EDAN65: Compilers
Computer Science
Lund University

September 15, 2019

E06: LR parsing

E06-1: Consider the following context-free grammar

p_0: `start` → `program $`
p_1: `program` → `statement`
p_2: `statement` → `statement ";" statement`
p_3: `statement` → `ID "=" INT`
p_4: `statement` → ϵ

where p_0 is an extra start rule to handle end-of-file (`$` is the token for end-of-file).

Consider the string `;;` (the string consisting of two semi-colons). The grammar is ambiguous, which you can confirm by drawing two different parse trees for this string.

(a) Draw the two parse trees.

(b) Write down the sequence of shift/reduce/accept actions that an LR parser would take for constructing each of the two parse trees. For each action, show also the resulting stack and remaining input.