

Compte-rendu projet Compilation

Partie utilisateur

Compilation

make	Compile l'ensemble des fichiers .c, .y et .lex
On peut lancer make avec différentes options.	<p>clean : supprime tous les .o et garde uniquement l'exécutable</p> <p>mrproper : supprime tout ce que le make a créé</p> <p>copy : copie le projet dans un nouveau fichier (utile pour zip ou tar) zip : créé un .zip du projet tar : créé un .tar.gz du projet</p>

Exécution

Se situer dans le dossier /bin puis : ./tpcc < nom_fichier ou directement : ./bin/tpcc	Permet d'exécuter l'analyseur lexical sur le fichier entré en paramètre.
Le programme peut être lancé avec différentes options.	<p>-t ou --tree : affiche l'arbre de dérivation généré</p> <p>-s ou --syntabs : affiche la table de symbole générée</p> <p>-h ou --help : affiche le texte d'aide et quitte, pas d'autre sortie</p>

Test

sh Test.sh	Permet d'exécuter un banc de test sur l'analyseur lexical. Ces tests sont situés dans le répertoire Tests contenant un répertoire négatifs et positifs. Ces répertoires contiennent des fichiers .tpc. Après le test un compte-rendu est créé de nom resultat.log.
------------	--

Partie développeur

Sommaire

abstract-tree.c/abstract-tree.h	Fichiers contenant les différentes fonctions permettant de créer l'arbre de dérivation.
decl-var.c/decl-var.h	Fichiers contenant les différentes fonctions permettant de créer la table des symboles.
generate-code.c/generate-code.h	Fichiers contenant les différentes fonctions permettant de générer le code NASM. (En cours de développement)
lexer.lex	Analyseur lexical.
Option.c/Option.h	Fichiers contenant les différentes fonctions permettant de gérer les options en paramètre du programme.
parser.y	Analyseur syntaxique.
verif_type.c/verif_type.h	Fichiers contenant les différentes fonctions permettant de détecter les erreurs sémantiques. (Quelques cas non vérifiés)

Partie non terminée

On peut apercevoir dans generate-code.c que certains 'case' sont mis en commentaire, c'est simplement qu'ils ne sont pas faits ou finis. Toujours dans generate-code.c certaines fonctions portent en début de commentaire 'UNUSED', ce qui signifie que la fonction n'est pas utilisée car pas complètement implémentée.

La vérification des types ne fonctionne pas pour les arguments d'une fonction.

Erreurs

Problème avec les ASSERT provoquant une erreur portant la valeur 134 dans resultat.log, causé par les instructions Assign.

Globalement, generate-code n'est pas très fonctionnel (sauf pour faire des opérations, créer des fonctions, renvoyer une valeur...).