Pablo Rauzy <pr@up8.edu>
https://pablorauzy.fr/teaching/compil

Interprétation et compilation

TP 7 : Typage

Pour compiler le code fourni : `dune build test.exe`.

État des lieux :

- Le fichier `lexer.ml` s'occupe de l'analyse lexicale, n'y touchez pas.
- Le fichier `parser.ml` s'occupe de l'analyse syntaxique, n'y touchez pas.
- Le fichier `ast.ml` défini :
  - un sous-module `Syntax` qui représente l'AST en sortie de l'analyseur syntaxique, avec encore toutes les informations de position notamment ; et
  - un sous-module `IR` qui représente l'AST de notre représentation intermédiaire en sortie de l'analyse sémantique.

  Vous remarquerez que par rapport à ce qu'on a appris à interpréter et compiler jusque là, le langage est simplifié.
- Le fichier `baselib.ml` défini un environnement de typage de base qui contient les types des opérations primitives de notre langage.
- Le fichier `semantics.ml` devrait s'occuper de l'analyse sémantique mais ne fait rien d'intéressant en l'état puisqu'il se contente de faire la traduction de `Syntax` en `IR` sans faire aucune vérification. C'est sur ce fichier que vous allez devoir travailler.
- Le fichier `test.ml` est le point d'entrée de notre programme, il ouvre le fichier qu'on lui passe en premier argument le parse, puis le passe à l'analyseur sémantique et affiche la représentation intermédiaire obtenue en sortie.

Exercices : modifier le code du module `Semantics` pour gérer les erreurs des fichiers fournis dans le dossier `tests`.

Bonus : gérer le cas du warning "*unassigned variable*".