Quiz6

Due Feb 7 at 12pm **Points** 7 **Questions** 7
Available Feb 5 at 12am - Feb 7 at 12pm **Time Limit** 7 Minutes

Submission Details:

Time:	7 minutes
Current Score:	6 out of 7
Kept Score:	6 out of 7

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	7 minutes	6 out of 7

ⓘ Correct answers will be available on Feb 7 at 2pm.

Score for this quiz: 6 out of 7
Submitted Feb 7 at 12:34am
This attempt took 7 minutes.

Question 1 — 1 / 1 pts

A grammar is ambiguous if

- multiple ASTs can be made out of a parse tree
- a syntactic structure has multiple acceptable syntax trees
- a syntax tree refers to multiple derivations
- multiple parse trees correspond to the same AST

Question 2 — 1 / 1 pts

In the set of rules below, what is the order of precedence from high to low?

expr ::= *expr* + *term* | *term*

term ::= *term* * *factor* | *factor*

factor ::= (*expr*) | *number*

- *, ( ), +
- +, *, ( )
- ( ), +, *
- ( ), *, +

Question 3 — 1 / 1 pts

In the set of rules below, are + and * left-associative or right-associative?

expr ::= *expr* + *term* | *term*

term ::= *factor* * *term* | *factor*

factor ::= (*expr*) | *number*

- + left-associative, * left-associative
- + right-associative, * right-associative
- + left-associative, * right-associative
- + right-associative, * left-associative

Incorrect

Question 4 — 0 / 1 pts

What would be the EBNF version of the rule below?

term ::= *factor* * *term* | *factor*

- term ::= factor [* factor]
- term ::= factor {* term}
- term ::= factor [* term]
- term ::= factor {* factor}

Question 5 — 1 / 1 pts

Shift-reduce parser builds the syntax tree from leaves toward the root.

- True
- False

Question 6 — 1 / 1 pts

Recursive-descend parser builds the syntax tree from root toward the leaves

- True
- False

Question 7 — 1 / 1 pts

A recursive CFG rule results in a recursive function definition for parsing according to that rule.

- True
- False

Quiz Score: 6 out of 7