

Lab Four

Alex Smith

alex.smith1@Marist.edu

March 6, 2019

1 CRAFTING A COMPILER

1.1 EXERCISE 4.9

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

```
1 S → a S e
2   | B
3 B → b B e
4   | C
5 C → c C e
6   | d
```

$\text{FIRST}(S) = \{a, b, c, d\}$

$\text{FIRST}(B) = \{b, c, d\}$

$\text{FIRST}(C) = \{c, d\}$

$\text{FOLLOW}(S) = \{e\}$

$\text{FOLLOW}(B) = \{e\}$

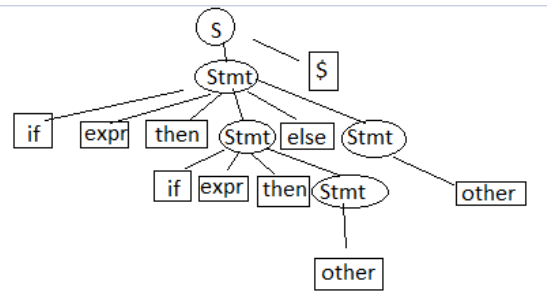
$\text{FOLLOW}(C) = \{e\}$

1.2 EXERCISE 5.10

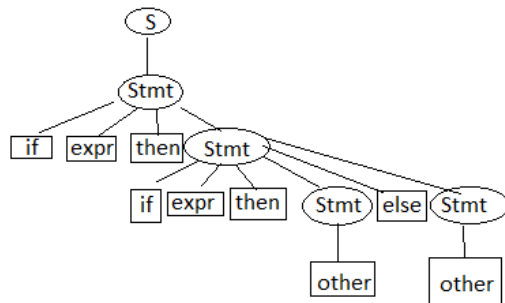
Show the two distinct parse trees that can be constructed for:

if *expr* **then** **if** *expr* **then** *other* **else** *other*

using the grammar given in Figure 5.17. For each parse tree, explain the correspondence of then and else.



In this parse tree, the "if expr then Stmt else Stmt" is put first. Since the Stmt after else is always going to be "other", we can only change the first Stmt. After then is going to be the next "if expr then Stmt".



In this parse tree, the "if expr then Stmt else Stmt" is put second. Since the given input has "if expr then" twice, we can put the else either first or second since it has tail recursion.

2 THE DRAGON BOOK

2.1 EXERCISE 4.4.3

Compute FIRST and FOLLOW for the grammar of Exercise 4.2.1.

$S \rightarrow SS + \mid SS * \mid a$

$\text{FIRST}(S) = \{a\}$

$\text{FOLLOW}(S) = \{a, +, *\}$