

Q1. (3 pts)

Tell me what is the so-called Dangling-else problem in the programming language, in your word. (1 pt) (I just want to know your understanding on this problem)

Give me an example when the Dangling-else problem happens? (1 pt)

Give me an example how to eliminate the ambiguity for the Dangling-else problem? (1 pt)

Q2. (2 pts)

If, I want to say a grammar is “ambiguous”, there are some conditions have to satisfy.

What are they? [Hint] 2 conditions

Q3. (4 pts)

Based on the definition in Q2 as well as the example in this grammar

$$E \rightarrow E + E \mid E * E \mid (E) \mid \text{id}$$

Show me,

- 1) It can yield the same string (2 pts) (For example, 2 result strings, the order of the replacement is different but eventually, they look totally the same!)**

- 2) The 2 strings can build up different parsing trees, respectively (2 pts)**

So, in this way, we can say the grammar we provided above is ambiguous

Q4. (2 pts)

Beirfly tell me what is the job for Lexical Analyzer and Syntax Analyzer (parser) and what's the difference of their roles?

Q5. (4 pts)

In Ch. 4.1.3 Syntax Error Handling, there are different levels of errors during the compilation. Tell me 1) what are those different levels of errors and 2) give me an example on each of them.