Parsing: Chart Parsing, Earley-Algorithm, top down/bottom up, left-right

Permanent data structure:

grammar (context free phrase structure grammar)

Input: a string of words $w_1 w_2 \dots w_3$

Working structures:

<u>Input string</u> - input with position markers 0 w_1 1 w_2 2 ... w_n n <u>Position</u> – point in the input string

<u>Chart</u> - set of items (edges) Every item is a triple $\langle h, i, A \rightarrow \alpha. \beta \rangle$, where h and i are positions of the input string, and $A \rightarrow \alpha. \beta$ a dotted rule. α and β are members of $(V_T \cup V_N)^*$.

 $A \rightarrow \alpha$ β is a rule of the grammar. α is the part of the right-handside of the rule that has already been applied to the segment of the input string from h to i.

Initially the chart contains for each rule of the form $S \to \alpha$ exactly one item $\langle 0, 0, S \to .\alpha \rangle$. (initialization)

Algorithm:

For each position *i* from 0 to *n* repeat the following steps until no more items can be produced:

Scanner (consumption of input)

If $w_i = a$ add for each item $\langle h, i-1, A \rightarrow \alpha.a \beta \rangle$ a new item $\langle h, i, A \rightarrow \alpha.a.\beta \rangle$.

Completer (completion of constituents)

For each pair of items of the form $\langle h, i, A \rightarrow \alpha. \rangle$ and $\langle k, h, B \rightarrow \beta. A \gamma \rangle$ add a new entry $\langle k, i, B \rightarrow \beta A. \gamma \rangle$ to the chart if it is not already present in the chart.

<u>Predictor</u> (top down prediction of constituents)

For each item $\langle h, i, A \rightarrow \alpha.B \beta \rangle$ add for each rule $B \rightarrow \gamma$ a new item $\langle i, i, B \rightarrow .\gamma \rangle$ to the chart if it is not already present in the chart.

If the chart contains at least one item $\langle 0, n, S \rightarrow \alpha. \rangle$ return **success**, else return **failure**.