Brian Cullinan

CS 481 HW2

9/8/08

* 1. LL is a top down parser for a subset of context-free grammers. It parses input from left to right and constructs a left more derivation of the sentence.
  2. LR reads from left to right and produces the right most derivation of the sentence.
  3. LARL is a special form of LR parser that can deal with more types of context free grammers.

1. LR can be implanted using recursive descent parsing.
2. LL(k) uses k tokens of look ahead when parsing a sentence.
3. A compiler compiler creates a parser, interprets, or compiler from some form of formal description.
   1. Yacc is available for Unix and it parses C programming language. It is also available for the languages Ratfor, EFL, ML, Ada, Java, and Limbo. Lex is also available for Unix
   2. Flex is a lexical anylizer written in C. Bison is a LARL parser. It converts grammers into C or C++ code.
   3. JavaCC is an open source parser generator for Java. It produces a formal grammer provided in EBNF. It generates top down parsers in LL. It is licensed under BSD.
   4. GOLD uses LARL and DFA tables. It supports the languages Assembly - Intel x86, ANSI C, C#, D, Delphi, Java, Pascal, Python, Visual Basic, Visual Basic .NET, Visual C++.
4. * 1. /\*\*\* Definition section \*\*\*/
     2. %{
     3. /\* C code to be copied verbatim \*/
     4. #include <stdio.h>
     5. %}
     6. /\* This tells flex to read only one input file \*/
     7. %option noyywrap
     8. %%
     9. /\*\*\* Rules section \*\*\*/
     10. /\* [0-9]+ matches a string of one or more digits \*/
     11. [0-9]+ {
     12. /\* yytext is a string containing the matched text. \*/
     13. printf("Saw an integer: %s\n", yytext);
     14. }
     15. . { /\* Ignore all other characters. \*/ }
     16. %%
     17. /\*\*\* C Code section \*\*\*/
     18. int main(void)
     19. {
     20. /\* Call the lexer, then quit. \*/
     21. yylex();
     22. return 0;
     23. }