[MetaParserGrammer] [MetaParserRoot]

<MetaParserRoot> \*::= [StringInSquareBrackets] [StringInSquareBrackets]

{SemGrammerStart} <ProductionRule> <ProductionRuleListOpt> \*:

\*END-OF-RULE

<ProductionRuleListOpt> \*::= <ProductionRule> <ProductionRuleListOpt> \*:

END-OF-GRAMMER {SemGrammerEnd} \*:

\*END-OF-RULE

<ProductionRule> \*::= \*<StringInAngledBrackets> {SemRuleStart} \*::=

{SemProductionListStart} <ProductionList> \*:

\*END-OF-RULE

<ProductionList> \*::= <Production> <ProductionListOpt> \*:

\*END-OF-RULE

<ProductionListOpt> \*::= <Production> <ProductionListOpt> \*:

END-OF-RULE {SemProductionListEnd} {SemRuleEnd} \*:

\*END-OF-RULE

<Production> \*::= {SemProductionStart} <Element> <ElementListOpt> \*:

\*END-OF-RULE

<ElementListOpt> \*::= <Element> <ElementListOpt> \*:

: {SemProductionEnd} \*:

\*END-OF-RULE

<Element> \*::= \*MAKE-SPECIAL-SYMBOLS \*:

\*<SinAB> \*: # Generates an SinAB

\*\*<SSinAB> \*: # Generates an SSinAB

\*\*\*<SSSinAB> \*: # Generares an SSSinB

[SinSB] \*:

\*{SinCB} \*: # Generates an SinCB

\*\*{SSinCB} \*: # Generates an SSinCB

\*\*\*{SSSinCB} \*: # Generates ...

ABCD \*:

1234 \*:

END-OF-RULE \*:

END-OF-GRAMMER \*:

::= \*:

: \*:

\*END-OF-RULE

\*END-OF-GRAMMER