VIETNAM NATIONAL UNIVERSITY - HO CHI MINH CITY
HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING

PRINCIPLES OF PROGRAMMING LANGUAGES - CO3005

ASSIGNMENT 1

Recognizer

HO CHI MINH CITY, 01/2023

ASSIGNMENT 1

Recognizer

After completing this assignment, you will be able to

- define formally lexicon of a programming language.
- use ANTLR to implement a lexer for a programming language.
- define formally grammar of a programming language.
- use ANTLR to implement a recognizer for a programming language.

1 Specification

In this assignment, you are required to write a lexer and a recognizer for a program written in MT22. To complete this assignment, you need to:

- Install Python 3 if you have not installed it yet.
- Download initial.zip and unzip it.
- Download antlr-4.9.2-complete.jar from https://www.antlr.org/download.html, set the environment variable **ANTLR_JAR** to this file; install antlr4-python3-runtime version 4.9.2 (see instructions in section Python Targets of the above webpage).
- Remove all files in folders initial/src/main/mt22/utils, initial/src/main/mt22/astgen, initial/src/main/mt22/checker if any.
- Delete files initial/src/test/ASTGenSuite.py, initial/src/test/CheckerSuite.py, and initial/src/test/CodeGenSuite.py
- Comment out five lines 11-15 and from line 103-end of file initial/src/test/TestUtils.py and test the initial code again with just three following instructions in README
- Change folder initial into assignment1

To complete this assignment, you need to:

- read carefully the specification of language.
- modify **MT22.g4** in the initial code to describe formally MT22 language. **Please fill in your id in the header of this file.**

This assignment is divided two phases: lexer phase and recognizer phase. **These phases are assessed independently**.

1.1 Phase 1: Lexer

In this phase, you are required to write a lexer for a program written in ANTLR. To complete this phase, you need to:

- Modify MT22.g4 to detect tokens in MT22 language.
- Make 100 testcases for LexerSuite to test your code.
- For lexical errors, please return the following tokens together with specific lexemes:
  - **ERROR_TOKEN** with <unrecognized char> lexeme: when the lexer detects an unrecognized character.
  - **UNCLOSE_STRING** with <unclosed string> lexeme: when the lexer detects an unterminated string. The <unclosed string> lexeme does not include the opening quote.
  - **ILLEGAL_ESCAPE** with <wrong string> lexeme: when the lexer detects an illegal escape in string. The wrong string is from the beginning of the string (without the opening quote) to the illegal escape.
- You can assume that there is only one error in each test case.

1.2 Phase 2: Recognizer

In this phase, you are required to write a recognizer for a program written in MT22. To complete this phase, you need to:

- Modify MT22.g4.
- Make 100 testcases for ParserSuite to test your code.
- You can assume that there is at most one error in each test case.

2 Requirements

Run `python run.py gen` to generate the following files in folder **target** or its subfolder. Submit these files together with MT22.g4, LexerSuite.py and ParserSuite.py (10 files).

DO NOT COMPRESS THEM when submitting. Then click on Check button to check your submission. There are some simple testcases to check your submission which is marked later. Students can submit as many times as they want, but only the final submission will be graded. Because the system cannot bear the load when too many students submit their work at once,

students should submit their work as soon as possible. Students will take all responsibility for their risk if they submit their work near the deadline. When the submission deadline is over, the system will close so students will not be able to submit their work any more. Other methods for submission will not be accepted.

The deadline of both phases of assignment 1 is announced in the class website. You must complete the assignment by yourself and do not let your work seen by someone else, otherwise, you will be punished by the university rule for plagiarism.

3 Other regulations

- Students must complete this assignment on their own and must prevent others from stealing their results. Otherwise, the student treat as cheating according to the regulations of the school for cheating.
- Any decision made by the teachers in charge of this assignment is the final decision.
- Students will not be given any testcases after grading, but will only be provided with information of the testcase design strategie and the distribution of the correct number of students according to each test case.