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 & 2

Lexer and Recognizer

HO CHI MINH CITY, 06/2021

ASSIGNMENT 1 & 2

Lexer and 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 D95. 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-4.8-complete.jar from https://www.antlr.org/download.html, set the environment variable **ANTLR_JAR** to this file; install antlr4-python3-runtime version 4.8 (see instructions in section Python Targets of the above webpage).
- Remove all files in folders initial/src/main/d95/utils, initial/src/main/d95/astgen, initial/src/main/d95/checker if any.
- Test the initial code again with just three following instructions in README
- Change folder initial into assignment

To complete this assignment, you need to:

- read carefully the specification of language.
- modify **D95.g4** in the initial code to describe formally D95 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 Assignment 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 D95.g4 to detect tokens in D95 language.
- Make 100 testcases for LexerSuite to test your code.
- For lexical errors, please return the following tokens together with specific lexemes:
  - **ERROR_CHAR** 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.
  - **UNTERMINATED_COMMENT** without any lexeme: when the detects an unterminated comment.
- You can assume that there is only one error in each test case.

1.2 Assignment 2: Recognizer

In this phase, you are required to write a recognizer for a program written in D95. To complete this phase, you need to:

- Modify D95.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

Note that you must NOT compress your files when submit them. You **MUST** submit three files D95.g4, LexerSuite.py and ParserSuite.py in BKeL.

The deadline of both phases of assignment 1 & 2 is given in the assessment tool. 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.