German University in Cairo
Media Engineering and Technology
Assoc. Prof. Haythem Ismail

Compilers Lab, Spring term 2019

Milestone 1

Grammar for Nim in ANTLR

Please read the following instructions carefully:

- Read Rules & regulations first
- It is **YOUR responsibility** to ensure that you have:
  - Submitted before the deadline (20th of April).
  - Submitted the correct file(s).
  - Submitted the correct file(s) names.
  - Submitted the correct output format that matches each task.
  - Submitted correct logic of the task as it will be tested both publicly & privately.
  - Submitted your code in the format TeamName_milestone_2.zip where TeamName is your team name for example compteam_milestone_2.zip google form link https://forms.gle/M1TmtooVa65XmyAa8.

  .
- Good luck! =D

1 Grammar for Nim using ANTLR

In this part, you are required to implement the grammar for the language “Nim” using ANTLR4.

- Follow the exact file name:
- “milestone_2.py”, should contain the code to tokenize the the input given.
- “milestone_2.g4”, should contain the lexer rules for the grammar.
- “milestone_2_result.txt”, should contain the parse tree.
- The ANTLR file (.g4) should contain the regular definitions needed for Tokenization.
- You should submit all files contains your python code for the solution.
- All files should have the extension “.py” & the “main” method should be in a file with the correct name.
- You should make sure that the output is produced in a text file with correct name.

1. Write the grammar for the language Nim.
   Follow Nim's manual https://nim-lang.org/docs/manual.html#lexical-analysis to guide you in covering the implementation of the grammar & the test cases provided on the MET website.

   For example, the .g4 should contain the grammar to parse the input:

```
// test.g4 file
grammar test;
...
start : stmt;
stmt : complexOrSimpleStmt;
complexOrSimpleStmt : VARIABLE variable;
...
```

   The Python file should contain the code to output whether the input is "valid" or "invalid".

   Given the input "var x, y = 3" as follows:

```
var x, y = 3
```

   Then the output file will be:

```
valid
```