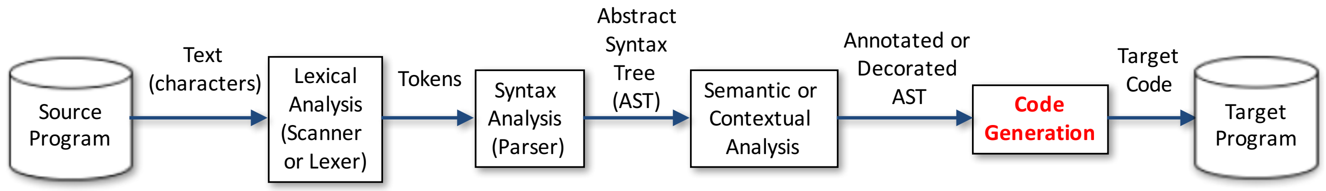
Ollscoil Teicneolaíochta na Mumhan
Munster Technological University

Programming Language Design
Department of Computer Science
Dr. Francisco Ortin
francisco.ortin@mtu.ie

Laboratory 10 – Runtime Environment

The objective of this laboratory is to annotate the AST with the offsets for global and local variables and parameters. An offset is an integer number indicating the relative memory address of one variable. The compiler statically sets the relative memory address used to store that variable at runtime (dynamically). This is part of the code generation phase.


Source Program
Text (characters)
Lexical Analysis (Scanner or Lexer)
Tokens
Syntax Analysis (Parser)
Abstract Syntax Tree (AST)
Semantic or Contextual Analysis
Annotated or Decorated AST
Code Generation
Target Code
Target Program

Offset Computation

Remember, in MAPL, characters, integers and real numbers are stored in, respectively, 1, 2 and 4 bytes.

Analyze the high-level code provided (input.txt) and its corresponding assembly code (output.txt) to figure out how offsets of variables and parameters are computed. Write down the equation that the compiler uses to compute the offsets of global, local and parameter variables (explained in lectures).

Go ahead

- Implement an `OffsetVisitor` in the code generation phase to compute the offsets of global, local and parameter variables. `VarDefinition` nodes in the AST should be decorated with an integer `offset` attribute (field). Notice that `VarDefinition` must also have a scope field indicating if they are global (scope = 0) or local (scope = 1) variables—parameters are considered local variables. The scope value in `VarDefinition` should have already been assigned in the `IdentificationVisitor` traversal in the semantic analysis.
- Make sure the AST is correctly annotated using Introspector. After semantic analysis, traverse the AST with your `OffsetVisitor` to compute `offsets` of `VarDefinitions`. Use Introspector to compare the actual offset values in the AST with the expected ones. You can use the input.txt and output.txt examples provided.