CS 327
IIT Gandhinagar

SeeSPIM Compiler

Project Presentation

Project Goals

Designing a Compiler in C that can run on simulators like SPIM & MARS

Aditya Tripathi	18110010
Devvrat Joshi	18110076
Kushagra Sharma	18110091
Nishikant Parmar	18110108

Thanks to Prof. Bireswar Das

Basic Features

- Data Types
- Declaration of Variables
- Conditional Statements
- Relational Operators
- Boolean Operators
- Unary Operators
- Arithmetic Operators
- Assignment Operator
- For Loop
- While Loop
- Control Flow Statements
- Scope of Variables
- Function Declaration
- Function Calling
- Input Function
- Output Function

Advanced Features

- Advanced Control Flow Statements
  - break
  - continue
- Recursive Functions and Dynamic Allocation
- Extensive Error Handling
  - Break, continue, return type, variable or function declaration, input/output type checking, assignment type checking, array indices - integer, function calling, type checking - arithmetic operators, comparison operators, main function not declared, params passing errors, parse errors, total errors.
  - All errors are printed using specific message with line number and position.
- Extra Data Types (Strings)
  - String declaration
  - Assignment (Right side static string or another string variable)
  - Mutation (Changing the ith index of the string)
  - Concatenation of two strings using + operator
  - String input and output
- Single Dimensional Array
  - Can use any generic expressions for indexing
- Multi Dimensional Array
  - Static integer for indexing
- Newline Function
  - Function that prints a newline when called
- Parameter passing to functions
- Ternary Operators
- Complete scope maintenance

Techniques and New Ideas

- Stack allocation is used for memory allocation. A stack pointer and a top of stack pointer are maintained throughout the assembly generation for each activation record

- Parameter passing is done by placing the parameters on the activation record of the callee function, by the caller function. The return address is automatically linked to $ra via jal.

- Strings defined in any function gets globally defined in .data section of mips code.

- Assigning/copying a string is done by looping over the length of the strings.

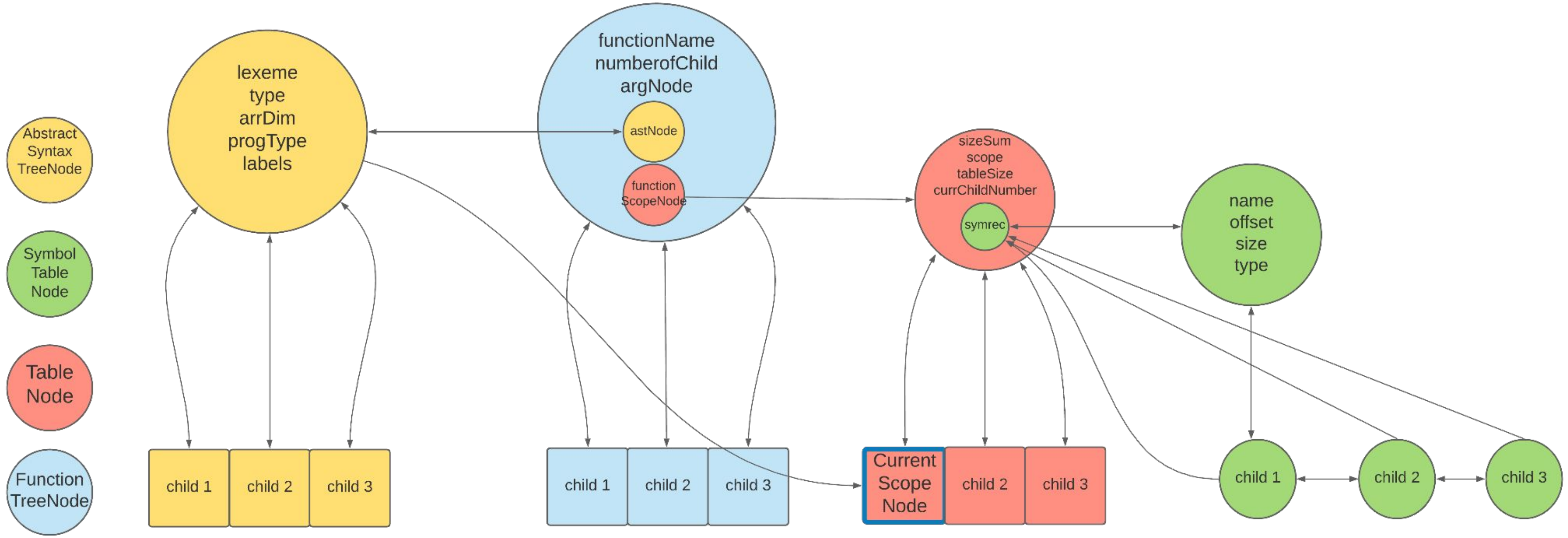
- For multi-dimensional array the array index is flattened and the array is treated like a static one dimensional array.

- Type checking: Types are checked using their entry in symbol table for assignment, parameter passing, etc.

- Error handling is done while parsing. This includes type checking, declaration checking etc.

- Very basic register allocation is done. The registers are spilled into memory after any computation.

- Scope is maintained using a data structure storing separate symbol tables for each scope.
- The scope can change when:
  - A function starts/ends
  - A loop starts/ends
  - if-else statements start/end
  - Declaring blocks using **{ }**
- The scope checking would handle the accessibility of the string. All the constant strings are assigned a name and stored in .data section.



Thank You

References -
https://github.com/amankr/Mini-Compiler
For Logos and Graphic Icons used in slides - Google Images