Parsing Problem - Given a Grammar \( G \) and a string \( \omega \in T^* \), is \( \omega \in L(G) \)?

Younger's Algorithm (This is not a standard parsing algorithm. It is very easy to understand but is not efficient enough to be used in practice.)

Grammar:

\[
\begin{align*}
S & \rightarrow bA \\
A & \rightarrow a \\
B & \rightarrow b \\
A & \rightarrow aS \\
B & \rightarrow bS \\
A & \rightarrow bAA \\
B & \rightarrow aBB
\end{align*}
\]

String: \( \omega = aaabbb aa bbb \)

(1) First step: put grammar in Chomsky Normal Form.

\[
\begin{align*}
S & \rightarrow C_1 A \\
C_3 & \rightarrow AA \\
C_1 & \rightarrow b \\
C_2 & \rightarrow a \\
A & \rightarrow a \\
S & \rightarrow C_2 B \\
C_4 & \rightarrow BB \\
A & \rightarrow C_2 S \\
B & \rightarrow b \\
C_1 & \rightarrow C_1 \ C_3 \\
B & \rightarrow C_1 S
\end{align*}
\]
2. Construct recognition matrix $REC$. Let $a_1 a_2 \ldots a_n$ be the sentence we are attempting to recognize. $REC(i,j)$ will contain all nonterminals from which $a_j a_{j+1} \ldots a_{j+i-1}$ can be derived.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>a</td>
<td>a</td>
<td>a</td>
<td>b</td>
<td>b</td>
<td>a</td>
<td>a</td>
<td>b</td>
<td>b</td>
<td>b</td>
</tr>
</tbody>
</table>

Accept string if $S$ is contained in $REC(n, 1)$ after matrix is computed.
amt of work needed:

\[ n + \sum_{i=2}^{n} (n-i+1)(i-1) \]

\[ \sim n^3 \]

By noticing the similarity of this algorithm to matrix multiplication and using Strassen's trick, a clever Post doctoral student at CMU was able to modify the algorithm to run in time \( n^{2.78} \).

No one knows if there is a better algorithm or not.