

Group Members:

Alexander Benson (bensonalec)
Jacob Garcia (jdcia)
Ole Jeger Hoffstuen (ohunter)
Keaton Jones (kjones4648)

Compiler Pipeline:

Frontend
 Preprocessor
 Tokenizer/Lexer
 Parser
Optimizer
 TBD
Backend
 TBD

Language Choice:

Python 3.7

Infrastructure:

We decided to use Continuous Integration for the purpose of gaining experience with it, and ensuring that we always have a working, tested build on our master branch. For CI, we use CircleCI in order to run a docker container that performs our Python unit tests.

Documentation:

Maybe Sphinx?

Preprocessor

TDB

Tokenizer/Lexer

For the tokenizer, we decided to use the library “rply”, which is a python implementation of lex and yacc. Using this, we have tokenized (what we believe to be) the entirety of the C language.

Parser

For the parser, we also used “rply”. We have implemented the following features:

- Function Definition (without arguments)
- Return statements
- Function calls with parameters of the form/type:
 - Integer
 - Arithmetic Expression ($x + y$)

At this time, we can handle the given “bare minimum” c program.