https://ruslanspivak.com/lsbasi-part1/

$ python genptdot.py "14 + 2 \* 3 - 6 / 2" > parsetree.dot && dot -Tpng -o parsetree.png parsetree.dot

$ python genptdot.py > parsetree.dot && dot -Tpng -o parsetree.png parsetree.dot

$ python genastdot.py nestedscopes02.pas > ast.dot && dot -Tpng -o ast.png ast.dot

VSC : env to add path

"configurations": [

{

"name": "Python: Current File (Integrated Terminal)",

"type": "python",

"request": "launch",

"program": "${file}",

"console": "integratedTerminal",

"env": {"PYTHONPATH":"${workspaceRoot}"},

},

Token Class - contains a list of program tokens

Lexer Class - breaking a sentence apart into tokens one character at a time;

Parser Class -

AST Class -

SymbolTable Class -

SemanticAnalyzer Class -

NodeVisitor Class -

def main():

text = open('src/test/test\_14/nested\_function\_01\_14.pas', 'r').read()

lexer = Lexer(text)

parser = Parser(lexer)

tree = parser.parse()

semantic\_analyzer = SemanticAnalyzer()

try:

semantic\_analyzer.visit(tree)

except SystemError as sys\_error:

print(sys\_error)

for token in lexer.tokens:

print(token)

if \_\_name\_\_ == '\_\_main\_\_':

main()