Jared Harrington

EECS 510 Final Project

For this project I decided to implement a pushdown automaton(PDA) to accept strings that represent basic arithmetic expressions. Specifically, the language only accepts strings over the alphabet $\sum = \{(,), +, -, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$. Some limitations of the grammar are that it can only accept single digit numbers and can only do basic addition and subtraction.

The context-free grammar for the language is as follows:

S -> E

E-> E+T | E-T | T

T -> (E) | D

D -> 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

For a more detailed view of the handling of the grammar q0 is the start state which can accept a “(“ and remain in the starting state or any number (0-9) and transition to state q1. When a “(“ is read an X is pushed onto the stack of the PDA. When in the number state q1 it accepts “+” or “-“ and moves to the operator state q2. If a “)” is read the X on the stack is consumed and a lambda is pushed onto the stack to check to make sure all parentheses are closed. When in the operator state q2 it accepts a digit and goes back to q1 or accepts a “(“ and stays in q2.