

Analyse Syntaxique - Année 2010-2011

Feuille 6

Construction d'analyseurs syntaxiques avec yacc

Exercice 6.1

1. Reprendre la grammaire des expressions arithmétiques du TD 5. Ajouter des règles de grammaire permettant d'analyser plusieurs lignes de calcul.
2. Modifier l'analyseur afin d'utiliser des nombres réels (vous devrez modifier à la fois le fichier `lex` et le fichier `yacc`). Les attributs des nombres seront de type `double`.

Exercice 6.2

1. Reprendre la grammaire précédente. Ajouter des règles pour pouvoir affecter à une variable le résultat d'un calcul numérique.
Par exemple $x = 1.5 + 17 * 9.3;$
2. Utiliser pour les variables des attributs qui sont des couples (`nom`, `valeur`). Le programme devra afficher après chaque affectation l'identificateur concerné, ainsi que sa valeur.

Exercice 6.3

1. Créer dans le fichier `yacc` de l'exercice précédent une table de symboles, destinée à contenir les attributs des variables. On pourra utiliser les fichiers `symtab.c`, `symtab.h` pour le traitement de la table de symboles (recherche et insertion).
Le programme devra afficher en fin d'analyse la liste de tous les identificateurs, avec leur valeur.
2. Modifier votre analyseur pour pouvoir évaluer des expressions contenant des variables auxquelles on a précédemment affecté des valeurs. Exemple :
 $x = 9.3;$
 $y = 1.5 + 17 * x;$

Exercice 6.4

Écrire un source `lex` et `yacc` permettant de reconnaître les expressions booléennes correctement formées, utilisant l'alphabet de terminaux $\{0, 1, (,), \vee, \wedge, \neg\}$. Le programme devra évaluer la valeur de l'expression. On utilisera une grammaire qui respecte la priorité des opérateurs.