soluzione del I compitino del 20/5/2017

Linguaggi liberi da contesto

1) Data la seguente grammatica libera da contesto G:
S -> aS | aSbS | ε
dimostrare che il linguaggio L(G) contiene solo stringhe tali che ogni loro prefisso abbia un numero di a almeno pari al numero dei b.
Soluzione: per prima cosa va notato che viene chiesto solo di dimostrare che ogni w in L(G) soddisfa la proprietà che
ogni prefisso di w abbia un numero di a almeno pari al numero dei b (e non che ogni stringa che soddisfi questa proprietà sia in L(G)).
Si procede per induzione sulla lunghezza della derivazione S=>*w.
Base: lunghezza 1. La sola derivazione di lunghezza 1 di G è S => ε.
Ovviamente ε obbedisce la condizione richiesta visto che non ha né a né b.

induzione: assumiamo che tutte le stringhe di L(G) derivabili da S con derivazioni di n passi, $n>=1$, soddisfino la proprietà e dimostriamo che allora anche le stringhe derivabili in $n+1$ passi la soddisfano.

Prendiamo $S =>^{(n+1)} w$, allora, il primo passo sarà

a)$S => aS =>^{n} w$ oppure

b)$S=>aSbS =>^{n} w$

nel caso (a), w =aw’ con w’ in L(G) e derivabile in n passi per cui, per potesi induttiva, ogni prefisso di w’ ha almeno tanti a quanti sono i b. Da questo è ovvio che, a w’=w gode della stessa proprietà.

Nel caso (b), w = aw’bw’’ e w’ e w’’ sono derivate da S in non più di n passi, per cui entrambe godono della proprietà sui loro prefissi. Ragioniamo ora sui prefissi di w=aw’bw’’. Certamente i prefissi che terminano in w’ godranno della proprietà per lo stesso motivo usato nel caso (a). Il prefisso aw’b funziona perché aw’ ha almeno un a in più rispetto ai b e quindi aw’b avrà al massimo tanti a quanti b. Per finire consideriamo i prefissi che terminano in w’’ . La proprietà vale anche per loro in quanto essi saranno costituiti da aw’b (che soddisfa la proprietà) seguito da un prefisso di w’’ (che soddisfa anch’esso la proprietà).

2)Costruire un automa a pila che P che accetta lo stesso linguaggio L(G) generato dalla grammatica G del punto precedente.

Soluzione. Usiamo la costruzione generale per trovare un automa a pila che riconosce il linguaggio generato da una CFG.

L'automa ha un solo stato q. I suoi terminali sono a e b e i simboli di stack sono S, a e b. S ha il ruolo di Z0. Accetta per pila vuota.

$\delta(q,\varepsilon,S) =\{(q,aS),(q,aSbS), (q,\varepsilon) \}$
$\delta(q,a,a)=\{(q, \varepsilon)\}$
$\delta(q,b,b)=\{(q, \varepsilon)\}$

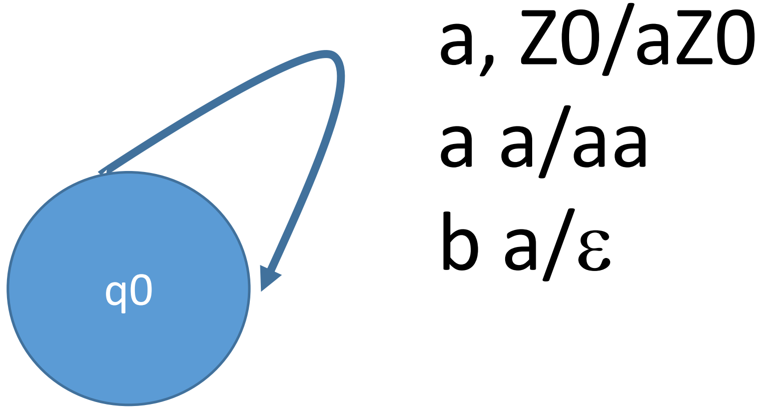
3) In generale gli automi a pila possono accettare per pila vuota o per stati finali. I linguaggi riconosciuti sono gli stessi.
Per gli automi a pila deterministici questo non è più vero. Spiegate le ragioni di questa differenza. Cercate di specificare quali linguaggi vengono accettati nelle due modalità di accettazione. La differenza tra le 2 classi di linguaggi vi pare importante?

Soluzioni. Ii DPDA che accettano per pila vuota possono accettare solamente linguaggi che hanno la proprietà del prefisso. Cioè tali che non esistano 2 stringhe x e y nel linguaggio t. c. x=yw'.
Questa limitazione è molto seria perché ci sono linguaggi regolari che non hanno la proprietà del prefisso e quindi questi DPDA non riconoscono neppure tutti i linguaggi regolari

Quindi i DPDA che accettano per pila vuota accettano esattamente tutti i linguaggi che hanno la proprietà del prefisso e che sono accettati da DPDA che accettano per stato finale.
In più possiamo dire che entrambi i tipi di DPDA accettano linguaggi non ambigui.
La differenza tra i 2 modelli di DPDA è rilevante, ma facilmente colmabile. Infatti è facile trasformare qualsiasi linguaggio in uno che soddisfa la proprietà del prefisso: basta aggiungere un simbolo $ alla fine di ogni stringa, con l’ipotesi che questo sia l’unico $.

1) Costruire un automa a pila $P=(Q,\Sigma, \Pi,\delta,q0,Z0,F)$ che accetta per stato finale (F) il linguaggio composto dalle stringhe in {a,b}* tali che, in ogni prefisso, il numero di a sia almeno pari al numero di b.

soluzione:



osserva che F=q0

Se in un prefisso ci sono più b di a, l’automa si troverà in (q0,b,Z0) per cui non c’è mossa, quindi il b non viene consumato. Insomma l’automa si blocca con input non vuoto e quindi rifiuta. Altrimenti l’automa consuma tutto l’input

2) Considerare un automa a pila P che accetta a stack vuoto e che ha 2 stati, q e p e ha la seguente mossa:

δ (q,a,X)={(p, XYX)},

con a simbolo terminale. Ricordando la costruzione che, partendo da un automa a pila, definisce una grammatica libera da contesto che genera il linguaggio che l'automa riconosce, specificare le produzioni che corrispondono alla mossa di P data prima.

soluzione. Si tratta di una costruzione vista in classe.

La grammatica ha nonterminali della forma [q X p] e corrispondentemente alla mossa indicata, avrà le seguenti 8 produzioni:

[q X q] -> a[p X q][q Y q][q X q] | a[p X q][q Y p][p X q] |
[p X p][p Y q][q X q] | [p X p][p Y p][p X q]

[q X p]->[p X p][p Y q][q X p] | [p X p][p Y p][p X p]
[p X q][q Y q][q X p] | [p X q][q Y p][p X p]

3) Dimostrare che la grammatica libera da contesto G:
$S \rightarrow aS \mid aSbS \mid \varepsilon$ è ambigua, esibendo due derivazioni leftmost (sinistre) diverse che generano una stessa stringa terminale.

Soluzione. Basta prendere la stringa aab che ha 2 derivazioni lm diverse:

$S \Rightarrow aS \Rightarrow aaSbS \Rightarrow aabS \Rightarrow aab$

$S \Rightarrow aSbS \Rightarrow aaSbS \Rightarrow aabS \Rightarrow aab$