Travaux Dirigés de Traduction n°5

Licence d'informatique

———

Analyse syntaxique avec `bison`

Dans ce TD, nous allons manipuler un petit analyseur d'expressions numériques. Nous découvrirons la gestion de mots-clé, et nous verrons comment gérer des valeurs de types différents.

▸ Exercice 1. *Étude de programme*

Téléchargez les fichiers `flex` *et* `bison` *suivants :*

```
http://igm.univ-mlv.fr/~bedon/Enseignement/Traduction/td5/calc.lex
http://igm.univ-mlv.fr/~bedon/Enseignement/Traduction/td5/calc.y
http://igm.univ-mlv.fr/~bedon/Enseignement/Traduction/td5/makefile
```

Compilez ces programmes.

1. *Que fait le programme* `calc` *ainsi obtenu ?*
2. *A quoi sert l'option* `-d` *de* `bison` *? Si on ajoute l'option* `-v`*, on obtient un fichier supplémentaire. Que contient-il ?*
3. *Que fait la déclaration*

   ```
   %union {
      int entier;
      float reel;
   }
   ```

   dans le fichier `calc.y` *?*

▸ Exercice 2. *Améliorons le programme*

1. *Modifiez le code* `bison` *pour que l'on puisse utiliser les opérateurs* `-` *(binaire et unaire) et* `/`*, ainsi que les parenthèses. Vous devrez utilisez les priorités et associativités usuelles.*

2. *On souhaite utiliser 26 variables, désignées par les lettres majuscules. Ajoutez du code C pour gérer ces variables à l'aide d'un tableau.*
3. *Modifiez le code* `flex` *de façon à ce que, quand on lit un nom de variable, l'analyseur lexical renvoie un token* `IDENT`*, et stocke dans* `yylval` *un caractère désignant la variable.*
4. *Modifiez le code* `bison` *pour qu'on puisse utiliser ces variables dans les expressions. Exemple :* `1.2+(A*6)`
5. *Modifiez le code* `bison` *pour que l'on puisse affecter à une variable la valeur d'une expression. Exemple :* `C=1.2+(A*6)`
6. *Ajoutez les fonctions mathématiques : log, exp, sin, cos, tan, $sqrt$ et pow.*
7. *Gérez le symbole $ dont la valeur est le résultat de la ligne précédente.*