

Esercizio 15

Sia L il linguaggio definito dalla seguente espressione su insiemi:

$$L = \{ a^i a^j b^j \mid i, j \geq 0 \text{ e se } i, j > 0 \text{ allora } i \leq j \}.$$

- a) Si definisca una grammatica (non ambigua e context free) che lo generi
- b) Si diano una grammatica e un analizzatore ascendente deterministico per il linguaggio
- c) Si diano una grammatica e un analizzatore discendente deterministico per il linguaggio

Esercizio 15a

$$\begin{aligned} S &::= S' \mid \varepsilon \\ S' &::= a a S' b \mid A \mid B \\ A &::= a a A \mid a a \\ B &::= b B \mid b \end{aligned}$$

Esercizio 15b

Potremmo partire dalla grammatica non ambigua e context-free data nella parte (a), ed esaminarla per un'analisi LR.

$$\begin{aligned} S &::= S1 \mid \varepsilon \\ S1 &::= a a S1 b \mid A \mid B \\ A &::= a a A \mid a a \\ B &::= b B \mid b \end{aligned}$$

Questa grammatica non è LR(k) per nessun k. Infatti, la stringa "aa" è viable prefix sia per maniglie $S1 ::= aaS1b$, sia per $A ::= aa$. Purtroppo il contesto di lookahead che può seguire $S1$ è esattamente lo stesso che può seguire A (notare la produzione $S1 ::= A$). Potete verificare facilmente che la collezione LR(1) ha un conflitto shift/reduce nello stato raggiunto, a partire da I_0 , dopo la lettura di "aa" (items $B \rightarrow \cdot bB$, $A \rightarrow aa \cdot$). In realtà, tale contesto di lookahead è della forma b^n , con n arbitrario naturale. Quindi anche utilizzando $k \gg 1$ simboli di lookahead, troveremmo che per stringhe con suffisso b^n per $n > k$, l'analizzatore LR(k) ha un conflitto.

Una più attenta osservazione delle ragioni di tale fallimento dovrebbe farci osservare strette analogie tra il linguaggio di questo esercizio e il linguaggio dell'[esercizio 4](#). In definitiva dobbiamo introdurre una grammatica che definisca il

linguaggio in modo tale che l'analisi ascendente di stringhe della forma $(aa)^h(aa)^i b^i$ riconosca prima le coppie annidate $(aa)^i b^i$ e dopo il prefisso $(aa)^h$. Riscriviamo l'espressione di insiemi che definisce il linguaggio evidenziando questi aspetti:

$$\begin{aligned} L(S) &= \{\varepsilon\} \cup L(A) \cup L(B) \\ L(A) &= \{(aa)^h (aa)^i b^i \mid i \geq 0, h > 0\} \\ L(B) &= \{(aa)^i b^i b^h \mid i \geq 0, h > 0\} \end{aligned}$$

Otteniamo la grammatica seguente [\(oppure\)](#):

$$\begin{aligned} S &::= aaA \mid B \mid \varepsilon \\ A &::= aaA \mid C \quad \text{le produzioni di } A \text{ consentono di avere come prima maniglia } C \\ C &::= aa C b \mid \varepsilon \\ B &::= C D \\ D &::= bD \mid b \end{aligned}$$

Calcoliamo la collezione LR(0) della grammatica aumentata:

I0=Clos(S'→.S)	{S'→.S, S→.aaA, S→.B, S→. , B→.CD, C→.aaCb, C→.}	<u>FOLLOW</u>	
		S	{S}
		A	{S}
		B	{S}
		C	{S,b}
		D	{S}
		conflitto reduce/reduce {S} insect {S,b} = {S}	

La grammatica non è SLR(1). Osserviamo che il fallimento è dovuto ad un conflitto falso, dovuto alla mancanza di contesto. Infatti la maniglia a C è attraversata in modi diversi dalla produzione per A e da quella per B. Possiamo modificare la grammatica per risolverlo [\(soluzione bis\)](#).

Calcoliamo la collezione LALR della grammatica aumentata:

I0=Clos(S'→.S)	{S'→.S/\$, S→.aaA/\$, S→.B/\$, S→./\$, B→.CD/\$, C→.aaCb/b, C→./b}	nessun conflitto
I1=G(0,S)	{S'→S./\$}	
I2=G(0,a)	{S→a.aA/\$, C→a.aCb/b}	
I3=G(0,B)	{S→B./\$}	

I4=G(0,C)	{B->C.D/\$, D->.bD/\$, D->.b/\$}	
I5=G(2,a)	{S->aa.A/\$, C->aa.Cb/b, A->.aaA/\$, A->.C/\$, C->.aaCb/\$/b, C->./\$/b}	nessun conflitto
I6=G(4,D)	{B->CD./\$}	
I7=G(4,b)=G(7,b)	{D->b.D/\$, D->b./\$, D->.bD/\$, D->.b/\$}	nessun conflitto
I8=G(5,A)	{S->aaA./\$}	
I9=G(5,C)=G(13,C)	{C->aaC.b/b/\$, A->C./\$}	nessun conflitto
I10=G(5,a)=G(13,a)	{A->a.aA/\$, C->a.aCb/\$/b}	
I11=G(7,D)	{D->bD./\$}	
I12=G(9,b)	{C->aaCb./b}	
I13=G(10,a)	{A->aa.A/\$, C->aa.Cb/\$/b, A->.aaA/\$, A->.C/\$, C->.aaCb/\$/b, C->./\$/b}	nessun conflitto
I14=G(13,A)	{A->aaA./\$}	

Dobbiamo calcolare ora, la tabella di analisi LALR(1):

----- omessa -----

Una grammatica SLR per il linguaggio

Riprendiamo la grammatica precedente e modifichiamo le produzioni per C nel seguente modo:

$S ::= A \mid B$
 $A ::= aa A \mid aa C \mid \varepsilon$
 $C ::= aa C b \mid aab$
 $B ::= C D \mid D$
 $D ::= bD \mid b$

in questo modo, come vedremo, rimuoviamo il conflitto. Da osservare che il linguaggio calcolato e' lo stesso della precedente grammatica (ovvio, ripensate alla relazione tra produzioni ed equazioni tra linguaggi), tuttavia la struttura delle frasi e' cambiata in particolare il sottolinguaggio L(C) non contiene piu la stringa vuota e ogni maniglia a C deve attraversare almeno una stringa aab.

Calcoliamo la collezione SLR(1):

I0=Clos(S'->.S)	{S'->.S, S->.A, S->.B, A->., A->.aaA, A->.aaC, B->.CD, B->.D, C->.aaCb, C->.aab, D->.bD, D->.b}	<u>FOLLOW</u>	
		S	{}
		A	{}
		B	{}

		<table><tr><td>C</td><td>{b,\$}</td></tr><tr><td>D</td><td>{}</td></tr></table> nessun conflitto reduce/reduce nessun conflitto shift/reduce {a,b} insect {} = {}	C	{b,\$}	D	{}
C	{b,\$}					
D	{}					
I1=G(0,S)	{S'-.S}					
I2=G(0,A)	{S->A.}					
I3=G(0,B)	{S->B.}					
I4=G(0,a)=G(8,a)	{A->a.aA, A->a.aC, C->a.aCb, C->a.ab}					
I5=G(0,C)	{B->C.D, D->.bD, D->.b}					
I6=G(0,D)	{B->D.}					
I7=G(0,b)=G(5,b)=G(7,b)	{D->b.D, D->b., D->.bD, D->.b}	nessun conflitto shift/reduce {b} insect {} = {}				
I8=G(4,a)	{A->aa.A, A->aa.C, C->aa.Cb, C->aa.b, A->.aaA, A->.aaC, A->., C->.aaCb, C->.aab}	nessun conflitto shift/reduce {a,b} insect {} = {}				
I9=G(5,D)	{B->CD.}					
I10=G(7,D)	{D->bD.}					
I11=G(8,A)	{A->aaA.}					
I12=G(8,C)	{A->aaC., C->aaC.b}	nessun conflitto shift/reduce {b} insect {} = {}				
I13=G(8,b)	{C->aab.}					
I14=G(12,b)	{C->aaCb.}					

Da notare che la collezione SLR(1) della nuova grammatica e' 'pesante' quanto la LALR della prima grammatica data.