Project #2
CS 3510 – Spring 2014
Seth Yost, Mitch Birti

I. Requirements:
Develop a parser which can parse a program written in the C- language described in Appendix A of the text.

II. Design:
We built a recursive descent parser as described in class and in the book, using the scanner developed in project 1A/1B.

III. Implementation:
We made an object for each language structure (variable declarations, expressions, loops, etc.) and wrote parse methods to parse those structures using the first and follow sets of the grammar.

IV. Testing:
We wrote a Tester class that outputs the tree generated by the parser when it is given a test file. We tested using a program given in the book and with 3 test programs from the course website.

V. Summary/Conclusion:
Our project is working properly. We didn't encounter any major problems with our Parser implementation, which was extremely surprising. We did have a bit of trouble with our grammar not being up to specifications, and we were missing some tokens in our first and follow sets. This was resolved when we tried to parse structures that depended on those first/follow set tokens.