Blog Entry © Saturday, December 6, 2025, by James Pate Williams, Jr. Earley Parser

The Earley Parser is used in natural language processing. My C/C++ application that parses sentences in two simple grammars is from the textbook ***AI Algorithms, Data Structures, and Idioms in Prolog, Lisp, and Java*** © 2009 by George F. Luger and William A. Stubblefield. I translated the Java code first into C# source code in either 2011 or 2015 and on December 5, 2025, to the Win32 C/C++ Desktop. Below are my results:

```
Sentence: 2 + 3 * 4.
Parse Successful: true
Charts produced by the sentence: 2 + 3 * 4.
$ 0 0 @ S @ S
S 0 0 @ S + M @ S + M
S 0 0 @ M @ M
M 0 0 @ M * T @ M * T
M 0 0 @ T @ T
T 0 0 @ Number @ Number

Number 0 1 1 2 3 @ 1 2 3 @
T 0 1 Number @ Number @
M 0 1 T @ T @
S 0 1 M @ M @
M 0 1 M @ * T M @ * T
$ 0 1 S @ S @
S 0 1 S @ + M S @ + M

+ 1 2 @ @
S 0 2 S + @ M S + @ M
M 2 2 @ M * T @ M * T
M 2 2 @ T @ T
T 2 2 @ Number @ Number

Number 2 3 1 2 3 @ 1 2 3 @
T 2 3 Number @ Number @
M 2 3 T @ T @
S 0 3 S + M @ S + M @
M 2 3 M @ * T M @ * T
$ 0 3 S @ S @
S 0 3 S @ + M S @ + M

* 3 4 @ @
M 2 4 M * @ T M * @ T
T 4 4 @ Number @ Number

Number 4 5 1 2 3 @ 1 2 3 @
T 4 5 Number @ Number @
M 2 5 M * T @ M * T @
```

```
S 0 5 S + M @ S + M @
M 2 5 M @ * T M @ * T
$ 0 5 S @ S @
S 0 5 S @ + M S @ + M
```

```
Sentence: + 1 2 * 3.
Parse Successful: false
Charts produced by the sentence: + 1 2 * 3.
$ 0 0 @ S @ S
S 0 0 @ S + M @ S + M
S 0 0 @ M @ M
M 0 0 @ M * T @ M * T
M 0 0 @ T @ T
T 0 0 @ Number @ Number

Sentence: John called Mary.
Parse Successful: true
Charts produced by the sentence: John called Mary.
$ 0 0 @ S @ S
S 0 0 @ NP VP @ NP VP
NP 0 0 @ NP PP @ NP PP
NP 0 0 @ Noun @ Noun

Noun 0 1 @ @
NP 0 1 Noun @ Noun @
S 0 1 NP @ VP NP @ VP
NP 0 1 NP @ PP NP @ PP
VP 1 1 @ VP NP @ VP NP
VP 1 1 @ Verb PP @ Verb PP
VP 1 1 @ Verb Noun @ Verb Noun
VP 1 1 @ Verb @ Verb
PP 1 1 @ Prep NP @ Prep NP
PP 1 1 @ Prep @ Prep

Verb 1 2 @ @
VP 1 2 Verb @ PP Verb @ PP
VP 1 2 Verb @ Noun Verb @ Noun
VP 1 2 Verb @ Verb @
PP 2 2 @ Prep NP @ Prep NP
PP 2 2 @ Prep @ Prep
S 0 2 NP VP @ NP VP @
VP 1 2 VP @ NP VP @ NP
$ 0 2 S @ S @
NP 2 2 @ NP PP @ NP PP
NP 2 2 @ Noun @ Noun

Noun 2 3 @ @
VP 1 3 Verb Noun @ Verb Noun @
NP 2 3 Noun @ Noun @
S 0 3 NP VP @ NP VP @
VP 1 3 VP @ NP VP @ NP
VP 1 3 VP NP @ VP NP @
```

```
NP 2 3 NP @ PP NP @ PP
$ 0 3 S @ S @
NP 3 3 @ NP PP @ NP PP
NP 3 3 @ Noun @ Noun
PP 3 3 @ Prep NP @ Prep NP
PP 3 3 @ Prep @ Prep

Sentence: John called Mary from Denver.
Parse Successful: true
Charts produced by the sentence: John called Mary from Denver.
$ 0 0 @ S @ S
S 0 0 @ NP VP @ NP VP
NP 0 0 @ NP PP @ NP PP
NP 0 0 @ Noun @ Noun

Noun 0 1 @ @
NP 0 1 Noun @ Noun @
S 0 1 NP @ VP NP @ VP
NP 0 1 NP @ PP NP @ PP
VP 1 1 @ VP NP @ VP NP
VP 1 1 @ Verb PP @ Verb PP
VP 1 1 @ Verb Noun @ Verb Noun
VP 1 1 @ Verb @ Verb
PP 1 1 @ Prep NP @ Prep NP
PP 1 1 @ Prep @ Prep

Verb 1 2 @ @
VP 1 2 Verb @ PP Verb @ PP
VP 1 2 Verb @ Noun Verb @ Noun
VP 1 2 Verb @ Verb @
PP 2 2 @ Prep NP @ Prep NP
PP 2 2 @ Prep @ Prep
S 0 2 NP VP @ NP VP @
VP 1 2 VP @ NP VP @ NP
$ 0 2 S @ S @
NP 2 2 @ NP PP @ NP PP
NP 2 2 @ Noun @ Noun

Noun 2 3 @ @
VP 1 3 Verb Noun @ Verb Noun @
NP 2 3 Noun @ Noun @
S 0 3 NP VP @ NP VP @
VP 1 3 VP @ NP VP @ NP
VP 1 3 VP NP @ VP NP @
NP 2 3 NP @ PP NP @ PP
$ 0 3 S @ S @
NP 3 3 @ NP PP @ NP PP
```

```
NP 3 3 @ Noun @ Noun
PP 3 3 @ Prep NP @ Prep NP
PP 3 3 @ Prep @ Prep

Prep 3 4 @ @
PP 3 4 Prep @ NP Prep @ NP
PP 3 4 Prep @ Prep @
NP 4 4 @ NP PP @ NP PP
NP 4 4 @ Noun @ Noun
NP 2 4 NP PP @ NP PP @
VP 1 4 VP NP @ VP NP @
NP 2 4 NP @ PP NP @ PP
S 0 4 NP VP @ NP VP @
VP 1 4 VP @ NP VP @ NP
PP 4 4 @ Prep NP @ Prep NP
PP 4 4 @ Prep @ Prep
$ 0 4 S @ S @

Noun 4 5 @ @
NP 4 5 Noun @ Noun @
PP 3 5 Prep NP @ Prep NP @
NP 4 5 NP @ PP NP @ PP
VP 1 5 VP NP @ VP NP @
NP 2 5 NP PP @ NP PP @
PP 5 5 @ Prep NP @ Prep NP
PP 5 5 @ Prep @ Prep
S 0 5 NP VP @ NP VP @
VP 1 5 VP @ NP VP @ NP
NP 2 5 NP @ PP NP @ PP
$ 0 5 S @ S @
NP 5 5 @ NP PP @ NP PP
NP 5 5 @ Noun @ Noun
```