Assignment 1 Report

In dealing with Java’s reserved words, which are all lowercase, instead of rewriting the code from scratch, we simply removed the toLowerCase() in JavaWordToken class after RESERVED\_WORDS.contains(text). With this simple change, we made all keyword case-sensitive.

In order to let Java scanner skip Java comments, including single line comments and multi-line comments, as whitespaces properly, we added a new method peekChar() in Scanner.java, which is the superclass of JavaScanner.java. This method enables JavaScanner to see the next char after current without consuming it, so that we can determine if this is the beginning of a comment easily.

We also updated the reserved word tokens and special symbol tokens in the JavaTokenType.java because there are differences of these tokens between pascal and java.

We added a new class JavaCharacterToken.java to handle consuming java single character. By implementing this class, all the single character including ‘\’’, ‘\n’, ‘\t’ and ‘\\’ will be recognized.

To handle Java’s string, we first have the JavaScanner.java class to recognized the start of the string by consuming double-quote(“), which is different from the pascal string of single quote(‘). We compared the difference of pascal string and java string and made changes in the JavaStringToken.java. We deal with the issue of empty string(“”), (\n), (\t), (\”) and (\\) inside a string.