CIIC 4030/ICOM 3046 Programming Languages

Exam #1 (Take Home)

Due: March 19, 2019 @ 11:59pm

The goal of this assignment is to implement a parser program in **Racket** to determine if code written in a functional language named CICOM is syntactically correct.

**CICOM Definition**

**Tokens**

Character ::= a-z | A-Z | ? | \_

Digit ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |9

Delimiter ::= ( | ) | [ | ] | , | ;

Operator ::= "+" | - | ~ | "\*" | / | = | != | < | > | <= | >= | & | "|" | :=

**Grammar**

Exp ::= Term { Binop Exp }

| if Exp then Exp else Exp

| let Def+ in Exp

| map IdList to Exp

Term ::= Unop Term

| Factor { ( ExpList ) }

| Empty

| Int

| Bool

Factor ::= ( Exp ) | Prim | Id

ExpList ::= { PropExpList }

PropExpList ::= Exp | Exp , PropExpList

IdList ::= { PropIdList }

PropIdList ::= Id | Id , PropIdList

Def ::= Id := Exp ;

Empty ::= empty

Bool ::= true | false

Unop ::= Sign | ~

Sign ::= "+" | -

Binop ::= Sign | "\*" | / | = | != | < | > | <= | >= | & | "|"

Prim ::= number? | function? | list? | empty? | cons? | cons | first | rest | arity

Id ::= Character {Character | Digit}\*

Int ::= Digit+