Assignment 2: Expression Parser

1 Implementation

You are to generate a lexer and parser for a small expression language using JavaCC and JTB. You should then write a Java class `Parser` that reads a file in the given language from standard input (`System.in`) and prints it to standard output (`System.out`). You are to use JTB to implement a pretty printer that will print the binary parse tree using indentation to indicate depth. If the file does not parse, the program should print "Parse error." followed by newline to standard error (`System.err`). Note that in order for the pretty printer to work, you must name your terminals and non-terminals in the grammar appropriately.

2 Informal Syntax Specification

The language consists of base-4 numbers (i.e. only the characters 0, 1, 2 and 3 can be used to construct numbers), the binary infix operators P, M, S and T with precedence in that order (P lowest, T highest) and parentheses "(" and ")" for grouping of expressions. P and M are left associative (hint: use n-ary operator for internal representation of binary left associative operations), S is not associative and T is right associative. Spaces, tabs and newline characters must be treated as whitespace.

3 Examples

- 32 ⇒ 32
- 1 M 2 M 3 ⇒ [1 M 2] M 3
- 33 P 3 M 32 ⇒ 33 P [3 M 32]
- (33 P 3) M 32 ⇒ [33 P 3] M 32
- 33 P 3 P 32 ⇒ [33 P 3] P 32
- 3 T 2 T 1 ⇒ 3 T [2 T 1]

- 3 S 3 S 1 ⇒ error
- 3 1 ⇒ error
- 62 ⇒ error

The tree “[1 M 2] M [3 T [0 S 1]]” should be printed by your pretty printer exactly as follows:

```
M
  M
    1
    2
  T
    3
    S
      0
      1
```

4 Submission

You must submit the implementations to your subversion repository to the directory `comp3351/f2007/$USER/P2/`. Include only the original grammar, the Java Parser implementation, instrumented pretty printer and build script. The files must be called

- `language.jj`
- `Makefile`
- `src/edu/du/cs/comp3351/p2/PrettyPrinter.java`

You must check that the submitted code compiles by invoking `make`. Verify that the output of your program matches the expected output using both the provided examples and your own testcases.

You will not get any points if your submission does not compile without modifications or if your submission is in the wrong directory. Make sure to use “P2” for the directory name and **not** “p2”.