הסרת רקורסיה שמאלית
רקורסיה שמאלית ישירה

\[ A \rightarrow A\alpha | \beta \]

העתקה של תבניות פסוקיות, קיימים \( \beta \) ו- \( \alpha \) כאשר \( \beta \) לא מתוחילה

כשלושา עצמן עם \( A \).
רקורסיה שמאלית עקיפה

דוגמה פשוטה:

\[
A \rightarrow B\alpha | C \\
B \rightarrow A\beta | D
\]

\[A \Rightarrow B\alpha \Rightarrow A\beta\alpha \Rightarrow \ldots\]

–-entity-לפי-או-הוא-ה授信-hunger:

באובן כללי:

\[
A_0 \rightarrow A_1\alpha_1 | \ldots \\
A_1 \rightarrow A_2\alpha_2 | \ldots \\
\ldots \\
A_n \rightarrow A_0\alpha_{n+1} | \ldots
\]
 לעבור לקורסית יישור

נחקק את הכללים:

\[
A \rightarrow \alpha_1 | \alpha_2 | \cdots | \alpha_n | \beta_1 | \beta_2 | \cdots | \beta_n
\]

בכללים:

\[
A \rightarrow \beta_1 A' | \beta_2 A' | \cdots | \beta_n A' \\
A' \rightarrow \alpha_1 A' | \alpha_2 A' | \cdots | \alpha_n A' | \varepsilon
\]

שים לב שהשיטה לא עובדת אם \( \alpha_i \) ריק.

(נוצרת Kıורסית שמאלית של \( A' \) (A' Kıורסית שמאלית של \( A \)

בנוסף, היהvlaolina ליוצר Kıורסית שמאלית עקיפה אם \( \beta_i \) ריק,

כ示范区 \( \alpha_j \) מתחליל ב:

\[
A' \rightarrow A \ldots \text{としても} \ A \rightarrow A'
\]
A → Aa | b

A → bA'
A' → aA' | ε where A' is a non-terminal
 descargar los algoritmos de eliminación de recursividad hacia la izquierda (aparatos manuales)

• קלט: דקדוק שלול ייש רקורסיה שמאלית, ללא מעגלים, לכל אפסילון.
• פלט: דקדוק שלול ייש רקורסיה שמאלית.

. דוגמה לכלל אפסילון: $A \rightarrow \varepsilon$
. דוגמה למעגל: $A \rightarrow B; B \rightarrow A$.

ניתן לטיל כללי אפסילון ומעגלים בדקדוק (באוגר אוטומטי).

רעיון האלגוריתם לסילוק רקורסיה שמאלית: נסדר את המשнтנים לפי סדר $A_1, A_2, \ldots, A_n$ לכלשון: נעבור על המשнтנים לפי הסדר, לכל אפסילון נדאג לכל שול של יהי ממוזרה $A_i \rightarrow A_j \beta$ with $j > i$.

מדוע זו מספיק?
Input: Grammar $G$ possibly left-recursive, no cycles, no $\epsilon$ productions.
Output: An equivalent grammar with no left-recursion
Method: Sort the non-terminals $A_1, A_2, \ldots, A_n$

```
for i in [1..n]:
    for s in [1..i-1]:
        replace each rule $A_i \rightarrow A_s \beta$
        with $A_i \rightarrow d_1 \beta | d_2 \beta | \ldots | d_k \beta$
        where $A_s \rightarrow d_1 | d_2 | \ldots | d_k$

eliminate immediate left recursion among the $A_i$-productions
```
Example:

\[
\begin{align*}
S & \rightarrow C \mid a \\
C & \rightarrow Dd \mid c \\
D & \rightarrowCc \mid d \\
D' & \rightarrow DdC \mid cc \mid d
\end{align*}
\]

Let \( A_1 = S, A_2 = C, A_3 = D \).

Remove indirect recursion in \( A_3(D) \)

\[
\begin{align*}
S & \rightarrow C \mid a \\
C & \rightarrow Dd \mid c \\
D & \rightarrow DdC \mid cc \mid d \\
D' & \rightarrow ccD' \mid dD' \mid \epsilon
\end{align*}
\]

The productions are now:

\[
\begin{align*}
S & \rightarrow C \mid a \\
C & \rightarrow Dd \mid c \\
D & \rightarrow ccD' \mid dD' \\
D' & \rightarrow dcD' \mid \epsilon
\end{align*}
\]

No left recursion!
ניתוח האלגוריתם

נראה שבסיום האלגוריתם כל חוק גזירה מהצורה \( A_k \rightarrow A_t\beta \) מקיים \( t > k \).

שמורה 1: י阐释 ofere את החללים הפינימיים עבורה \( s \) כלשהיה (עם בלולאת \( A_i \) החלק האורכי \( A_j \) הפינימי של \( A_i \) מתחלים בטרמינלים או במשתנים \( A_j \) עבור \( j > s \).

שמורה 2: \( \forall i \) כלכל הגזירה של \( A_i \) מתחלים בטרמינלים או במשתנים \( A_j \) עבור \( j > i \) או בטרמינלים.

הוכחת השמורות יחד באינדוקציה על \( i \) ו-\( s \).

מסקנה: בנבי הגזירה עסקו ראוניו בפקודת \( A_i \) של \( A_i \) מתחלים במשתנים המקוריים (רישעה או עקרפה).

نوוט משמורה 2.

לגביו המשתנים החדשים, הם תמיים יוכדיעים כימגיים בצדית, لكل苎ר לא יתייר מעורבם בפקודת \( A_i \) מתחלים במשתנים.