

Rapport du Projet de Compilation

Rémy Tuyéras

12 janvier 2009



1 Description générale

Le compilateur *petitcaml* du langage **Mini-MI** s'articule en les fichiers suivant :

`ast.mli` = contient l'Arbre de syntaxe abstraite pour le parser, le lexer et le typer.
`lexer.*` = fichiers constituant le *lexer*.
`parser.*` = fichiers constituant le *parser*.
`parse.[ml,mli]` = contient les fonctions utilisées par le parser pour traduire le sucre syntaxique.
`typer.ml` = fichier constituant le *typer* (auquel sont jointes des fonctions d'affichage).
`spellchecker.[ml,mli]` = fichier constituant le correcteur vers une syntaxe plus proche de l'assembleur mips, en particulier il crée les clôtures des fonctions.
`compiler.[mli,ml]` = fichier constituant le *compiler* qui traduit la correction vers le mips.
`mips.[mli,ml]` = contient l'Arbre de syntaxe abstraite du mips et les fonctions d'écritures dans le fichier de sortie.
`main.ml` = recolle tous les fichiers pour générer le compilateur.
`Makefile`, `.depend` = lance `main.ml`.
`test.ml` = fichier de test.

2 Améliorations

Le compilateur a reçu les améliorations suivantes par rapport à ce qui était demandé :

1. Sucre syntaxique `fun motif → expression` pour `function motif → expression` ;
2. Possibilité de mettre le symbole `;;` à la fin des déclarations ;
3. Sucre syntaxique `let motif = expression and motif = expression` pour `let (motif,motif) = (expression,expression)`.