**Kennesaw State University**

**College of Computing and Software Engineering**

**DEPARTMENT OF COMPUTER SCIENCE**

CS 4408 Section (01) Concepts of Programming Language – (Spring 18)

**Project Deliverable No. 2**

Karim Rattani, Beenish Zafar  
[krattani@students.kennesaw.edu](mailto:krattani@students.kennesaw.edu)

March 21, 2018

# Initial Problem Statement:

To create a Syntax Analyzer(parser) for subset of LUA Language

# Detailed description of the solution:

Test program creates an object Parser, which calls the scanner to get token list. If getting token list, it goes through each token and match it against the grammar. If something doesn’t match, then it throws an illegalArgumentException error with the line number, what’s found, and what’s expected. If there’re no errors, then it outputs Parse Successful. The parser keeps processing the lines until it gets EOF token.

# Output:

**LUA Source Code:**

function f ( )

a = 5

b = 6

c = 9

d = 10

print ( a )

end

function f ( )

a = 5

a = + a 1

a = - a 1

a = \* a 2

a = / a 2

print ( a )

end

function f ( )

a = 5

b = 6

c = 9

d = 10

if == a 5 then

b = 6

else

c = 9

end

end

function f ( )

a = 5

b = 6

c = 9

d = 10

while == a 5 do

b = + b 6

end

end

function f ( )

a = 6

b = a

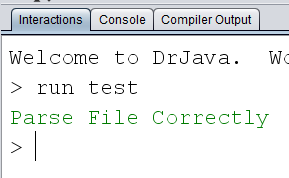
repeat b = + b a

until == b 36

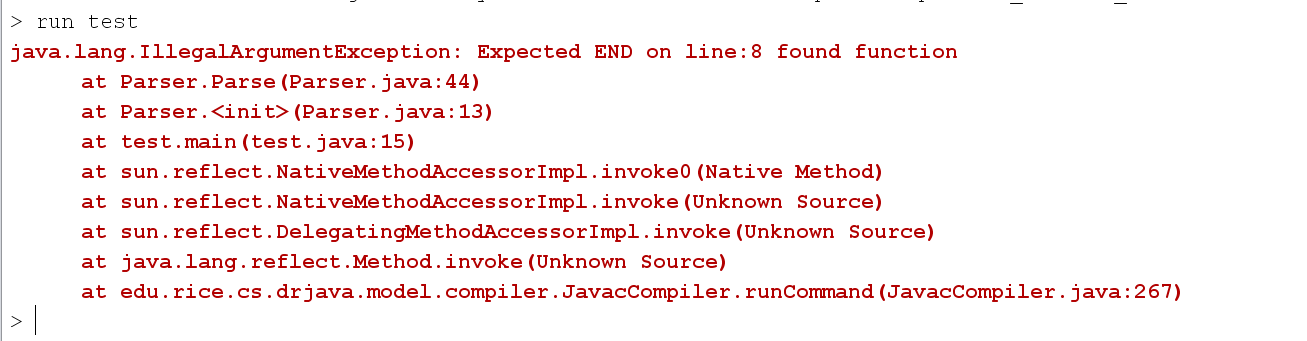
end

**Test File Output:**

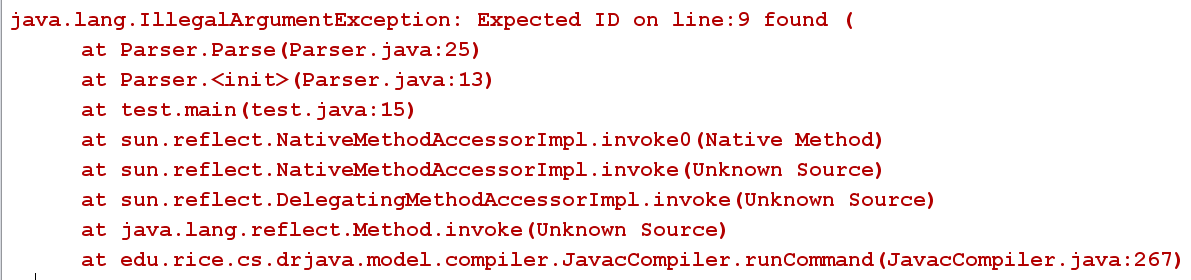
**Program following the grammar:**

****

**Program has an ‘end’ missing at the end of the function:**

****

**Program missing an ‘id’ after function declaration:**

****