



**National University of Sciences and Technology (NUST)**  
**School of Electrical Engineering and Computer Science**

## **Department of Computing**

**CS 354: Compiler Construction**

**Class: BSCS-5AB**

**Lab [9]: First and Follow Sets**

**Date: 6<sup>th</sup> Dec, 2018**

**Time: [09:00 – 11:50hrs & 14:00 – 16:50 hrs]**

**Instructor: Dr. Rabia Irfan**

**Lab Engineer: Mr. Azaz Farooq**



## **Lab [9]: First and Follow Sets**

### **Introduction**

The syntactic or the structural correctness of a program is checked in the syntax analysis phase of compilation. The structural properties of language constructs can be specified in different ways. Different styles of specifications are useful for different purposes. Syntax analysis or parsing, is needed to determine if the series of tokens given are appropriate in a language - that is, whether or not the sentence has the right shape/form.

### **Objectives**

1. Successful computation of First and Follow Sets, paving the ways for LL(1) parsing.

### **Tools/Software Requirement**

1. GCC on Linux or Windows platform

### **Description**

Predictive parsing is possible only for the class of LL(k) grammars, which are the context-free grammars for which there exists some positive integer k that allows a recursive descent parser to decide which production to use by examining only the next k tokens of input. The LL(k) grammars therefore exclude all ambiguous grammars, as well as all grammars that contain left recursion. Any context-free grammar can be transformed into an equivalent grammar that has no left recursion, but removal of left recursion does not always yield an LL(k) grammar. In our course, we shall implement LL(1) parsing which is widely used. Since LL(1) parsing needs the algorithmic computation of first and follow sets. In this lab, we shall write programs that produce first and follow sets by the input of LL(1) grammar productions.

For more details:

[https://en.wikipedia.org/wiki/LL\\_parser](https://en.wikipedia.org/wiki/LL_parser)

For the Generation of LL(1) parsing table:

<https://web.cs.wpi.edu/~kal/PLT/PLT4.3.html>



### Lab Tasks

1. Write a program that can compute the first sets, for an input grammar.
2. Write a program that can compute the follow sets, for an input grammar.
3. Make an LL(1) parsing table, using first and follow sets computed before.

Run your code on the following grammar (discussed in class)

$E \rightarrow TE'$   
 $E' \rightarrow +TE' \mid -TE' \mid \varepsilon$   
 $T \rightarrow FT'$   
 $T' \rightarrow *FT' \mid /FT' \mid \varepsilon$   
 $F \rightarrow (E) \mid \text{id} \mid \text{num}$

Nonterminal	Nullable	FIRST	FOLLOW
$E$	No	{(, id, num}	{\$, )}
$E'$	Yes	{+}	{\$, )}
$T$	No	{(, id, num}	{\$, ), +}
$T'$	Yes	{*}	{\$, ), +}
$F$	No	{(, id, num}	{*, \$, ), +}

You may choose to (slightly) modify the above grammar, if needed, to suit the requirements of our Java-like language.

### Deliverables

You are required to upload your task (Sources & PDF document) using the link created on LMS followed by a viva.