



RUBIC

Le but de ce projet est de réaliser un compilateur d'un langage objet proche du langage Ruby. Ce travail sera à réaliser par groupe de 2. Le langage sera compilé en du code intermédiaire du compilateur LLVM. L'appel au compilateur LLVM permettra ainsi de finir la compilation en binaire et de générer du code efficace.

- [Description du langage Rubic](#)
- [Assembleur LLVM](#)
- Les [sources](#) des analyseurs lexicaux et syntaxiques à compléter. Ils sont fournis avec un Makefile fonctionnel, à adapter en fonction de vos besoins.

A Faire

La réalisation du compilateur donnera lieu à la remise d'un rapport décrivant ce qui a été réalisé (et ce qui ne l'a pas été) ainsi qu'à la remise du code source. Des fichiers de test devront par ailleurs faire partie de ces sources.

La date de remise du rapport et du code source doit être au plus tard le *xx janvier 2013*.

Le compilateur devra vérifier les types, et générer la représentation intermédiaire sous forme de fichier texte. Le rapport devra préciser tous les choix d'implémentation, ou de typage, ainsi que d'éventuelles limites à ce que le compilateur peut faire.

Une soutenance orale sera organisée, ou chaque groupe présentera en 5min ce que fait son compilateur Rubic.

Fonctionnalités facultatives

Un ensemble de fonctionnalités facultatives peut être implémenté:

- Des conditionnelles construites avec `unless`, des conditionnelles sur une ligne avec `if` et `unless` (voir syntaxe Ruby)
- L'implémentation de tableaux: on peut n'envisager que des tableaux constants, déclarés par `a = [val1, val2, ...]`
- La récursivité des fonctions

Evaluation

L'évaluation sera basée sur ces critères (dans cet ordre, du plus important au moins important):

- Le compilateur fonctionne pour l'ensemble des fonctionnalités décrites pour le langage Rubic et génère un code correct.
- La stabilité du compilateur (pas de plantage lors de la compilation)
- La qualité du rapport.
- Les fonctionnalités facultatives implémentées.