

RAPPORT : PROJET DE COMPILATION

Difficultés et potentielle solution apportées aux difficultés :

- Arbre abstrait :

Nous avons eu du mal à voir l'assignation type/token ainsi que le fonctionnement du type union dans le bison. Nous pensions aussi au départ que les règles vides ne devaient pas avoir d'action alors que ceux-ci sont là pour initialiser les nœuds.

- Partie nasm :

Nous avons eu énormément de mal à intégrer la partie de code nasm dans le programme donc nous avons décidé de nous concentrer sur la partie qui nous fera plus avancer dans le projet : les erreurs sémantiques.

- Mise en place des structures dans le tableau de symbole :

nous avons choisie d'afficher les champs des structures dans une table de symbole différente de celle de la table de symbole globale car cela serait plus simple de vérifier les erreurs liées aux structures.

- Erreurs sémantiques :

Nous avons eu des difficultés sur la gestion des erreurs sémantiques, certaines erreurs n'étaient pas identifiées par le programme et d'autres étaient identifiées alors qu'elles ne l'étaient pas. Pour résoudre ce problème, nous avons ajouté plus de précision sur les conditions dans la fonction ou sur les conditions d'utilisation des fonctions.

Nous avons décidé de regarder uniquement sur le nœud de gauche car notre grammaire nous permet de lire toutes les valeurs d'une opération. Nous sommes partis sur cette décision car nous pensions que regarder sur les 2 nœuds concernés dans une opération allait nous permettre de lire les types pour les 2 nœuds mais au final, sur les opérations compliquées, il y avait des erreurs là où il n'y en devrait pas (conflit de types alors que toutes les valeurs étaient de même type). Nous n'avons pas pu faire en sorte d'envoyer un warning dans le cas où on essaye d'assigner une valeur de type int à une valeur de type char.

Sur l'erreur sémantique du nombre d'arguments insuffisant dans un appel de fonction, nous avons voulu compter le nombre de paramètres durant l'initialisation de la fonction, l'intégrer au nombre de paramètres dans la structure *STentry* de la fonction et la comparer au nombre d'arguments trouvés dans son appel, mais on a fait l'erreur de ré-initialiser le compteur du nombre de paramètres à 0 à chaque fois qu'on sort de la grammaire, et cela nous a coûté du temps à trouver le soucis. On a fini par trouver le problème et on a arrêté de ré-initialiser.