Compiler Project 1 “Scanner” Report
2013-11432 Cho Sung Min

1. Tokens
The following are possible types of tokens :
[tCharacter, tString, tIdent, tNumber, tBoolean, tBaseType, tFactOp, tTermOp, tRelOp, tNot, tLBrak, tRBrak, tLSqrBrak, tRSqrBrak, tAssign, tSemicolon, tColon, tComma, tDot, tModule, tBegin, tEnd, tIf, tThen, tElse, tWhile, tDo, tReturn, tVar, tProcedure, tFunction, tEOF, tIOError, tUndefined].

Most of tokens are read in a straightforward way. In below sections, I only explained how to read tokens that can be controversial.

2. Handling comments
If the scanner encounters “//“, it reads and ignores everything until it meets ‘\n” or EOF.

3. Handling character
If the scanner encounters a single quote(‘), it reads everything until it meets an unescaped single quote or EOF. If it meets an unescaped single quote, it checks whether the string between the quotes is a valid description of a character. If it is a valid description, it returns tCharacter token where token value is the described character itself. If it is not a valid character description, or it met EOF before meeting closing single quote, it returns tUndefined. Note that in the case of tUndefined, the scanner will save appropriate error message in the token value (ex: “Unmatched single quote ‘a “), so that the parser can later emit useful error messages.
Valid character description has to be either one printable ascii character (ascii code : 32 ~ 126), or one escaped character (one of the six escaped characters). Any string other than this is not a valid character description.

4. Handling string
This does the same thing as handling character, except the quotation mark is (“) instead of (‘).
Valid string description has to be a stream of printable ascii characters. Any string other than this is not a valid string description. Note that when the scanner saves token value for tString, it “unescapes” the string description. For example, if the string description contains (\)(n), they are merged into (\n).