Feuille de TP n° 5 (séance 7) : typage d'expressions

Nous allons travailler en TP sur un langage très simple d'expressions logiques et arithmetiques. Le but de ce TP est de réaliser son vérificateur de types.

Pour cela, vous disposez de l'analyseur lexical et syntaxique du langage dans le répertoire `/ens/sighirea/Prog2/miniExp/` :

- `explexer.mll` définition des unitées lexicales ;
- `expparser.mly` définition de l'analyseur syntaxique ;
- `expsyntax.ml` définition de la syntaxe abstraite ;
- `analyseur.ml` appel des fonctions de l'analyseur syntaxique.
- un `Makefile` pour compiler.

Le travail ˜réaliser est :

1. Observer l'analyseur et comprendre son fonctionnement.
2. Compiler et exécuter les sources données.
3. Écrire les règles de typage pour les expressions (`exp` dans la syntaxe abstraite).
4. Programmer les règles de typage dans un fichier `exptyper.ml`. Pour cela, écrire une fonction :

   ```
   typer_exp :  (string * texp) list -> exp -> texp
   ```

   dont le premier argument est l'environnement de typage (liste associative), le deuxieme est l'expression à typer et le résultat est une des valeurs `Tint` ou `Tbool`.
5. Compiler et tester.