Les types des valeurs sémantiques en Yacc

En Yacc le type des valeurs sémantiques s'il est élémentaire et si c'est le même pour tous les non-terminaux peut être déclaré par :

```
{#define YYSTYPE}  nom-de-type
```

Si les non-terminaux ont des types différents on utilise la directive `%union` de Yacc pour donner tous les types possibles , puis `%type <nomdetypeunion> nonterminal` pour donner le type de chacun des non terminaux.

Par exemple pour déclarer que la valeur sémantique est : soit un entier pour les non terminaux *expression terme facteur*, soit une chaîne de caractères pour*statement* on écrit :

```
%union{int nombre;
       char *chaine;
      }
%type <nombre> expression terme facteur
%type <chaine> statement
```

Si on veut déclarer qu'un non-terminal à un type `struc` on déclare ce type dans une `%union` et on attribue le type à un non-terminal par `%type`.

Par exemple on veut que la valeur sémantique soit un enregistrement dont un des champs est le nombre de registres et l'autre le code on a alors :

```
%union {struct enregitrement{
            int nbreg;
            char* code;
        }enregistrement;}

%type <enregistrement> expression facteur terme ID
```

Dans le source C associé aux règles on peut alors écrire `$.code et $.nbreg`