UFPI - CCN - DC
CIÊNCIA DA COMPUTAÇÃO
COMPILADORES - Prof. Dr. Raimundo Santos Moura

Atividade de Participação 2: Análise Sintática

José Augusto Oliveira da Silva Almeida
José Cassiano de Melo Junior

Teresina
2023

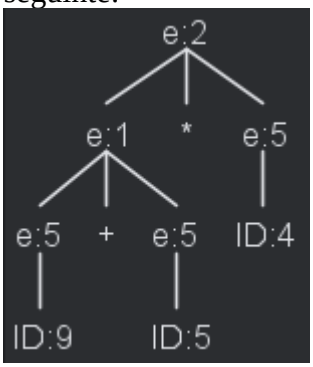
Sumário

1. Questões Práticas Sobre o ANTLR

1.1. Como o ANTLR trata o problema da ambiguidade para a gramática abaixo? Qual a *Parse Tree* gerada para a entrada: w = id + id * id?

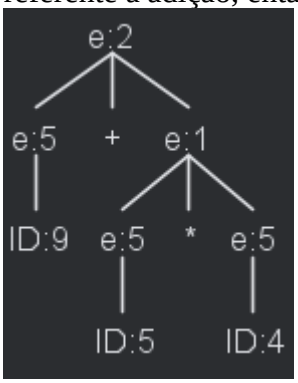
E --> E + E
| E * E
| - E
| (E)
| id

R – Segundo o próprio guia do ANTLR4, "ANTLR resolves the ambiguity by choosing the first alternative involved in the decision", ou seja, em uma situação de ambiguidade, ele sempre optará pela produção que vem primeiro na sequência. Dessa forma, no exemplo em questão, a prioridade será sempre o "E+E" em detrimento das produções seguintes. Assim, para a entrada em questão, a Parse Tree obtida seria a seguinte:



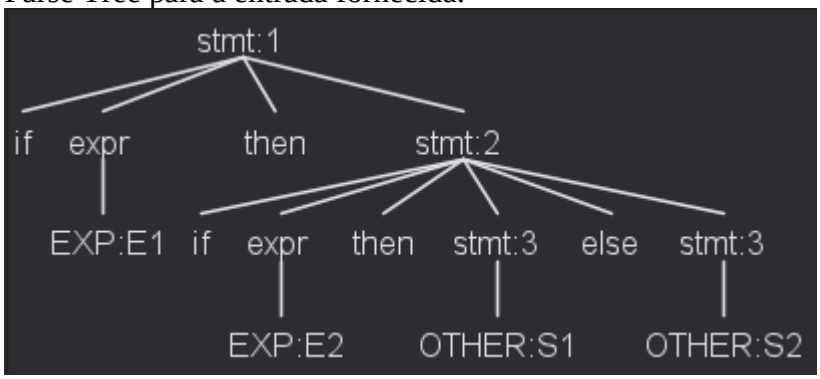
1.2. Na gramática da questão anterior, trocar a posição das regras: E --> E + E e E --> E * E, ou seja, colocar o operador de multiplicação antes do operador de adição. Neste caso, qual a Parse Tree gerada para a entrada: w = id + id * id, considerando a nova gramática?

R – Como agora a produção referente à multiplicação irá preceder a produção referente à adição, então ela terá prioridade. Desse modo, a Parse Tree gerada será:

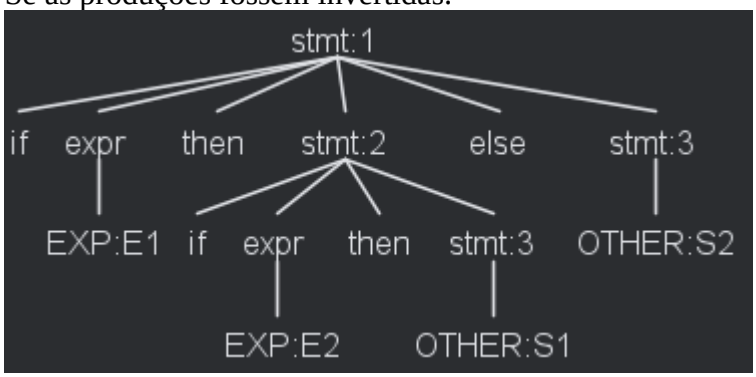


1.3.Considerando a gramática ambígua que gera comandos if-then-else,
stmt --> **if** expr **then** stmt
| **if** expr **then** stmt **else** stmt
| **other**
expr --> EXP
EXP --> 'E'[0-9]
Normalmente, nas linguagens de programação a regra é que o else **casa com o** then **mais próximo. Como o AntLR trata isso? Qual a *Parse Tree* gerada para a entrada: w = if E1 then if E2 then S1 else S2?**

R – Considerando que o ANTLR prioriza as produções em sequência, então sempre que a produção com o "else" for invocada, isso implicará no casamento com o "if" mais próximo. Por outro lado, se a primeira e a segunda regra de produção fossem invertidas, então, caso houvesse dois "if's", o "else" casaria com o mais distante.
Parse Tree para a entrada fornecida:



Se as produções fossem invertidas:



1.4.Escrever uma gramática no AntLR para resolver o problema da ambiguidade.

R – O próprio ANTLR já se encarrega de resolver os problemas de ambiguidade dando prioridade às produções de acordo com a sequência em que ocorrem. Nesse sentido, cabe ao usuário se comprometer em criar a gramática com a precedência correta para que não haja resultados inesperados.