Théorie des Langages
Épisode 5 — Analyse descendante
Grammaires LL(1)

Thomas Pietrzak

Université Paul Verlaine — Metz

Analyse descendante : grammaires LL(1)

L'analyse descendante (top-down parsing) peut être considérée comme une tentative pour déterminer une dérivation gauche associée à un mot d'entrée ou un programme à compiler.

Elle peut aussi être vue comme une tentative pour construire un arbre abstrait syntaxique d'un mot d'entrée ou un programme à compiler, en partant de la racine et en créant les noeuds de l'arbre en préordre.

Note

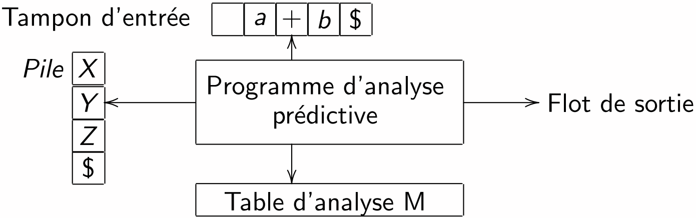
Nous considérons les grammaires hors contexte sans symboles inutiles.

Analyseur prédictif

Dans la plupart des cas, en écrivant soigneusement une grammaire, après avoir éliminé les récursivités gauche et l'avoir factorisée à gauche, nous pouvons obtenir une grammaire qui peut être analysée par descente récursive ou non récursive sans rebroussement, c'est à dire obtenir un analyseur prédictif.

Nous considérons un analyseur prédictif non récursif. Le problème clé de l'analyse prédictive est la détermination de la production à appliquer pour développer un non-terminal.

L'analyseur non récursif de la figure ci-dessous recherche la production à appliquer dans une table d'analyse.
Dans ce qui suit, nous verrons comment cette table peut, pour certaines grammaires, être construite directement.



Un analyseur syntaxique prédictif dirigé par table possède :

- Un tampon d'entrée : contient la chaîne à analyser, suivie de \$, symbole utilisé comme marqueur d'extremité droite qui indique la fin de la chaîne d'entrée.
- Une pile : contient une séquence de symboles grammaticaux, avec \$ marquant le fond de pile. Initialement, la pile contient l'axiome de la grammaire au dessus du \$
- Une table d'analyse : tableau à deux dimensions $M[A, a]$ où A est un non-terminal et a un terminal ou le symbole \$.
- Un flot de sortie : ce qu'on récupère.

Principe

L'analyse syntaxique est contrôlée par un programme qui a le comportement suivant, sachant que X est le symbole au sommet de la pile, et a le symbole d'entrée courant. Ces deux symboles déterminent l'action de l'analyseur. Il y a trois possibilités :

1. Si $X = a = \$$: l'analyseur s'arrête et annonce la réussite finale de l'analyse.
2. Si $X = a \neq \$$: l'analyseur enlève X de la pile et avance son pointeur d'entrée sur le symbole suivant.
3. Si X est un non-terminal, le programme consulte l'entrée $M[X, a]$ de la table d'analyse M. Cette entrée sera soit une X-production de la grammaire soit une erreur.
   - Si par exemple, $M[X, a] = \{X \rightarrow UVW\}$, l'analyseur remplace X en sommet de pile par WVU (avec U au sommet). Nous supposerons que l'analyseur se contente pour tout résultat d'imprimer la production utilisée. N'importe quelle autre action pourrait être exécutée à la place.
   - Si $M[X, a] = erreur$, l'analyseur appelle une procédure de récupération sur l'erreur.

Configuration

Le comportement de l'analyseur peut se décrire en termes de ses configurations, qui décrivent le contenu de la pile et le texte d'entrée restant.

Algorithme d'analyse prédictive non récursive

- **Donnée** : une chaîne w et une table d'analyse M pour une grammaire G.
- **Résultat** : si w est dans $L(G)$, une dérivation gauche pour w, sinon une indication d'erreur.
- **Méthode** : initialement, l'analyseur est dans une configuration dans laquelle il y a $\$S$ dans la pile, avec S l'axiome de G au sommet, et $w\$$ dans son tampon d'entrée. Le programme qui utilise la table d'analyse prédictive M pour effectuer une analyse syntaxique de la chaîne d'entrée est présenté sur la page suivante.

Algorithme

- Positionner le pointeur source *ps* sur le premier symbole de $w\$$
- Répéter jusqu'à ce que $X = \$$ (tant que la pile n'est pas vide quoi)
  - Soit X le symbole en sommet de pile et a le symbole pointé par *ps*
  - Si X est un terminal ou \$ alors
    - Si $X = a$ alors enlever X de la pile et avancer *ps*
    - Sinon $Erreur()$
  - Sinon (X est un non-terminal)
    - Si $M[X, a] = X \rightarrow Y_1 Y_2 \ldots Y_n$ alors
    - Début :
    - Enlever X de la pile
    - Mettre $Y_n, Y_{n-1}, \ldots, Y_1$ sur la pile (avec Y_1 au sommet
    - Émettre en sortie la production $X \rightarrow Y_1 Y_2 \ldots Y_n$ (ou faire l'action appropriée)
    - Fin
    - Sinon $Erreur()$

Exemple

Considérons la grammaire :

$E \rightarrow TE'$
$E' \rightarrow +TE' \mid \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT' \mid \epsilon$
$F \rightarrow (E) \mid id$

La table d'analyse prédictive pour cette grammaire est représentée ci-dessous. Dans cette table, les entrées vides représentent les erreurs, les entrées non vides indiquent les productions qu'il faut utiliser pour développer le non terminal en sommet de pile.

Non terminal	Symbole d'entrée					
	id	$+$	$*$	$($	$)$	$\$$
E	$E \rightarrow TE'$			$E \rightarrow TE'$		
E'		$E' \rightarrow +TE'$			$E' \rightarrow \epsilon$	$E' \rightarrow \epsilon$
T	$T \rightarrow FT'$			$T \rightarrow FT'$		
T'		$T' \rightarrow \epsilon$	$T' \rightarrow *FT'$		$T' \rightarrow \epsilon$	$T' \rightarrow \epsilon$
F	$F \rightarrow id$			$F \rightarrow (E)$		

Sur la chaîne source $id + id * id$ l'analyseur prédictif effectue la qéquence d'actions de la table ci-dessous.

Pile	Entrée	Sortie
$\$E$	$id + id * id\$$	
$\$E'T$	$id + id * id\$$	$E \rightarrow TE'$
$\$E'T'F$	$id + id * id\$$	$T \rightarrow FT'$
$\$E'T'id$	$id + id * id\$$	$F \rightarrow id$
$\$E'T'$	$+id * id\$$	
$\$E'$	$+id * id\$$	$T' \rightarrow \epsilon$
$\$E'T+$	$+id * id\$$	$E' \rightarrow +TE'$
$\$E'T$	$id * id\$$	
$\$E'T'F$	$id * id\$$	$T' \rightarrow FT'$
$\$E'T'id$	$id * id\$$	$F \rightarrow id$
$\$E'T'$	$*id\$$	
$\$E'T'F*$	$*id\$$	$T' \rightarrow *FT'$
$\$E'T'F$	$id\$$	
$\$E'T'id$	$id\$$	$F \rightarrow id$
$\$E'T'$	$\$$	
$\$E'$	$\$$	$T' \rightarrow \epsilon$
$\$$	$\$$	$E' \rightarrow \epsilon$

Si nous étudions soigneusement les actions de cet analyseur nous voyons qu'elles décrivent une dérivation gauche de la chaîne source.

Premier et *Suivant* (*First* et *Follow*)

- La configuration d'un analyseur syntaxique prédictif est facilitée par deux fonctions associées à une grammaire G.
- Ces fonctions *Premier* et *Suivant* nous permettent, si possible, de remplir les entrées de la table d'analyse prédictive pour G.
- Si α est une chaîne de symboles grammaticaux, *Premier*(α) désigne l'ensemble des terminaux qui commencent les chaînes qui se dérivent de α. Si $\alpha \Rightarrow^* \epsilon$ alors ϵ est aussi dans *Premier*(α).
- Pour chaque non-terminal A, *Suivant*(A) définit l'ensemble des terminaux a qui peuvent apparaître immédiatement à droite de A dans une proto-phrase, c'est à dire l'ensemble des terminaux a tels qu'il existe une dérivation de la forme $S \Rightarrow^* \alpha A a \beta$ où α et β sont des chaînes de symboles grammaticaux.
- Si A peut être le symbole le plus à droite d'une proto-phrase, alors \$ est dans *Suivant*(A).

Calcul de *Premier* pour un symbole

Pour calculer $Premier(X)$, pour tout symbole de la grammaire X, appliquer les règles suivantes jusqu'à ce qu'aucun terminal ni ϵ ne puisse être ajouté aux *Premier*.

1. Si X est un terminal, $Premier(X)$ est $\{X\}$.
2. Si $X \rightarrow \epsilon$ une production, ajouter ϵ à $Premier(X)$.
3. Si X est non-terminal et $X \rightarrow Y_1 Y_2 \ldots Y_n$ une production, mettre a dans $Premier(X)$ s'il existe i tel que a est dans $Premier(Yi)$ et que ϵ est dans tous les $Premier(Y_1), \ldots, Premier(Y_{i-1})$, c'est à dire $Y_1 \ldots Y_{i-1} \Rightarrow^* \epsilon$. Si ϵ est dans $Premier(Y_j)$ pour tous les $J = 1, 2, \ldots, n$, ajouter ϵ à $Premier(X)$.

Calcul de *Premier* pour une chaîne

Maintenant nous pouvons calculer *Premier* pour une chaîne $X_1X_2\ldots X_n$ de la façon suivante :

1. Ajouter à $Premier(X_1X_2\ldots X_n)$ tous les symboles de $Premier(X_1)$ différents de ϵ.
2. Si ϵ est dans $Premier(X_1)$, ajouter également les symboles de $Premier(X_2)$ différents de ϵ, etc.
3. Finalement, si quel que soit i, $Premier(X_i)$ contient ϵ, ajouter ϵ à $Premier(X_1X_2\ldots X_n)$.

Calcul de *Suivant* pour les non-terminaux

Pour calculer $Suivant(A)$ pour tous les non-terminaux A, appliquer les règles suivantes jusqu'à ce qu'aucun terminal ne puisse être ajouté aux ensembles *Suivant*.

1. Mettre \$ dans $Suivant(S)$ où S est l'axiome et \$ est le marqueur indiquant la fin du texte source.
2. S'il y a une production $A \longrightarrow \alpha B \beta$, le contenu de $Premier(\beta)$, excepté ϵ est ajouté à $Suivant(B)$.
3. S'il existe une production $A \longrightarrow \alpha B$ ou une production $A \longrightarrow \alpha B \beta$ telle que $Premier(\beta)$ contient ϵ (c'est à dire $\beta \Rightarrow^* \epsilon$), les éléments de $Suivant(A)$ sont ajoutés à $Suivant(B)$.

Reprenons la grammaire :
$E \rightarrow TE'$
$E' \rightarrow +TE' \mid \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT' \mid \epsilon$
$F \rightarrow (E) \mid id$
$Premier(E) = \{\}$
$Premier(E') = \{\}$
$Premier(T) = \{\}$
$Premier(T') = \{\}$
$Premier(F) = \{\}$

- $E' \rightarrow \epsilon$ donc on ajoute ϵ à $Premier(E')$
- $T' \rightarrow \epsilon$ donc on ajoute ϵ à $Premier(T')$
- $E' \rightarrow +TE'$ donc on ajoute $+$ à $Premier(E')$
- $T' \rightarrow *FT'$ donc on ajoute $*$ à $Premier(T')$
- $F \rightarrow (E)$ donc on ajoute (à $Premier(F)$
- $F \rightarrow id$ donc on ajoute id à $Premier(F)$
- $T \rightarrow FT'$ donc on ajoute $Premier(F)$ à $Premier(T)$
- $E \rightarrow TE'$ donc on ajoute $Premier(T)$ à $Premier(E)$

Reprenons la grammaire :
$E \rightarrow TE'$
$E' \rightarrow +TE' \mid \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT' \mid \epsilon$
$F \rightarrow (E) \mid id$
$Premier(E) = \{\}$
$Premier(E') = \{\epsilon\}$
$Premier(T) = \{\}$
$Premier(T') = \{\}$
$Premier(F) = \{\}$

- $E' \rightarrow \epsilon$ donc on ajoute ϵ à $Premier(E')$
- $T' \rightarrow \epsilon$ donc on ajoute ϵ à $Premier(T')$
- $E' \rightarrow +TE'$ donc on ajoute $+$ à $Premier(E')$
- $T' \rightarrow *FT'$ donc on ajoute $*$ à $Premier(T')$
- $F \rightarrow (E)$ donc on ajoute (à $Premier(F)$
- $F \rightarrow id$ donc on ajoute id à $Premier(F)$
- $T \rightarrow FT'$ donc on ajoute $Premier(F)$ à $Premier(T)$
- $E \rightarrow TE'$ donc on ajoute $Premier(T)$ à $Premier(E)$

Reprenons la grammaire :

$E \rightarrow TE'$

$E' \rightarrow +TE' \mid \epsilon$

$T \rightarrow FT'$

$T' \rightarrow *FT' \mid \epsilon$

$F \rightarrow (E) \mid id$

$Premier(E) = \{\}$

$Premier(E') = \{\epsilon\}$

$Premier(T) = \{\}$

$Premier(T') = \{\epsilon\}$

$Premier(F) = \{\}$

- $E' \rightarrow \epsilon$ donc on ajoute ϵ à $Premier(E')$
- $T' \rightarrow \epsilon$ donc on ajoute ϵ à $Premier(T')$
- $E' \rightarrow +TE'$ donc on ajoute $+$ à $Premier(E')$
- $T' \rightarrow *FT'$ donc on ajoute $*$ à $Premier(T')$
- $F \rightarrow (E)$ donc on ajoute (à $Premier(F)$
- $F \rightarrow id$ donc on ajoute id à $Premier(F)$
- $T \rightarrow FT'$ donc on ajoute $Premier(F)$ à $Premier(T)$
- $E \rightarrow TE'$ donc on ajoute $Premier(T)$ à $Premier(E)$

Reprenons la grammaire :
$E \rightarrow TE'$
$E' \rightarrow +TE' \mid \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT' \mid \epsilon$
$F \rightarrow (E) \mid id$
$Premier(E) = \{\}$
$Premier(E') = \{\epsilon, +\}$
$Premier(T) = \{\}$
$Premier(T') = \{\epsilon\}$
$Premier(F) = \{\}$

- $E' \rightarrow \epsilon$ donc on ajoute ϵ à $Premier(E')$
- $T' \rightarrow \epsilon$ donc on ajoute ϵ à $Premier(T')$
- $E' \rightarrow +TE'$ donc on ajoute $+$ à $Premier(E')$
- $T' \rightarrow *FT'$ donc on ajoute $*$ à $Premier(T')$
- $F \rightarrow (E)$ donc on ajoute (à $Premier(F)$
- $F \rightarrow id$ donc on ajoute id à $Premier(F)$
- $T \rightarrow FT'$ donc on ajoute $Premier(F)$ à $Premier(T)$
- $E \rightarrow TE'$ donc on ajoute $Premier(T)$ à $Premier(E)$

Reprenons la grammaire :
$E \rightarrow TE'$
$E' \rightarrow +TE' \mid \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT' \mid \epsilon$
$F \rightarrow (E) \mid id$
$Premier(E) = \{\}$
$Premier(E') = \{\epsilon, +\}$
$Premier(T) = \{\}$
$Premier(T') = \{\epsilon, *\}$
$Premier(F) = \{\}$

- $E' \rightarrow \epsilon$ donc on ajoute ϵ à $Premier(E')$
- $T' \rightarrow \epsilon$ donc on ajoute ϵ à $Premier(T')$
- $E' \rightarrow +TE'$ donc on ajoute $+$ à $Premier(E')$
- $T' \rightarrow *FT'$ donc on ajoute $*$ à $Premier(T')$
- $F \rightarrow (E)$ donc on ajoute (à $Premier(F)$
- $F \rightarrow id$ donc on ajoute id à $Premier(F)$
- $T \rightarrow FT'$ donc on ajoute $Premier(F)$ à $Premier(T)$
- $E \rightarrow TE'$ donc on ajoute $Premier(T)$ à $Premier(E)$

Reprenons la grammaire :
$E \rightarrow TE'$
$E' \rightarrow +TE' \mid \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT' \mid \epsilon$
$F \rightarrow (E) \mid id$
$Premier(E) = \{\}$
$Premier(E') = \{\epsilon, +\}$
$Premier(T) = \{\}$
$Premier(T') = \{\epsilon, *\}$
$Premier(F) = \{(\}$

- $E' \rightarrow \epsilon$ donc on ajoute ϵ à $Premier(E')$
- $T' \rightarrow \epsilon$ donc on ajoute ϵ à $Premier(T')$
- $E' \rightarrow +TE'$ donc on ajoute $+$ à $Premier(E')$
- $T' \rightarrow *FT'$ donc on ajoute $*$ à $Premier(T')$
- $F \rightarrow (E)$ donc on ajoute (à $Premier(F)$
- $F \rightarrow id$ donc on ajoute id à $Premier(F)$
- $T \rightarrow FT'$ donc on ajoute $Premier(F)$ à $Premier(T)$
- $E \rightarrow TE'$ donc on ajoute $Premier(T)$ à $Premier(E)$

Reprenons la grammaire :
$E \rightarrow TE'$
$E' \rightarrow +TE' \mid \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT' \mid \epsilon$
$F \rightarrow (E) \mid id$
$Premier(E) = \{\}$
$Premier(E') = \{\epsilon, +\}$
$Premier(T) = \{\}$
$Premier(T') = \{\epsilon, *\}$
$Premier(F) = \{(, id\}$

- $E' \rightarrow \epsilon$ donc on ajoute ϵ à $Premier(E')$
- $T' \rightarrow \epsilon$ donc on ajoute ϵ à $Premier(T')$
- $E' \rightarrow +TE'$ donc on ajoute $+$ à $Premier(E')$
- $T' \rightarrow *FT'$ donc on ajoute $*$ à $Premier(T')$
- $F \rightarrow (E)$ donc on ajoute (à $Premier(F)$
- $F \rightarrow id$ donc on ajoute id à $Premier(F)$
- $T \rightarrow FT'$ donc on ajoute $Premier(F)$ à $Premier(T)$
- $E \rightarrow TE'$ donc on ajoute $Premier(T)$ à $Premier(E)$

Reprenons la grammaire :
$E \rightarrow TE'$
$E' \rightarrow +TE' \mid \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT' \mid \epsilon$
$F \rightarrow (E) \mid id$
$Premier(E) = \{\}$
$Premier(E') = \{\epsilon, +\}$
$Premier(T) = \{(, id\}$
$Premier(T') = \{\epsilon, *\}$
$Premier(F) = \{(, id\}$

- $E' \rightarrow \epsilon$ donc on ajoute ϵ à $Premier(E')$
- $T' \rightarrow \epsilon$ donc on ajoute ϵ à $Premier(T')$
- $E' \rightarrow +TE'$ donc on ajoute $+$ à $Premier(E')$
- $T' \rightarrow *FT'$ donc on ajoute $*$ à $Premier(T')$
- $F \rightarrow (E)$ donc on ajoute (à $Premier(F)$
- $F \rightarrow id$ donc on ajoute id à $Premier(F)$
- $T \rightarrow FT'$ donc on ajoute $Premier(F)$ à $Premier(T)$
- $E \rightarrow TE'$ donc on ajoute $Premier(T)$ à $Premier(E)$

Reprenons la grammaire :

$E \rightarrow TE'$

$E' \rightarrow +TE' \mid \epsilon$

$T \rightarrow FT'$

$T' \rightarrow *FT' \mid \epsilon$

$F \rightarrow (E) \mid id$

$Premier(E) = \{(, id\}$

$Premier(E') = \{\epsilon, +\}$

$Premier(T) = \{(, id\}$

$Premier(T') = \{\epsilon, *\}$

$Premier(F) = \{(, id\}$

- $E' \rightarrow \epsilon$ donc on ajoute ϵ à $Premier(E')$
- $T' \rightarrow \epsilon$ donc on ajoute ϵ à $Premier(T')$
- $E' \rightarrow +TE'$ donc on ajoute $+$ à $Premier(E')$
- $T' \rightarrow *FT'$ donc on ajoute $*$ à $Premier(T')$
- $F \rightarrow (E)$ donc on ajoute (à $Premier(F)$
- $F \rightarrow id$ donc on ajoute id à $Premier(F)$
- $T \rightarrow FT'$ donc on ajoute $Premier(F)$ à $Premier(T)$
- $E \rightarrow TE'$ donc on ajoute $Premier(T)$ à $Premier(E)$

Reprenons la grammaire :
$E \rightarrow TE'$
$E' \rightarrow +TE' \mid \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT' \mid \epsilon$
$F \rightarrow (E) \mid id$
$Premier(E) = \{(, id\}$
$Premier(E') = \{\epsilon, +\}$
$Premier(T) = \{(, id\}$
$Premier(T') = \{\epsilon, *\}$
$Premier(F) = \{(, id\}$
$Suivant(E) = \{\}$
$Suivant(E') = \{\}$
$Suivant(T) = \{\}$
$Suivant(T') = \{\}$
$Suivant(F) = \{\}$

- E est l'axiome donc on ajoute \$ à $Suivant(E)$
- $E \rightarrow TE'$ donc on ajoute $Premier(E') \setminus \{\epsilon\}$ à $Suivant(T)$
- $T \rightarrow FT'$ donc on ajoute $Premier(T') \setminus \{\epsilon\}$ à $Suivant(F)$
- $F \rightarrow (E)$ donc on ajoute) à $Suivant(E)$
- $E \rightarrow TE'$ donc on ajoute $Suivant(E)$ à $Suivant(E')$
- $E' \rightarrow +TE'$ et $E' \rightarrow \epsilon$ donc on ajoute $Suivant(E')$ à $Suivant(T)$
- $T \rightarrow FT'$ donc on ajoute $Suivant(T)$ à $Suivant(T')$
- $T' \rightarrow *FT'$ et $T' \rightarrow \epsilon$ donc on ajoute $Suivant(T')$ à $Suivant(F)$

Reprenons la grammaire :

$E \rightarrow TE'$
$E' \rightarrow +TE' \mid \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT' \mid \epsilon$
$F \rightarrow (E) \mid id$
$Premier(E) = \{(, id\}$
$Premier(E') = \{\epsilon, +\}$
$Premier(T) = \{(, id\}$
$Premier(T') = \{\epsilon, *\}$
$Premier(F) = \{(, id\}$
$Suivant(E) = \{\$\}$
$Suivant(E') = \{\}$
$Suivant(T) = \{\}$
$Suivant(T') = \{\}$
$Suivant(F) = \{\}$

- E est l'axiome donc on ajoute \$ à $Suivant(E)$
- $E \rightarrow TE'$ donc on ajoute $Premier(E') \setminus \{\epsilon\}$ à $Suivant(T)$
- $T \rightarrow FT'$ donc on ajoute $Premier(T') \setminus \{\epsilon\}$ à $Suivant(F)$
- $F \rightarrow (E)$ donc on ajoute) à $Suivant(E)$
- $E \rightarrow TE'$ donc on ajoute $Suivant(E)$ à $Suivant(E')$
- $E' \rightarrow +TE'$ et $E' \rightarrow \epsilon$ donc on ajoute $Suivant(E')$ à $Suivant(T)$
- $T \rightarrow FT'$ donc on ajoute $Suivant(T)$ à $Suivant(T')$
- $T' \rightarrow *FT'$ et $T' \rightarrow \epsilon$ donc on ajoute $Suivant(T')$ à $Suivant(F)$

Reprenons la grammaire :
$E \rightarrow TE'$
$E' \rightarrow +TE' \mid \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT' \mid \epsilon$
$F \rightarrow (E) \mid id$
$Premier(E) = \{(, id\}$
$Premier(E') = \{\epsilon, +\}$
$Premier(T) = \{(, id\}$
$Premier(T') = \{\epsilon, *\}$
$Premier(F) = \{(, id\}$
$Suivant(E) = \{\$\}$
$Suivant(E') = \{\}$
$Suivant(T) = \{+\}$
$Suivant(T') = \{\}$
$Suivant(F) = \{\}$

- E est l'axiome donc on ajoute \$ à $Suivant(E)$
- $E \rightarrow TE'$ donc on ajoute $Premier(E') \setminus \{\epsilon\}$ à $Suivant(T)$
- $T \rightarrow FT'$ donc on ajoute $Premier(T') \setminus \{\epsilon\}$ à $Suivant(F)$
- $F \rightarrow (E)$ donc on ajoute) à $Suivant(E)$
- $E \rightarrow TE'$ donc on ajoute $Suivant(E)$ à $Suivant(E')$
- $E' \rightarrow +TE'$ et $E' \rightarrow \epsilon$ donc on ajoute $Suivant(E')$ à $Suivant(T)$
- $T \rightarrow FT'$ donc on ajoute $Suivant(T)$ à $Suivant(T')$
- $T' \rightarrow *FT'$ et $T' \rightarrow \epsilon$ donc on ajoute $Suivant(T')$ à $Suivant(F)$

Reprenons la grammaire :
$E \to TE'$
$E' \to +TE' \mid \epsilon$
$T \to FT'$
$T' \to *FT' \mid \epsilon$
$F \to (E) \mid id$
$Premier(E) = \{(, id\}$
$Premier(E') = \{\epsilon, +\}$
$Premier(T) = \{(, id\}$
$Premier(T') = \{\epsilon, *\}$
$Premier(F) = \{(, id\}$
$Suivant(E) = \{\$\}$
$Suivant(E') = \{\}$
$Suivant(T) = \{+\}$
$Suivant(T') = \{\}$
$Suivant(F) = \{*\}$

- E est l'axiome donc on ajoute \$ à $Suivant(E)$
- $E \to TE'$ donc on ajoute $Premier(E') \setminus \{\epsilon\}$ à $Suivant(T)$
- $T \to FT'$ donc on ajoute $Premier(T') \setminus \{\epsilon\}$ à $Suivant(F)$
- $F \to (E)$ donc on ajoute) à $Suivant(E)$
- $E \to TE'$ donc on ajoute $Suivant(E)$ à $Suivant(E')$
- $E' \to +TE'$ et $E' \to \epsilon$ donc on ajoute $Suivant(E')$ à $Suivant(T)$
- $T \to FT'$ donc on ajoute $Suivant(T)$ à $Suivant(T')$
- $T' \to *FT'$ et $T' \to \epsilon$ donc on ajoute $Suivant(T')$ à $Suivant(F)$

Reprenons la grammaire :
$E \rightarrow TE'$
$E' \rightarrow +TE' \mid \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT' \mid \epsilon$
$F \rightarrow (E) \mid id$
$Premier(E) = \{(, id\}$
$Premier(E') = \{\epsilon, +\}$
$Premier(T) = \{(, id\}$
$Premier(T') = \{\epsilon, *\}$
$Premier(F) = \{(, id\}$
$Suivant(E) = \{\$,)\}$
$Suivant(E') = \{\}$
$Suivant(T) = \{+\}$
$Suivant(T') = \{\}$
$Suivant(F) = \{*\}$

- E est l'axiome donc on ajoute \$ à $Suivant(E)$
- $E \rightarrow TE'$ donc on ajoute $Premier(E') \setminus \{\epsilon\}$ à $Suivant(T)$
- $T \rightarrow FT'$ donc on ajoute $Premier(T') \setminus \{\epsilon\}$ à $Suivant(F)$
- $F \rightarrow (E)$ donc on ajoute) à $Suivant(E)$
- $E \rightarrow TE'$ donc on ajoute $Suivant(E)$ à $Suivant(E')$
- $E' \rightarrow +TE'$ et $E' \rightarrow \epsilon$ donc on ajoute $Suivant(E')$ à $Suivant(T)$
- $T \rightarrow FT'$ donc on ajoute $Suivant(T)$ à $Suivant(T')$
- $T' \rightarrow *FT'$ et $T' \rightarrow \epsilon$ donc on ajoute $Suivant(T')$ à $Suivant(F)$

Reprenons la grammaire :

$E \rightarrow TE'$
$E' \rightarrow +TE' \mid \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT' \mid \epsilon$
$F \rightarrow (E) \mid id$
$Premier(E) = \{(, id\}$
$Premier(E') = \{\epsilon, +\}$
$Premier(T) = \{(, id\}$
$Premier(T') = \{\epsilon, *\}$
$Premier(F) = \{(, id\}$
$Suivant(E) = \{\$,)\}$
$Suivant(E') = \{\$,)\}$
$Suivant(T) = \{+\}$
$Suivant(T') = \{\}$
$Suivant(F) = \{*\}$

- E est l'axiome donc on ajoute \$ à $Suivant(E)$
- $E \rightarrow TE'$ donc on ajoute $Premier(E') \setminus \{\epsilon\}$ à $Suivant(T)$
- $T \rightarrow FT'$ donc on ajoute $Premier(T') \setminus \{\epsilon\}$ à $Suivant(F)$
- $F \rightarrow (E)$ donc on ajoute) à $Suivant(E)$
- $E \rightarrow TE'$ donc on ajoute $Suivant(E)$ à $Suivant(E')$
- $E' \rightarrow +TE'$ et $E' \rightarrow \epsilon$ donc on ajoute $Suivant(E')$ à $Suivant(T)$
- $T \rightarrow FT'$ donc on ajoute $Suivant(T)$ à $Suivant(T')$
- $T' \rightarrow *FT'$ et $T' \rightarrow \epsilon$ donc on ajoute $Suivant(T')$ à $Suivant(F)$

Reprenons la grammaire :

$E \rightarrow TE'$

$E' \rightarrow +TE' \mid \epsilon$

$T \rightarrow FT'$

$T' \rightarrow *FT' \mid \epsilon$

$F \rightarrow (E) \mid id$

$Premier(E) = \{(, id\}$

$Premier(E') = \{\epsilon, +\}$

$Premier(T) = \{(, id\}$

$Premier(T') = \{\epsilon, *\}$

$Premier(F) = \{(, id\}$

$Suivant(E) = \{\$,)\}$

$Suivant(E') = \{\$,)\}$

$Suivant(T) = \{+, \$,)\}$

$Suivant(T') = \{\}$

$Suivant(F) = \{*\}$

- E est l'axiome donc on ajoute \$ à $Suivant(E)$
- $E \rightarrow TE'$ donc on ajoute $Premier(E') \setminus \{\epsilon\}$ à $Suivant(T)$
- $T \rightarrow FT'$ donc on ajoute $Premier(T') \setminus \{\epsilon\}$ à $Suivant(F)$
- $F \rightarrow (E)$ donc on ajoute) à $Suivant(E)$
- $E \rightarrow TE'$ donc on ajoute $Suivant(E)$ à $Suivant(E')$
- $E' \rightarrow +TE'$ et $E' \rightarrow \epsilon$ donc on ajoute $Suivant(E')$ à $Suivant(T)$
- $T \rightarrow FT'$ donc on ajoute $Suivant(T)$ à $Suivant(T')$
- $T' \rightarrow *FT'$ et $T' \rightarrow \epsilon$ donc on ajoute $Suivant(T')$ à $Suivant(F)$

Reprenons la grammaire :

$E \rightarrow TE'$

$E' \rightarrow +TE' \mid \epsilon$

$T \rightarrow FT'$

$T' \rightarrow *FT' \mid \epsilon$

$F \rightarrow (E) \mid id$

$Premier(E) = \{(, id\}$

$Premier(E') = \{\epsilon, +\}$

$Premier(T) = \{(, id\}$

$Premier(T') = \{\epsilon, *\}$

$Premier(F) = \{(, id\}$

$Suivant(E) = \{\$,)\}$

$Suivant(E') = \{\$,)\}$

$Suivant(T) = \{+, \$,)\}$

$Suivant(T') = \{+, \$,)\}$

$Suivant(F) = \{*\}$

- E est l'axiome donc on ajoute \$ à $Suivant(E)$
- $E \rightarrow TE'$ donc on ajoute $Premier(E') \setminus \{\epsilon\}$ à $Suivant(T)$
- $T \rightarrow FT'$ donc on ajoute $Premier(T') \setminus \{\epsilon\}$ à $Suivant(F)$
- $F \rightarrow (E)$ donc on ajoute) à $Suivant(E)$
- $E \rightarrow TE'$ donc on ajoute $Suivant(E)$ à $Suivant(E')$
- $E' \rightarrow +TE'$ et $E' \rightarrow \epsilon$ donc on ajoute $Suivant(E')$ à $Suivant(T)$
- $T \rightarrow FT'$ donc on ajoute $Suivant(T)$ à $Suivant(T')$
- $T' \rightarrow *FT'$ et $T' \rightarrow \epsilon$ donc on ajoute $Suivant(T')$ à $Suivant(F)$

Reprenons la grammaire :
$E \rightarrow TE'$
$E' \rightarrow +TE' \mid \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT' \mid \epsilon$
$F \rightarrow (E) \mid id$
$Premier(E) = \{(, id\}$
$Premier(E') = \{\epsilon, +\}$
$Premier(T) = \{(, id\}$
$Premier(T') = \{\epsilon, *\}$
$Premier(F) = \{(, id\}$
$Suivant(E) = \{\$,)\}$
$Suivant(E') = \{\$,)\}$
$Suivant(T) = \{+, \$,)\}$
$Suivant(T') = \{+, \$,)\}$
$Suivant(F) = \{*, +, \$,)\}$

- E est l'axiome donc on ajoute \$ à $Suivant(E)$
- $E \rightarrow TE'$ donc on ajoute $Premier(E') \setminus \{\epsilon\}$ à $Suivant(T)$
- $T \rightarrow FT'$ donc on ajoute $Premier(T') \setminus \{\epsilon\}$ à $Suivant(F)$
- $F \rightarrow (E)$ donc on ajoute) à $Suivant(E)$
- $E \rightarrow TE'$ donc on ajoute $Suivant(E)$ à $Suivant(E')$
- $E' \rightarrow +TE'$ et $E' \rightarrow \epsilon$ donc on ajoute $Suivant(E')$ à $Suivant(T)$
- $T \rightarrow FT'$ donc on ajoute $Suivant(T)$ à $Suivant(T')$
- $T' \rightarrow *FT'$ et $T' \rightarrow \epsilon$ donc on ajoute $Suivant(T')$ à $Suivant(F)$

Construction de l'analyseur

L'algorithme suivant peut être utilisé pour construire une table pour l'analyse prédictive d'une grammaire G. L'idée sous-tendant cet algorithme est le suivant.

- Soit $A \rightarrow \alpha$ une production et a dans $Premier(\alpha)$.
- L'analyseur développe A en α chaque fois que le symbole d'entrée courant est a.
- Une production se produit quand $\alpha = \epsilon$ ou $\alpha \Rightarrow^* \epsilon$. Dans ce cas, nous devons également développer A en α si le symbole d'entrée courant est dans $Suivant(A)$ ou si le \$ d'entrée a été atteint et que \$ est dans $Suivant(A)$.

Algorithme

Construction d'une table d'analyse prédictive :
Donnée : une grammaire G.
Résultat : une table d'analyse M pour G.
Méthode :

- Pour chaque production $A \rightarrow \alpha$ de la grammaire :
  - Pour chaque terminal a dans $Premier(\alpha)$, ajouter $A \rightarrow \alpha$ à $M[A, a]$
  - Si ϵ est dans $Premier(\alpha)$:
    - Ajouter $A \rightarrow \alpha$ à $M[A, b]$ pour chaque terminal b dans $Suivant(A)$.
    - Si \$ est dans $Suivant(A)$, ajouter $A \rightarrow \alpha$ à $M[A, \$]$.
- Faire de chaque entrée indéfinie de M une erreur.

Exemple

$E \to TE'$
$E' \to +TE'$
$E' \to \epsilon$
$T \to FT'$
$T' \to *FT'$
$T' \to \epsilon$
$F \to (E)$
$F \to id$

$Premier(E) = \{(, id\}$
$Premier(E') = \{\epsilon, +\}$
$Premier(T) = \{(, id\}$
$Premier(T') = \{\epsilon, *\}$
$Premier(F) = \{(, id\}$

$Suivant(E) = \{\$,)\}$
$Suivant(E') = \{\$,)\}$
$Suivant(T) = \{+, \$,)\}$
$Suivant(T') = \{+, \$,)\}$
$Suivant(F) = \{*, +, \$,)\}$

Non terminal	Symbole d'entrée					
	id	$+$	$*$	(	)	\$
E	$E \to TE'$			$E \to TE'$		
E'		$E' \to +TE'$			$E' \to \epsilon$	$E' \to \epsilon$
T	$T \to FT'$			$T \to FT'$		
T'		$T' \to \epsilon$	$T' \to *FT'$		$T' \to \epsilon$	$T' \to \epsilon$
F	$F \to id$			$F \to (E)$		

Exemple

$E \rightarrow TE'$
$E' \rightarrow +TE'$
$E' \rightarrow \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT'$
$T' \rightarrow \epsilon$
$F \rightarrow (E)$
$F \rightarrow id$

$Premier(E) = \{(, id\}$
$Premier(E') = \{\epsilon, +\}$
$Premier(T) = \{(, id\}$
$Premier(T') = \{\epsilon, *\}$
$Premier(F) = \{(, id\}$

$Suivant(E) = \{\$,)\}$
$Suivant(E') = \{\$,)\}$
$Suivant(T) = \{+, \$,)\}$
$Suivant(T') = \{+, \$,)\}$
$Suivant(F) = \{*, +, \$,)\}$

Non terminal	Symbole d'entrée					
	id	$+$	$*$	(	)	\$
E	$E \rightarrow TE'$			$E \rightarrow TE'$		
E'		$E' \rightarrow +TE'$			$E' \rightarrow \epsilon$	$E' \rightarrow \epsilon$
T	$T \rightarrow FT'$			$T \rightarrow FT'$		
T'		$T' \rightarrow \epsilon$	$T' \rightarrow *FT'$		$T' \rightarrow \epsilon$	$T' \rightarrow \epsilon$
F	$F \rightarrow id$			$F \rightarrow (E)$		

Exemple

$E \rightarrow TE'$
$E' \rightarrow +TE'$
$E' \rightarrow \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT'$
$T' \rightarrow \epsilon$
$F \rightarrow (E)$
$F \rightarrow id$

$$\begin{aligned}
Premier(E) &= \{(, id\} \\
Premier(E') &= \{\epsilon, +\} \\
Premier(T) &= \{(, id\} \\
Premier(T') &= \{\epsilon, *\} \\
Premier(F) &= \{(, id\}
\end{aligned}$$

$$\begin{aligned}
Suivant(E) &= \{\$,)\} \\
Suivant(E') &= \{\$,)\} \\
Suivant(T) &= \{+, \$,)\} \\
Suivant(T') &= \{+, \$,)\} \\
Suivant(F) &= \{*, +, \$,)\}
\end{aligned}$$

Non terminal	Symbole d'entrée					
	id	$+$	$*$	(	)	\$
E	$E \rightarrow TE'$			$E \rightarrow TE'$		
E'		$E' \rightarrow +TE'$			$E' \rightarrow \epsilon$	$E' \rightarrow \epsilon$
T	$T \rightarrow FT'$			$T \rightarrow FT'$		
T'		$T' \rightarrow \epsilon$	$T' \rightarrow *FT'$		$T' \rightarrow \epsilon$	$T' \rightarrow \epsilon$
F	$F \rightarrow id$			$F \rightarrow (E)$		

Exemple

$E \rightarrow TE'$
$E' \rightarrow +TE'$
$E' \rightarrow \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT'$
$T' \rightarrow \epsilon$
$F \rightarrow (E)$
$F \rightarrow id$

$Premier(E) = \{(, id\}$
$Premier(E') = \{\epsilon, +\}$
$Premier(T) = \{(, id\}$
$Premier(T') = \{\epsilon, *\}$
$Premier(F) = \{(, id\}$

$Suivant(E) = \{\$,)\}$
$Suivant(E') = \{\$,)\}$
$Suivant(T) = \{+, \$,)\}$
$Suivant(T') = \{+, \$,)\}$
$Suivant(F) = \{*, +, \$,)\}$

Non terminal	Symbole d'entrée					
	id	$+$	$*$	(	)	\$
E	$E \rightarrow TE'$			$E \rightarrow TE'$		
E'		$E' \rightarrow +TE'$			$E' \rightarrow \epsilon$	$E' \rightarrow \epsilon$
T	$T \rightarrow FT'$			$T \rightarrow FT'$		
T'		$T' \rightarrow \epsilon$	$T' \rightarrow *FT'$		$T' \rightarrow \epsilon$	$T' \rightarrow \epsilon$
F	$F \rightarrow id$			$F \rightarrow (E)$		

Exemple

$E \rightarrow TE'$
$E' \rightarrow +TE'$
$E' \rightarrow \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT'$
$T' \rightarrow \epsilon$
$F \rightarrow (E)$
$F \rightarrow id$

$Premier(E) = \{(, id\}$
$Premier(E') = \{\epsilon, +\}$
$Premier(T) = \{(, id\}$
$Premier(T') = \{\epsilon, *\}$
$Premier(F) = \{(, id\}$

$Suivant(E) = \{\$,)\}$
$Suivant(E') = \{\$,)\}$
$Suivant(T) = \{+, \$,)\}$
$Suivant(T') = \{+, \$,)\}$
$Suivant(F) = \{*, +, \$,)\}$

Non terminal	Symbole d'entrée					
	id	$+$	$*$	$($	$)$	$\$$
E	$E \rightarrow TE'$			$E \rightarrow TE'$		
E'		$E' \rightarrow +TE'$			$E' \rightarrow \epsilon$	$E' \rightarrow \epsilon$
T	$T \rightarrow FT'$			$T \rightarrow FT'$		
T'		$T' \rightarrow \epsilon$	$T' \rightarrow *FT'$		$T' \rightarrow \epsilon$	$T' \rightarrow \epsilon$
F	$F \rightarrow id$			$F \rightarrow (E)$		

Exemple

$E \rightarrow TE'$
$E' \rightarrow +TE'$
$E' \rightarrow \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT'$
$T' \rightarrow \epsilon$
$F \rightarrow (E)$
$F \rightarrow id$

$$Premier(E) = \{(, id\}$$
$$Premier(E') = \{\epsilon, +\}$$
$$Premier(T) = \{(, id\}$$
$$Premier(T') = \{\epsilon, *\}$$
$$Premier(F) = \{(, id\}$$

$$Suivant(E) = \{\$,)\}$$
$$Suivant(E') = \{\$,)\}$$
$$Suivant(T) = \{+, \$,)\}$$
$$Suivant(T') = \{+, \$,)\}$$
$$Suivant(F) = \{*, +, \$,)\}$$

Non terminal	Symbole d'entrée					
	id	$+$	$*$	(	)	\$
E	$E \rightarrow TE'$			$E \rightarrow TE'$		
E'		$E' \rightarrow +TE'$			$E' \rightarrow \epsilon$	$E' \rightarrow \epsilon$
T	$T \rightarrow FT'$			$T \rightarrow FT'$		
T'		$T' \rightarrow \epsilon$	$T' \rightarrow *FT'$		$T' \rightarrow \epsilon$	$T' \rightarrow \epsilon$
F	$F \rightarrow id$			$F \rightarrow (E)$		

Exemple

$E \rightarrow TE'$
$E' \rightarrow +TE'$
$E' \rightarrow \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT'$
$T' \rightarrow \epsilon$
$F \rightarrow (E)$
$F \rightarrow id$

$Premier(E) = \{(, id\}$
$Premier(E') = \{\epsilon, +\}$
$Premier(T) = \{(, id\}$
$Premier(T') = \{\epsilon, *\}$
$Premier(F) = \{(, id\}$

$Suivant(E) = \{\$,)\}$
$Suivant(E') = \{\$,)\}$
$Suivant(T) = \{+, \$,)\}$
$Suivant(T') = \{+, \$,)\}$
$Suivant(F) = \{*, +, \$,)\}$

Non terminal	Symbole d'entrée					
	id	$+$	$*$	(	)	\$
E	$E \rightarrow TE'$			$E \rightarrow TE'$		
E'		$E' \rightarrow +TE'$			$E' \rightarrow \epsilon$	$E' \rightarrow \epsilon$
T	$T \rightarrow FT'$			$T \rightarrow FT'$		
T'		$T' \rightarrow \epsilon$	$T' \rightarrow *FT'$		$T' \rightarrow \epsilon$	$T' \rightarrow \epsilon$
F	$F \rightarrow id$			$F \rightarrow (E)$		

Exemple

$E \rightarrow TE'$
$E' \rightarrow +TE'$
$E' \rightarrow \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT'$
$T' \rightarrow \epsilon$
$F \rightarrow (E)$
$F \rightarrow id$

$Premier(E) = \{(, id\}$
$Premier(E') = \{\epsilon, +\}$
$Premier(T) = \{(, id\}$
$Premier(T') = \{\epsilon, *\}$
$Premier(F) = \{(, id\}$

$Suivant(E) = \{\$,)\}$
$Suivant(E') = \{\$,)\}$
$Suivant(T) = \{+, \$,)\}$
$Suivant(T') = \{+, \$,)\}$
$Suivant(F) = \{*, +, \$,)\}$

Non terminal	Symbole d'entrée					
	id	$+$	$*$	(	)	\$
E	$E \rightarrow TE'$			$E \rightarrow TE'$		
E'		$E' \rightarrow +TE'$			$E' \rightarrow \epsilon$	$E' \rightarrow \epsilon$
T	$T \rightarrow FT'$			$T \rightarrow FT'$		
T'		$T' \rightarrow \epsilon$	$T' \rightarrow *FT'$		$T' \rightarrow \epsilon$	$T' \rightarrow \epsilon$
F	$F \rightarrow id$			$F \rightarrow (E)$		

Exemple

$E \rightarrow TE'$
$E' \rightarrow +TE'$
$E' \rightarrow \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT'$
$T' \rightarrow \epsilon$
$F \rightarrow (E)$
$F \rightarrow id$

$Premier(E) = \{(, id\}$
$Premier(E') = \{\epsilon, +\}$
$Premier(T) = \{(, id\}$
$Premier(T') = \{\epsilon, *\}$
$Premier(F) = \{(, id\}$

$Suivant(E) = \{\$,)\}$
$Suivant(E') = \{\$,)\}$
$Suivant(T) = \{+, \$,)\}$
$Suivant(T') = \{+, \$,)\}$
$Suivant(F) = \{*, +, \$,)\}$

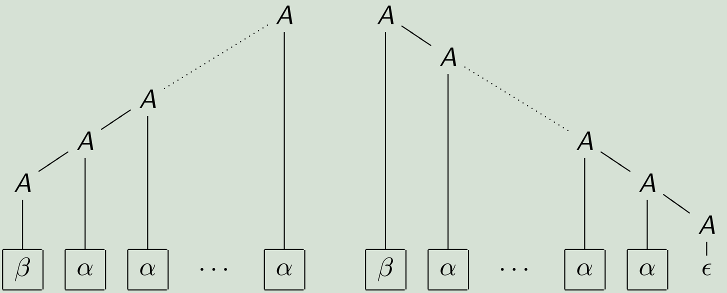
Non terminal	Symbole d'entrée					
	id	$+$	$*$	(	)	\$
E	$E \rightarrow TE'$			$E \rightarrow TE'$		
E'		$E' \rightarrow +TE'$			$E' \rightarrow \epsilon$	$E' \rightarrow \epsilon$
T	$T \rightarrow FT'$			$T \rightarrow FT'$		
T'		$T' \rightarrow \epsilon$	$T' \rightarrow *FT'$		$T' \rightarrow \epsilon$	$T' \rightarrow \epsilon$
F	$F \rightarrow id$			$F \rightarrow (E)$		

Suppression de la récursivité à gauche et factorisation à gauche

Nous étudions les transformations qui permettent de réécrire des grammaires de façon à les rendre aptes une analyse descendante.
Une grammaire est récursive à gauche si elles contient un non-terminal A tel qu'il existe une dérivation $A \Rightarrow^{+} A\alpha$, où α est une chaîne quelconque.
Les méthodes d'analyse descendante ne peuvent pas fonctionner avec les grammaires récursives à gauche. On a besoin d'une transformation grammaticale qui élimine cette récursivité à gauche.

On peut éliminer une production récursive à gauche en la réécrivant. Considérons le non-terminal A dans les deux productions : $A \rightarrow A\alpha \mid \beta$ où α et β sont des suites de terminaux et de non-terminaux qui ne commencent pas par A.

Le non-terminal A est récursif à gauche car la production $A \rightarrow A\alpha$ contient A comme symbole le plus à gauche de sa partie droite. Une application répétée de cette production fabrique une suite de α à droite de A, comme sur la figure ci-dessous. Quand A est finalement remplacé par β, on obtient un β et une suite éventuellement vide de α.



Exemple

On peut obtenir cet effet sur l'exemple précédent en réécrivant les productions $A \longrightarrow A\alpha \mid \beta$ de cette manière :

$A \longrightarrow \beta A'$

$A' \longrightarrow \alpha A' \mid \epsilon$

On remarque qu'on a un nouveau non-terminal : A'. La production

$A' \longrightarrow \alpha A'$ est récursive à droite car cette production définissant A' contient A' comme dernier symbole en partie droite.

Les productions récursives à droite conduisent à des arbres qui grandissent vers la droite comme sur la figure précédente.

Exemple complet

Considérons lcette grammaire représentant les expression arithmériques :

$E \rightarrow E + T \mid T$
$T \rightarrow T * F \mid F$
$F \rightarrow (E) \mid id$

En éliminant les récursivités gauche immédiates (productions de la forme $A \rightarrow A\alpha$) des productions pour E puis pour T, nous obtenons :

$E \rightarrow TE'$
$E' \rightarrow +TE' \mid \epsilon$
$T \rightarrow FT'$
$T' \rightarrow *FT' \mid \epsilon$
$F \rightarrow (E) \mid id$

Tiens on la connaît celle là.

Quel que soit le nombre de productions à partir de A (A-productions), il est possible d'éliminer les récursivités à gauche immédiates par la technique suivante.

Technique

Dans un premier temps, on groupe les A-productions comme suit : $A \rightarrow A\alpha_1 \mid A\alpha_2 \mid \ldots \mid A\alpha_n \mid \beta_1 \mid \beta_2 \mid \ldots \mid \beta_m$ où aucun β_i ne commence par un A. Puis on remplace les A-productions par : $A \rightarrow \beta_1 A' \mid \beta_2 A' \mid \ldots \mid \beta_m A'$ $A' \rightarrow \alpha_1 A' \mid \alpha_2 A' \mid \ldots \mid \alpha_n A' \mid \epsilon_m$

- Le non-terminal A engendre les mêmes chaînes que précédemment mais il n'est plus récursif à gauche.
- Cette procédure élimine toutes les récursivités à gauches immédiates des productions A et A' (en supposant qu'aucun des α_i ne soit ϵ), mais il n'élimine pas les récursivités à gauche impliquant des dérivations contenant au moins deux étapes.

Exemple

Par exemple avec la grammaire :
$S \longrightarrow Aa \mid b$
$A \longrightarrow Ac \mid Sd \mid \epsilon$
Le non-terminal S est récursif à gauche car $S \Rightarrow Aa \Rightarrow Sda$, mais il n'est pas immédiatement récursif à gauche.

Algorithme

Élimination des récursivités à gauche
Donnée : une grammaire G sans cycle et sans production vide.
Résultat : une grammaire équivalente sans récursivité à gauche.
Méthode : appliquer l'algorithme suivant.
Remarque : la grammaire résultante peut avoir des productions vides.

- On ordonne les non-terminaux $A_1, A_2, \ldots, A_n$
- Pour i de 1 à n faire
  - Pour j de 1 à $i-1$ faire
    - Remplacer chaque production de la forme $A_i \longrightarrow A_j\gamma$ par les productions $A_i \longrightarrow \delta_1\gamma \mid \delta_2\gamma \mid \ldots \mid \delta_k\gamma$, où $A_j \longrightarrow \delta_1 \mid \delta_2 \mid \ldots \mid \delta_k$ sont toutes les A_j-productions courantes
  - Éliminer les récursivités à gauche immédiates des A_i-productions.

Explications

La raison pour laquelle la procédure précédente produit l'effet voulu est qu'après la $(i-1)^e$ itération de la boucle *Pour* la plus externe (sur i), chaque production de la forme $A_k \longrightarrow A_l\alpha$, où $k < i$ doit être telle que $l > k$.

Il en résulte qu'à l'itération suivante la boucle interne (sur j) élève progressivement la limite inférieure sur m dans toutes les productions de la forme $A_i \longrightarrow A_m\alpha$, jusqu'à ce que l'on ait $m \geq i$.

L'élimination à gauche immédiate sur les A_i-productions force donc m à devenir supérieur à i.

Exemple

$S \longrightarrow Aa \mid b$
$A \longrightarrow Ac \mid Sd$

- On collecte les non-terminaux $\{S, A\}$.
- $i = 1$: aucune récursivité à gauche immédiate dans les S-productions.
- $i = 2$: on substitue la production $A \longrightarrow Sd$ par $A \longrightarrow Aad \mid bd$
- On élimine les récursivités à gauche immédiates des A-productions et on obtient la grammaire suivante :

$S \longrightarrow Aa \mid b$
$A \longrightarrow bdA'$
$A' \longrightarrow cA' \mid adA' \mid \epsilon$

Factorisation à gauche

- La factorisation à gauche est une transformation grammaticale utile pour obtenir une grammaire convenant à l'analyse prédictive.
- L'idée de base est que, pour développer un non-terminal A, quand il n'est pas évident de choisir l'alternative à utiliser, on doit réécrire les A-productions, de façon à différer la décision jusqu'à ce que suffisament de texte ait été lu pour faire le bon choix.
- Par exemple si nous avons les deux productions :
  inst $\longrightarrow$ **si** *expr* **alors** *inst* **sinon** *inst*
  inst $\longrightarrow$ **si** *expr* **alors** *inst*
  Nous ne pouvons pas, à la vue de l'inité lexicale **si**, déterminer immédiatement laquelle des productions utiliser pour développer *inst*.

- En général si $A \rightarrow \alpha\beta_1 \mid \alpha\beta_2$ sont deux A-productions et si l'entrée commence par une chaîne non vide dérivée de α, nous ne savons pas s'il faut développer A en $\alpha\beta_1$ ou en A en $\alpha\beta_2$.
- Il est possible de différer cette décision en étendant A en $\alpha A'$. Alors après avoir lu la chaîne dérivée de α, nous étendons A' en β_1 ou en β_2.
- Nous venons de réaliser une factorisation à gauche.
- Les productions d'origine $A \rightarrow \alpha\beta_1 \mid \alpha\beta_2$ deviennent :
  $A \rightarrow \alpha A'$
  $A' \rightarrow \beta_1 \mid \beta_2$
  où A' est un nouveau non-terminal.

Algorithme

Factorisation à gauche d'une grammaire.
Donnée : une grammaire G.
Résultat : une grammaire équivalente factorisée à gauche.
Méthode :

- Pour chaque non-terminal A, trouver le plus long préfixe α commun à deux de ses alternatives ou plus.
- Si $\alpha \neq \epsilon$, c'est-à-dire s'il y a un préfixe commun non-trivial, remplacer toutes les A-productions $A \rightarrow \alpha\beta_1 \mid \alpha\beta_2 \mid \ldots \mid \alpha\beta_n \mid \gamma$ où γ représente toutes les alternatives qui ne commencent pas par α, par :
  $A \rightarrow \alpha A' \mid \gamma$
  $A' \rightarrow \beta_1 \mid \beta_2 \mid \ldots \mid \beta_n$
  Avec A' un nouveau non-terminal. On applique cette transformation de manière répétitive jusqu'à ce qu'aucune des alternatives d'un même non-terminal n'ait de préfixe commun.

Exemple

Soit la grammaire :

$S \longrightarrow abS \mid abA \mid aB$

$A \longrightarrow c$

$B \longrightarrow d$

On fait une itération avec $\alpha = ab$ et on obtient la grammaire :

$S \longrightarrow abS' \mid aB$

$S' \longrightarrow S \mid A$

$A \longrightarrow c$

$B \longrightarrow d$

On simplifie une nouvelle fois avec $\alpha = a$ et on obtient la grammaire :

$S \longrightarrow aS''$

$S'' \longrightarrow bS' \mid B$

$S' \longrightarrow S \mid A$

$A \longrightarrow c$

$B \longrightarrow d$

Grammaires LL(1)

L'algorithe de construction d'une table d'analyse prédictive peut être appliqué à une grammaire G pour produire une table d'analyse M. Cependant, pour certaines grammaires, M peut avoir des entrées qui sont définies de façon multiple. Par exemple si G est récursive à gauche ou ambigüe, alors M aura au moins une de ses entrées définie de façon multiple.

Exemple

Soit la grammaire :

$S \longrightarrow iEtSS' \mid a$

$S' \longrightarrow eS \mid \epsilon$

$E \longrightarrow b$

Voici la table d'analyse pour cette grammaire :

Non terminal	Symbole d'entrée					
	a	b	e	i	t	$\$$
S	$S \longrightarrow a$			$S \longrightarrow iEtSS'$		
S'			$S' \longrightarrow \epsilon$ $S' \longrightarrow eS$			$S' \longrightarrow \epsilon$
E		$E \longrightarrow b$				

Exemple (suite)

L'entrée $M[S', e]$ contient à la fois $S' \rightarrow eS$ et $S \rightarrow \epsilon$, car $Suivant(S') = \{e, \$\}$. La grammaire est ambigüe et cette ambiguïté se manifeste par un choix sur la production à utiliser quand on voit un e.

Grammaire LL(1)

Une grammaire dont la table d'analyse n'a aucune entrée définie de façon multiple est appelée **LL(1)**.

- Le premier L de **LL(1)** signifie « parcours de l'entrée de la gauche vers la droite »(left to right scanning).
- Le second L de **LL(1)** signifie « dérivation gauche »(leftmost derivation).
- Le 1 indique qu'on utilise un seul symbole d'entrée de prévision à chaque étape nécessitant la prise d'une décision d'action d'analyse.

On peut montrer que l'algorithme de construction d'une table d'analyse prédictive produit pour toute grammaire **LL(1)** G une table qui analyse toutes les phrases de G et seulement celles là.

Propriétés

Les grammaires **LL(1)** ont un certain nombre de propriétés distinctives :

- Aucune grammaire ambigüe ou récursive à gauche ne peut être **LL(1)**.
- Une grammaire G est **LL(1)** si et seulement si chaque fois que $A \rightarrow \alpha \mid \beta$ sont deux productions distinctes de G, les conditions suivantes s'appliquent :
  1. Pour aucun terminal A, α, β ne se dérivent toutes les deux en commençant par un a.
  2. Au plus une des chaînes α et β peut se dériver en la chaîne vide.
  3. Si $\beta \Rightarrow^* \epsilon$, α ne se dérive pas en une chaîne commençant par un terminal de $Suivant(A)$.

Remarques

- Que fait-on si la table d'analyse a des entrées à valeur multiple ? $\Rightarrow$ On peut tenter d'éliminer les récursivités à gauche puis de factoriser à gauche quand c'est possible. Avec un peu de chance on obtient une table qui ne contient plus de telles entrées.
- Malheureusement il existe des grammaires pour lesquelles il n'existe pas de transformations les rendant **LL(1)**.
- Pour un langage donné, la plus grande difficulté lors de l'utilisation d'une analyse prédictive est d'écrire une grammaire permettant de construire un analyseur prédictif.

Remarques

- Bien que l'élimination des récursivités à gauche et la factorisation à gauche soient aisées à réaliser, elles rendent la grammaire résultante difficile à lire et à utiliser pour la traduction.
- Pour réduire ces difficultés, une solution consiste souvent pour un nanlyseur dans un compilateur à utiliser l'analyse prédictive pour les constructions contrôlant le flot de données et à utiliser la précédence d'opérateurs pour les expressions.
- Si on a un constructeur d'analyseur **LR**, tous les bénéfices de l'analyse prédictive et de la précédence d'opérateurs peuvent être obtenus automatiquement.