape pour soulever des erreurs de lexique : des mélanges de lettres et nombres non séparés par des espaces ou des opérations, des mots qui ne sont pas dans notre langage (c'est à dire différents de "int" ou "float"), des points trop nombreux, etc.

2 Parser

Le parser transforme ensuite la suite de lexemes générée par le lexer en un arbre (type exp) correspondant à l'arbre des opérations.

Dans le principe :

On fait un appel récursif sur la liste avec un deuxième argument (appelé "gauche") qui garde en mémoire l'arbre de la partie déjà vue de la liste. Lorsque l'on croise une opération op, gauche devient le sous-arbre gauche, et il reste à calculer le sous-arbre droit puis à continuer. Ceci est fait par la fonction récursive `parcourt_droite` qui renvoie un couple : l'arbre droit et le reste de la liste, où l'arbre droit correspond au bloc d'expressions qui est composé de blocs de parenthèses séparés par des * ou des mod ou quot, qui sont les opérations prioritaires. C'est grâce à cette fonction auxiliaire, ainsi que grâce aux deux fonctions `recherche_pack`, que l'on peut gérer les priorités opératoires. Les cas particuliers de la conversion (de int à float ou l'inverse), du modulo et des opérations unaires sont gérés séparément avec les fonctions `recherche_pack_`.

Enfin, avant de renvoyer l'arbre, on le parcourt pour vérifier le typage de l'expression avec la fonction `bon_typage`.

3 Compilation

On parcourt l'arbre généré par le parser, et lorsque l'on arrive sur une opération, on appelle la fonction récursivement sur chaque sous-arbre et on applique l'opération sur le résultat.

Pour gérer les enregistrements, on utilise la pile : dans les fonctions opération, on pop les résultats des deux précédents appels (`parcourt_e1` et `parcourt_e2`), puis on push le résultat de l'opération. Comme les flottants ne peuvent pas être push ou pop sur la pile, on incrémente le pointeur de la pile (`rsp`) avant de déplacer le flottant à cet endroit (i.e. on l'ajoute "à la main" en haut de la pile), et inversement lorsqu'on dépile.