

CIS4650 Checkpoint 1 Documentation

Team: Christian Foote, Drew Mainprize

What has been completed:

For this checkpoint we have implemented a scanner and parser for the C- language, using the provided SampleParser for the Tiny language as a baseline.

The scanner reads from a provided .cm file, outputting tokens as described in the C- specification, and displaying an error when an unrecognized character is read, along with the line and column it appeared on.

The parser takes the tokens output by the scanner to check its grammar against that defined by the C- language, as well as building the Abstract Syntax Trees representing the input file.

Lessons learned:

Through this assignment we have learnt to emphasize following the provided implementation steps, while we will come across parts that we have difficulty with, that can be when it is most important to keep at that part and make sure it is functioning, rather than jump forward to a part that we assume to be easier, as that may create more difficulties later as we go back, and then will have to adjust later to fit our current understanding, and adjust based on any changes that may have been made. Jumping

ahead may feel better in the moment but it significantly increases the chances for mistakes to be made.

We also learnt the importance of communicating with each other as we worked, ensuring that the other was aware of what we were working on, what we had completed, and any changes that we made, for the sake of avoiding issues such as both working on the same problem at the same time, making changes that the other is unaware of, and other things that increase the number of bugs present within the program.

The final major lesson we learnt was the importance of using the tools at our disposal such as Git to mitigate issues such as those discussed above, allowing us to work from the area, without worrying if we had the most up to date version of our files, as well as being able to look through what work the other had done to see if any changes were made that affected our work.

Assumptions:

Limitations:

At the moment the parser and scanner does not have error checking implemented, so we must assume that any file passed to it is error free, or has been checked and verified by hand

Potential Improvements:

Fixing error reporting by the parser, currently unrecognized tokens are reported with the line number off by 1, due to the indexing by jflex starting at 0, while text editors start at line 1.

Individual Contributions:

Drew:

- Abstract syntax classes
- CUP grammar actions

Christian:

- Flex regexes
- CUP symbols, precedences, and grammar
- Writing documentation