

Intro

Le TP consistait à réaliser, en utilisant Ocaml et les bibliothèques LLVM, un compilateur pour le langage VSL+ afin de générer du code 3 adresses. Plus précisément, on devait réaliser la vérification de type ainsi que la génération de code à partir d'un AST. En effet les analyseurs lexical et syntaxique de VSL+ étaient déjà fournis (c'est eux qui produisent la représentation sous forme d'AST d'un programme VSL+).

Description de la méthodologie de travail

La majeure partie du travail a été effectué en TP à deux sur le même ordinateur. A l'exception des expressions simples et du "IF" le code a été intégralement écrit par Rémy Sun. En effet sa maîtrise avancée du langage et sa compréhension du sujet permettait de progresser plus rapidement dans les différentes étapes. Par ailleurs, le travail personnel qu'il a fallu fournir en dehors des TP afin de boucler le projet à temps a été réalisé par Rémy.

Pour l'organisation du code, il y a été créé dans `codegen.ml` une fonction par type dans `ast.ml`. Et c'est dans `codegen.ml` qu'a été écrit l'ensemble du code.

Pour les tests, nous en avons créé pour chaque nouvelle fonctionnalité du compilateur. Puis, quand ceux-ci furent fonctionnels, nous avons exécutés les tests fournis.

Bilan de ce qui a été réalisé (complètement/partiellement)

Les expressions simples : Complètement réalisé.

Nous avons repris la structure pour l'addition et on l'a appliqué pour les autres expressions.

L'instruction d'affectation : Complètement réalisé

On stocke la valeur grâce à `Llvm.build_store`.

La gestion des blocs : Complètement réalisé

Ouverture d'un scope au début du block, on réalise la déclaration puis le statement et on ferme le scope.

La déclaration des variables : Complètement réalisé

Les expressions avec variables : Complètement réalisé

Les instructions de contrôle if, while et la séquence : Complètement réalisé

On crée un block pour le if où on évalue la condition puis soit on se branche sur le block du "then" ou alors du "else" puis ces deux blocs se branchent sur le "eblock" à la fin.

La définition et l'appel de fonctions (avec les prototypes) : Complètement réalisé

Les fonctions de la bibliothèque (PRINT et READ) : Complètement réalisé

La gestion des tableaux (declaration, expression, aectation et lecture) : Complètement réalisé