

Werkcollege Compilerconstructie
Woensdag 28 september 2016

1. Beschouw de context-vrije grammatica G' met startvariabele S en de volgende producties:

$$\begin{aligned} S &\rightarrow \mathbf{if} B S \mid \mathbf{id} = \mathbf{id} \\ B &\rightarrow EB' \\ B' &\rightarrow \mathbf{boolop} EB' \mid \epsilon \\ E &\rightarrow \mathbf{id} E' \\ E' &\rightarrow \mathbf{relop} \mathbf{id} \mid \epsilon \end{aligned}$$

In G' zijn S, B, B', E, E' dus de variabelen en **if, id, =, boolop, relop** de terminalen. G' is het resultaat van opgave 5a) uit werkcollege 3,

- (a) Bepaal voor elke variabele in de grammatica G' zowel de FIRST- als de FOLLOW-verzameling.
 - (b) Construeer de LR(0)-automaat bij grammatica G' .
 - (c) Construeer de SLR *parsing table* bij grammatica G' .
 - (d) Is G' een SLR grammatica? Motiveer je antwoord.
2. Verzin een voorbeeld van een *ondubbelzinnige* context-vrije grammatica G waarbij in de SLR *parsing table* reduce/reduce conflicten ontstaan.
3. Bij het opbouwen van een SLR *parsing table* is een van de regels:

Als $[A \rightarrow \alpha \cdot]$ in I_i zit en $a \in \text{FOLLOW}(A)$, voeg dan de actie 'reduce naar $A \rightarrow \alpha$ ' toe aan ACTION $[i, a]$.

Verzin een voorbeeld van een context-vrije grammatica G , zódat

- de LR(0)-automaat een itemset I_i bevat met daarin een item $[A \rightarrow \alpha \cdot]$,
- en er een terminal $a \in \text{FOLLOW}(A)$ is,
- terwijl je door de reductie naar $A \rightarrow \alpha$ (en eventueel daarna nog andere reducties) uit te voeren, toch niet in een itemset I_j kunt komen vanwaaruit er een transitie met een a is.

4. Beschouw de context-vrije grammatica G met startvariabele S en de volgende producties:

- (1) $S \rightarrow S - A$
- (2) $S \rightarrow A$
- (3) $A \rightarrow AB$
- (4) $A \rightarrow B$
- (5) $B \rightarrow B+$
- (6) $B \rightarrow p$
- (7) $B \rightarrow q$

- (a) Geef (ad hoc) een afleidingsboom in G voor de string $q + p$.
- (b) Gegeven is dat de SLR parsing table bij grammatica G er als volgt uit ziet:

State	Action					Goto		
	-	+	p	q	$\$$	S	A	B
0			s6	s7		1	9	8
1	s2				acc			
2			s6	s7			3	8
3	r1		s6	s7	r1			4
4	r3	s5	r3	r3	r3			
5	r5	r5	r5	r5	r5			
6	r6	r6	r6	r6	r6			
7	r7	r7	r7	r7	r7			
8	r4	s5	r4	r4	r4			
9	r2		s6	s7	r2			4

Parse de string $q + p$ met deze tabel. Laat bij iedere stap duidelijk zien wat je doet, bijvoorbeeld met behulp van een tabel van de volgende vorm:

States on stack	Corresponding Symbols on stack	Input	Action
...