German University in Cairo
Department of Computer Science
Assoc. Prof. Haythem O. Ismail

CSEN 1002 Compilers Lab, Spring Term 2020
Task 6: First and Follow

Due: 03.04.2020 by 23:59

1 Objective

For this task you will implement the algorithms computing the functions *First* and *Follow*, introduced in Lecture 4 of CSEN1003, for the *variables* of a given context-free grammar. Recall that a CFG is a quadruple (V, Σ, R, S) where V and Σ are disjoint alphabets (respectively, containing *variables* and *terminals*), $R \subseteq V \times (V \cup \Sigma)^*$ is a set of *rules*, and $S \in V$ is the *start variable*.

2 Requirements

- *Only Java may be used for this task.*
- We make the following assumptions about input CFGs for simplicity.
  a) The set V of variables consists of upper-case English symbols.
  b) The start variable is the symbol S.
  c) The set Σ of terminals consists of lower-case English symbols other than "`e`".
  d) The letter "`e`" represents ε.
- You should construct a `CFG` object using an input string encoding a CFG. Afterwards, you should implement two instance methods, `First` and `Follow`, which are invoked on the CFG object and return a string encoding of the *First*, respectively the *Follow*, set of each variable of the grammar.
- A string encoding a CFG is a semi-colon-separated sequence of items. Each item represents a largest set of rules with the same left-hand side and is a comma-separated sequence of strings. The first string of each item is a member of V, representing the common left-hand side. The first string of the first item is S.
- For example, consider the CFG $(\{S, T, L\}, \{\texttt{i},\texttt{a},\texttt{b},\texttt{c},\texttt{d}\}, R, S)$, where R is given by the following productions.

$$\begin{array}{lcl} S & \longrightarrow & S\texttt{c}T \mid T \\ T & \longrightarrow & \texttt{a}S\texttt{b} \mid \texttt{ia}L\texttt{b} \mid \varepsilon \\ L & \longrightarrow & S\texttt{d}L \mid S \end{array}$$

This CFG will have the following string encoding.

```
S,ScT,T;T,aSb,iaLb,e;L,SdL,S
```

- The output of each of `First` and `Follow` is, similar to the input, a semi-colon-separated sequence of items, where each item is a comma-separated pair. The first element of each pair is a variable of the grammar and the second element is a string representing the *First* or, respectively, the *Follow* set of that variable. The symbols in these strings should appear in alphabetical order. (`$` always appears last.) For example, the result of calling `First` on the above CFG may have the following form

  ```
  S,acei;T,aei;L,acdei
  ```

  Similarly, the result of calling `Follow` may be as follows

  ```
  S,bcd$;T,bcd$;L,b
  ```

- In order to simplify the evaluation, use the following main method

```
////////////////////////////////////////////////////////////
public static void main(String[] args) {
    String input = "S,ScT,T;T,aSb,iaLb,e;L,SdL,S";
    CFG cfg = new CFG(input);
    String firstEncoding = cfg.First();
    String followEncoding = cfg.Follow();
    System.out.println("First: " + firstEncoding);
    System.out.println("Follow: " + followEncoding);
}
////////////////////////////////////////////////////////////
```

3 Evaluation

- Your implementation will be tested by running `First` and `Follow` on five CFGs.
- You get one point for each correct output; hence, a maximum of ten points.