

Universidade de Évora

Compilador para linguagem *Ya!*

Compiladores

6 de Junho de 2019



Autores:

Diogo Rafael, 37859

Vasco Crespo, 37913

Professor:

Pedro Patinho

Introdução

Este trabalho tem como objetivo a realização e implementação de um compilador para a linguagem *Ya!* bem como a implementação de um gerador de código MIPS.

Para o compilador iremos criar um analisador lexical, sintático e semântico. O programa receberá um ficheiro com código na linguagem *Ya!* e irá devolver um ficheiro contendo código MIPS, irá criar uma Symbol Table e representá-la tal como a APT referente ao código.

Análise Lexical

Para a análise lexical estão definidos todos os tipos de variáveis possíveis e existentes na linguagem *Ya!*, estas sendo: Int, Float, Id, String, Bool, Int_LIT, Float_LIT, String_LIT, Bool_Lit) tal como as palavras específicas relativas a esta linguagem, estas sendo: mod, if, else, return, void.

Implementámos a análise lexical tendo em conta também todas as exceções desta linguagem.

Análise Sintática e Semântica

Para a análise sintática toda a gramática do código é analisada e caso esteja correta e não hajam erros, são feitos os nós da APT. Entretanto à medida que é feita a APT também é feita a análise semântica e a Symbol Table, adicionando as funções, as variáveis e os seus símbolos à tabela.

A APT após ser gerada é apresentada no terminal ao executar o programa tal como a Symbol Table. A apt.h contém todas as structs necessárias que representam várias expressões, tal como as condições, declarações, statements, etc.

Por outro lado, a Symbol Table foi implementada através de hashtables para que não haja conflitos entre as diferentes funções e respetivas variáveis.

Conclusão

Neste trabalho apenas conseguimos implementar a análise lexical, a análise semântica, a análise sintática e a APT e a Symbol Table. Tentámos implementar também o gerador de código para MIPS mas encontramos alguns problemas pelo que não conseguimos concluir.