1 Introduction - due in 30/10/2020

1.1 Problem Statement

In the world of computer science, and computer programming in general, there is an abundance of learning materials freely available from which students can learn how to program, how to use a certain programming language, how to use a framework for that language, etc.. One area of study that is not as well developed in terms of learning materials is compiler design. It would be unfair to say that there is nothing for a beginner to make use of, however these resources have a tendency to ignore or gloss over the gory details, advising that tools are used to generate complex parts of the compiler for them.

This is good advice, and indeed in a "real world" situation an organisation that wanted to create a compiler would use these tools as the popular ones [TODO: do reference here, LLVM is used by Apple] are tried and tested, and simplify the compiler creation process immensely. However, if a learner wished to understand what these tools did, they would need to either read the source code of these tools (this would be difficult, as some of the projects are massive and very complex.), or read books of a technical nature, such as the Dragon Book. [TODO: write proper name of book, say henceforth referred to as the dragon book.] While the Dragon book is an excellent resource for designing compilers, it is too dense for a learner who wants to apply any knowledge s/he can glean from it.

This is the problem. There is an absence of material that shows the learner that they can create a compiler with only their own code, and without becoming a computer science professor first!

In order to rectify this problem, I propose the following:

1.2 Product Description

The product that I will produce at the end of this project is a compiler for a general purpose programming language of my own creation. The compiler will be able to compile a program written in said programming language into assembly language targeting the x86 CPU architecture. The compiler will be written in a modular, easily understandable way using the Java programming language. It will be written in this way to encourage learning about the various stages of compilation.

1.3 Users or Audience

The users that I imagine would be interested in this project would be anyone who is interested in the inner workings of a compiler, and how they could go about creating their own compiler without leaning on other available tools. As compiler design is quite a technical subject, I would expect that the users of this project would be at university level or equivalent in terms of their education in computer science or programmming.

2 Background Research - due in 7/12/2020

2.1 Necessary Background Material

2.2 Related Work

2.3 Professional, Legal, Ethical & Social Issues

3 MISC

3.1 comments

Inline comments within the source code will be kept to a minimum, this is done to prevent noise when reading the source code. An overview of the structure of the program will be stored in the README.org file found at the root of the project.