# Lab Four

John DeFalco

john.defalco1@marist.edu

October 2019

## Goal

Analyzing grammars to build a better compiler. Focusing on First and Follow sets.

## Examples from the Readings

## 1 Crafting a Compiler

Below are the examples listed on the requirements document for the lab from the *Crafting a Compiler* textbook.

### 1.1 Problem 4.9

Compute First and Follow sets for the nonterminals of the following grammar:

```
S   -> a S e
     | B
B   -> b B e
     | C
C   -> c C e
     | d
```

| Symbol | First Set | Follow Set |
|---|---|---|
| S | {aSe, B} | {B, bBe, cCe, d} |
| B | {bBe, C} | {d} |
| C | {d} | $\lambda$ |

Table 1.1: First and Follow Sets for Problem 4.9

## 1.2 Problem 5.10

Show the two distinct parse trees that can be contstructed from the string: "if expr then if expr then other else other". For each tree, explain the correspondence of then and else.

Grammar provided:

```
S       -> Stmt $
Stmt    -> if expr then Stmt else Stmt
         | if expr then Stmt
         | other
```

**Tree 1**

```
-S
--Stmt
---if
---expr
---then
---Stmt
----if
----expr
----then
----Stmt
-----other
---else
---Stmt
----other
--$
```

In this tree, the correspondence between then and else is both tokens preced the Stmt nonterminal, where another production rule is then applied to further break the tree down.

**Tree 2**

```
–S
––Stmt
––––if
––––expr
––––then
––––other
––––––if
––––––expr
––––––then
––––––Stmt
––––––––other
––––else
––––Stmt
––––––other
––$
```

In this tree, both then and else preceed the other token.

# 2 Dragon Book

Below is the example listed on the requirements document for the lab from the *Compilers (Dragon)* textbook.

## 2.1 Problem 4.4.3

Compute First and Follow for each of grammar provided.

Grammar provided:

```
S    –> S S +
      | S S *
      | a
```

| Symbol | First Set | Follow Set |
|---|---|---|
| S | {S S +, S S *, a} | $\lambda$ |

Table 2.1: First and Follow Sets for Problem 4.4.3