17 giugno 2015

1. Dati i due linguaggi A = $\{0, 1\}^*\{0\}$ e B = $\{0, 1\} \cdot^* \{1\}$, rispondere alle seguenti domande
   giustificando la risposta:
   a. Che linguaggio è A ∩ B ?
   Soluzione: Considerando che A è l'insieme delle parole binarie che terminano con 0 e B
   l'insieme delle parole binarie che terminano con 1, A ∩ B = ∅. Infatti, non esiste nessuna
   parola binaria che abbia come suffisso contemporaneamente il simbolo 0 e 1.

   b. A ∪ B = $\{0, 1\}^*$?

   Soluzione: Considerando che $\{0, 1\}^*$ è l'insieme di tutte le parole binarie di lunghezza
   finita compresa la parola vuota ε, la risposta è NO. L'unione dei linguaggi A e B produce
   l'insieme delle parole binarie che terminano o con 0 o con 1, ma ε non soddisfa nessuna
   delle due condizioni perchè non contiene alcun simbolo, quindi non è compresa nell'unione
   A ∪ B.

   c. A · B = B · A ?

   Soluzione: In genere, il prodotto di due linguaggi non è commutativo. Ma questo non
   basta per giustificare la risposta che è NO. È sufficiente osservare che un linguaggio, ad
   esempio A·B, contiene una parola che l'altro linguaggio, B ·A, non contiene. Consideriamo
   la parola 01 essa è chiaramente una parola appartenente ad A · B, ma terminando con 1
   non può chiaramente appartenere a B · A, che contiene solo parole con suffisso 0.
   d. $A^* = A \cup \{\varepsilon\}$?

Soluzione: La risposta è SI. Per dimostrare l'uguaglianza di due linguaggi bisogna far verdere l'inclusione reciproca tra di essi, dobbiamo mostrare cioè che:

$$A^* \subseteq A \cup \{\varepsilon\} \text{ e che } A^* \supseteq A \cup \{\varepsilon\}$$

Si osservi che le parole in A^* sono parole ottenute dalla concatenazione di parole con suffisso 0 e pertanto contenute in A. L'unica parola che fa eccezione a questa regola è ε, ma essa è espressamente inclusa nel linguaggio $A \cup \{\varepsilon\}$.
$A^* \supseteq A \cup \{\varepsilon\}$. Il linguaggio A e il linguaggio $A^0 = \{\varepsilon\}$ sono chiaramente inclusi in A^* per definizione. Infatti, $A^* = A^0 \cup A^1 \cup A^2 \cup \cdots$.

2. Dare la definizione di linguaggio ricorsivo.

   Soluzione: Informalmente, un linguaggio ricorsivo è un linguaggio che ammette un sistema riconoscitivo, cioè un sistema che è in grado di stabilire l'appartenenza o meno delle parole allinguaggio stesso. Tale risposta non può essere completamente accetta perchè non è formalizzata e priva della possibilità di essere utilizzata a livello progettuale nella costruzione di programmi che mostrano proprietà sui linguaggi ricorsivi.
   Pertanto, la risposta che deve essere data è la definizione formale di linguaggio ricorsivo.
   Definizione:
   Un linguaggio $L \subseteq \Sigma^*$ si dice ricorsivo quando esiste un algoritmo implementato da un programma w tale che, per ogni $x \in \Sigma^*$, si ha:

$$F_w(x) = \begin{cases} 1 & x \in L \\ 0 & x \notin L \end{cases}$$

   dove con F_w si indica la semantica del programma w.

3. Sia la grammatica

   G = (T = {(,)}, V = {S}, S, P = {S → (S), S → ( )S, S → ( )}).

   a. Rispondere alle seguenti domande giustificando la risposta:

- La parola “( )( )((( )))” appartiene a L(G) ?

  Soluzione: Si, infatti una parola appartiene al linguaggio generato da una grammat-
  ica, in questo caso G, quando è possibile individuare una derivazione per essa in G.
  Per giustificare la risposta dobbiamo quindi dare l’intera derivazione della parola:
  S ⇒ ( )S ⇒ ( )( )S ⇒ ( )( )(S) ⇒ ( )( )((S)) ⇒ ( )( )((( ))).

- La parola “(S)” appartiene a L(G) ?

  Soluzione: No, una parola per appartiene al linguaggio generato da una grammat-
  ica, in questo caso G, non deve solo ammettere una derivazione per essa in G, ma
  deve anche essere composta da soli simboli terminali che in questo caso sono definiti
  dall’insieme T = {(,)}. La parola (S) non appartiene a T^*; infatti, ammette come
  fattore il simbolo S ∈ V . Nessuna parola contenente una variabile o metasimbolo di
  G appartiene a L(G).

b. Completare la frase:

La grammatica G non è di tipo . . . .

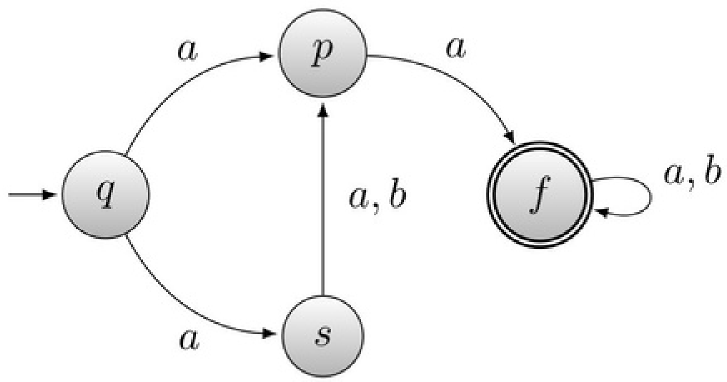
Soluzione: Tre.

4. Dare una grammatica di tipo 2 per il linguaggio $\{a^n c^m b^n \mid n, m \geq 0\}$.

   Soluzione: Diamo una possibile grammatica di tipo 2 per il linguaggio $\{a^n c^m b^n \mid n, m \geq 0\}$,
   ricordando la grammatica per il famoso linguaggio $\{a^n b^n \mid n \geq 0\}$ ed usandola per ottenere
   quella del linguaggio in questione. Definiamo:
   G = (T = {a, b, c}, V = {S, C}, S, P = {S → aSb, S → C, C → cC, C → ε}).
   La grammatica G permette di generare parole della forma $a^n S b^n$, per n ≥ 0, semplicemente
   iterando la regola S → aSb. Per eliminare la variabile S dobbiamo

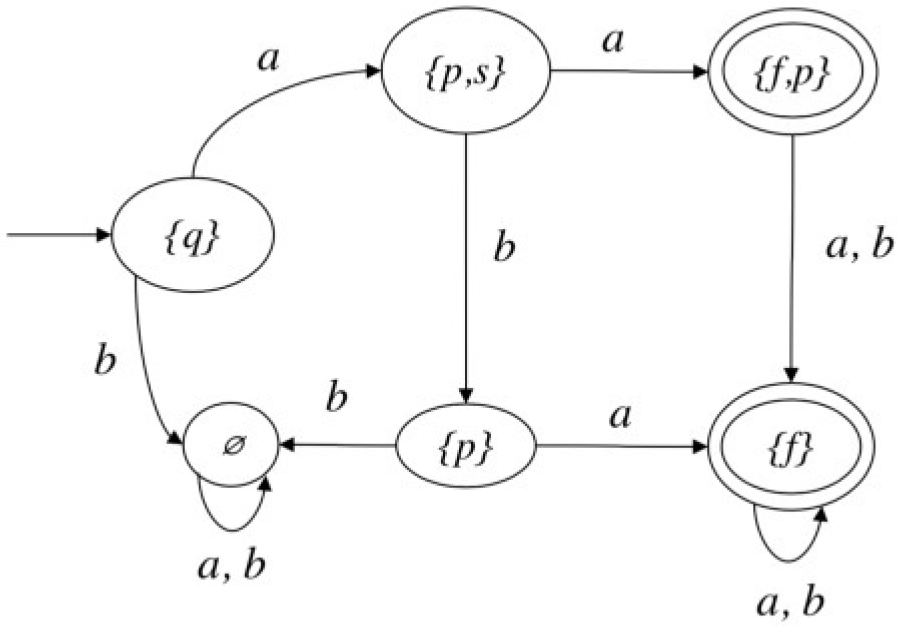
necessariamente passare dalla variabile C con la regola S → C, ottenendo $aC b$. A questo punto è possibile inserire un numero arbitrario m, di simboli c con le regole C → cC, C → ε, ottenendo una parola della forma $a^n c^m b^n$. Si noti che dovendo una derivazione partire da S, l'assioma di G, non è possibile ottenere altre parole che non siano della forma richiesta. Dobbiamo ora giustificare che la grammatica G che abbiamo appena dato sia di tipo 2. Una grammatica è di tipo 2 se le sue regole di produzione sono della forma A → β con A ∈ V e β ∈ $(V ∪ T)^*$. Ogni regola in P soddisfa questa condizione e pertanto G è di tipo 2.

5. Sia A il seguente automa non deterministico seguente:



Disegnare un automa deterministico equivalente ad A.

Soluzione: Usando la subset construction mostrata nel teorema di equivalenza tra gli automi
a stati finiti deterministici e non deterministici si ottiene l'automa deterministico seguente:



02 luglio 2015

1. Dati i due linguaggi $A = a^*cb^*$ e $B = c^*$, rispondere alle seguenti domande:
   a. Che linguaggio è $A \cap B$?
   Soluzione: $A \cap B = \{c\}$. Infatti, essendo B un linguaggio unario sull'alfabeto {c}, l'unico
   elemento in comune con A è proprio la parola fatta da una sola c. La parola c appartiene
   ad A dato che ε è presente sia a sinistra che a destra dell'espressione regolare a^*cb^* che
   denota A.

   b. Sia A^c il complemento di A. Vale $B \subseteq A^c$? Giustificare la risposta.
   Soluzione: No. Si noti che in B è presente la parola c, che appartiene anche ad A (vedi
   il punto sopra), di conseguenza si ha $c \notin A^c$. Pertanto esiste una parola che appartiene a B ma non ad A^c e questo prova che B non è un sottoinsieme di A^c.
   c. Che linguaggio è $(B \cdot A \cdot B) \cap B$?

   Soluzione: Le parole del linguaggio intersezione (il linguaggio $(B \cdot A \cdot B) \cap B$) devono
   essere composte solamente dal simbolo c visto che $B = c^*$. Il prodotto $B \cdot A \cdot B$ ha in

comune con B proprio queste parole perchè in a^* e b^* è denotata anche la parola vuota.
L'unica eccezione è ε che sebbene sia presente in B non è ottenibile nel linguaggio prodotto
(B · A · B) a causa di A che non la contiene. Pertanto vale (B · A · B) ∩ B = c+ .

d. Vale $B^* = B^+$? Giustificare la risposta.
Soluzione: Si, perchè l'unico elemento che può fare la differenza tra i due linguaggi è
$B^0 = ε$. Ma in questo caso il linguaggio B contiene la parola vuota ($B = c^*$), per cui
$B^+ = B$. Più precisamente vale $B = B^* = B^+ = c^*$.

2. Dare la definizione formale di linguaggio ricorsivamente numerabile.
   Soluzione: Informalmente, un linguaggio ricorsivamente numerabile è un linguaggio che am-mette un sistema generativo, un sistema cioè che consente di elencare le parole del linguaggio. Tale sistema però non permette un completo riconoscimento del linguaggio stesso. Questa risposta non può essere completamente accetta perchè non è formalizzata e priva della possibilità di essere utilizzata a livello progettuale nella costruzione di programmi che mostrano proprietà sui linguaggi ricorsivamente enumerabili. Pertanto, la risposta che deve essere data è la definizione formale di linguaggio ricorsivamente numerabile.
   Definizione: Un linguaggio $L \subseteq \Sigma^*$ si dice ricorsivamente numerabile quando esiste una procedura imple-mentata da un programma w tale che, per ogni $x \in \Sigma^*$ si ha:

$$F_w(x) = \begin{cases} 1 & x \in L \\ \uparrow & x \notin L \end{cases}$$

   dove con F_w si indica la semantica del programma w e con ↑ un programma che non termina.

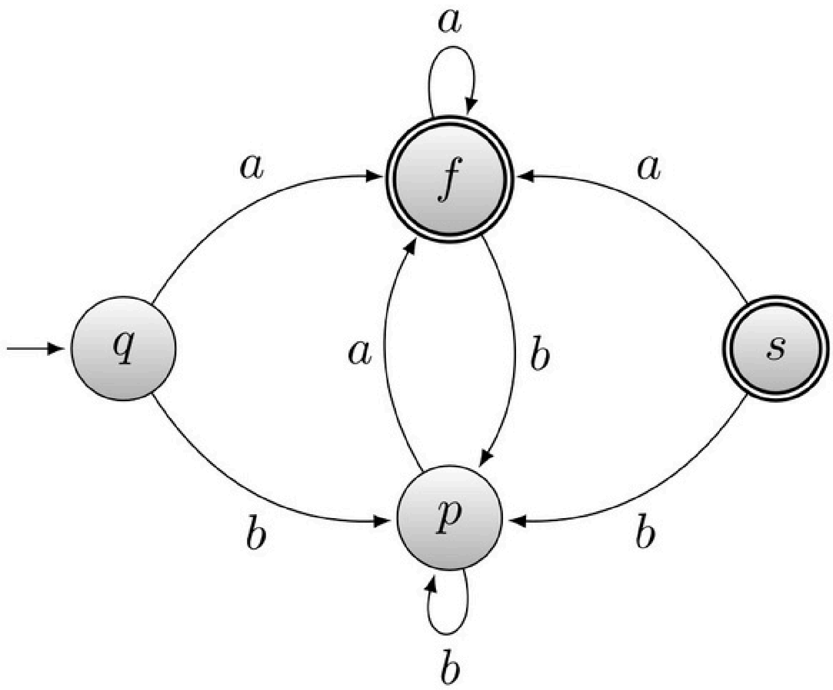
3. Sia la grammatica
   G = (T = {a, b, c}, V = {S}, S, P = {S → aSa, S → bSb, S → c}).

a. Mettere una crocetta di fianco alle parole appartenenti a L(G): ε c aacaa baab bacba bcb aSa aca baba abSab abcba abba

b. Mettere una crocetta sulle risposte corrette:

La grammatica G è di tipo 1? SI NO **soluzione**: SI La grammatica G è di tipo 2? SI NO **soluzione**: SI La grammatica G è di tipo 3? SI NO **soluzione**: NO

4. Sia A il seguente automa:



a. Mettere una crocetta sulle risposte corrette:

- aba ∈ L(A) ? SI NO **soluzione**: SI
- aba*⊆ L(A) ? SI NO **soluzione**: NO
- a*∗ba ⊆ L(A) ? SI NO **soluzione**: SI
- a^+ ⊆ L(A) ? SI NO **soluzione**: SI

b. In A sono presenti stati indistinguibili tra loro? Se sì quali?

Soluzione: Si, gli stati indistinguibili sono (p, q) ed (f , s). Per definizione p ≈ q se e solo
se per ogni w ∈ {a, b}*vale λ(δ(q*, w)) = λ(δ(p*, w)). Dimostrare che ciò è vero in questo
caso è facile. Infatti valgono le seguenti uguaglianze:

δ(q, a) = δ(p, a) e δ(q, b) = δ(p, b).

Qualunque sia la parola w che leggiamo a partire da p o da q, dopo il primo simbolo di w,
si raggiunge sempre lo stesso stato e pertanto le risposte dell'automa A non potranno che
essere uguali. Similarmente accade per f ed s.

5. Sia A l'automa dell'esercizio 4.
   a. Fornire una grammatica G equivalente ad A.
   Soluzione: Si utilizza la costruzione presente nella dimostrazione del teorema che afferma
   l'equivalenza tra le grammatiche di tipo 3 e gli automi a stati finiti.
   Senza cambiare notazione rispetto a quella che definisce l'automa A, definiamo la gram-
   matica:

   $$G = (T = \{a, b\}, V = \{q, p, f, s\}, q \text{ (assioma)}, P)$$

   dove le seguenti regole di produzione appartengono a P : • q → af, q → bp, • p → bp, p → af, • f → af, f → bp, f → ε, • s → af, s → bp, s → ε. **b.** Esistono regole di G che possono essere eliminate senza modificare L(G)? Se sì quali?

   Soluzione: Si sono le regole s → af, s → bp, s → ε. Possono essere eliminate in quanto
   la variabile s non è raggiungibile dall'assioma della grammatica e quindi le regole per s
   non aiutano a generare parole del linguaggio. Questo lo si deduce anche dal diagramma
   degli stati di A dove s risulta essere uno stato non osservabile (ovvero non raggiungibile
   da alcuna parola a partire dallo stato iniziale di A).

17 luglio 2015

1. Dati i due linguaggi binari $A = \{1\}^* \cdot \{0, 1\}^*$ e $B = \{0\}^* \cdot \{0, 1\}^*$, rispondere alle seguenti
   domande:

a. Sia A^c il complemento di A. Che linguaggio è A^c?

Soluzione: $A^c = \emptyset$. Si noti che $A = \{0, 1\}^*$, infatti 1^* comprende anche la parola vuota.

b. Vale $A \cap B = A \cup B$? Giustificare la risposta.

Soluzione: Come detto al punto sopra $A = \{0, 1\}^*$ e pertanto per simmetria $B = \{0, 1\}^*$,
di conseguenza sia il linguaggio unione che il linguaggio intersezione è $\{0, 1\}^*$.
Perciò, SI, vale l'uguaglianza.

c. Vale $A \cdot B = A$? Giustificare la risposta.

Soluzione: Dato che $A = B = \{0, 1\}^*$, anche $A \cdot B = \{0, 1\}^*$, e dunque l'uguaglianza
$A \cdot B = A$ vale.

d. Mettere una crocetta di fianco ai linguaggi che contengono la stringa nulla:

A A^c $A \cap A^c$ $A \cup A^c$ A^* A^+ A^0 A^2 $A \cap B$ $A \cdot B$ $(A \cdot B)^+$ $(A \cdot B)^*$

2. Dare un esempio di linguaggio ricorsivamente numerabile ma non ricorsivo. Dimostrare che il linguaggio dato è ricorsivamente numerabile.

   Soluzione: Sia u il programma interprete, ovvero il programma che soddisfa $F_u(w\$x) = F_w(x)$ (cioè, u su input w\$x mi restituisce il risultato del programma w su input x). Si definisce il linguaggio dell'arresto ristretto l'insieme:

$$D = x \in \{0,1\}^* | F_u(x\$x) \downarrow$$

dove $\downarrow$ indica che il programma termina. Il linguaggio D è un esempio di linguaggio ricorsivamente numerabile ma non ricorsivo. Per quanto riguarda la prima affermazione, si consideri la seguente procedura:

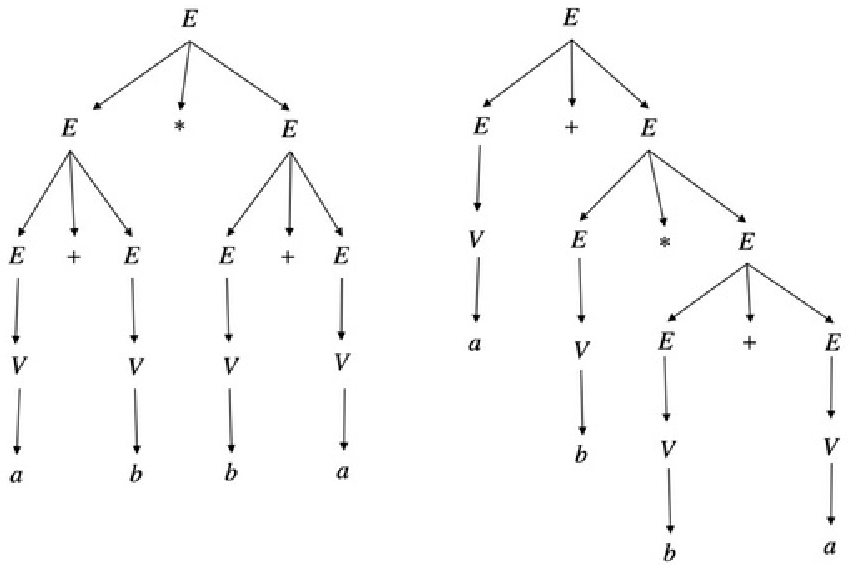
RICNUM(x $\in \{0, 1\}^*$)
y = F_u (x\$x)
return 1

Si noti che se x $\in$ D allora $F_{\mathcal{U}}$(x\$x) $\downarrow$ e RICNUM(x)= 1, mentre se x $\notin$ D allora $F_{\mathcal{U}}$(x\$x) $\uparrow$ e
RICNUM(x) $\uparrow$. Per dimostrare che non è ricorsivo:

Procedura ASS(x∈{0,1}*)
  if x∈A then return(1- $F_{\mathcal{U}}$(x\$x))
  else return (0)

3. Sia la grammatica
   G = (T = {a, b, +, ⋆}, V = {E, V }, E, P = {E → E+E, E → E⋆E, E → V, V → a, V → b}).
   a. Disegnare l'albero di derivazione per a + b ⋆ b + a. È unico?

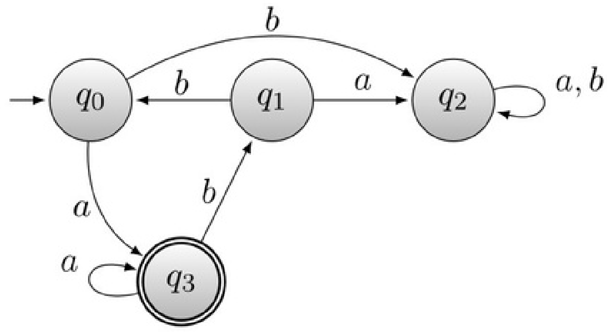
   Soluzione: No, per tale parola esistono almeno due alberi di derivazione in G differenti.
   Ad esempio i seguenti sono alberi di derivazione per a + b ⋆ b + a:



   b. Stabilire se la grammatica G è ambigua. Giustificare la risposta.

   Soluzione: La grammatica G è ambigua in quanto esiste una parola in L(G) che ammette
   due alberi di derivazione differenti. Ad esempio, la parola a + b ⋆ b + a citata sopra.

4. Sia A il seguente automa:



Ricavare un’espressione regolare per L(A):

Soluzione: Usando la tecnica data nella dimostrazione del Teorema di Kleene, che afferma
l’equivalenza tra le espressioni regolari e gli automi a stati finiti, si può impostare il seguente
sistema di equazioni di linguaggi dove L(A) è l’incognita X_0:

$$\begin{cases} X_0 = aX_3 + bX_2 \\ X_1 = aX_2 + bX_0 \\ X_2 = aX_2 + bX_2 \\ X_3 = aX_3 + bX_1 + \varepsilon \end{cases}$$

Ognuna di queste equazioni deve essere risolta sfruttando l’equazione tipo X = AX + B, la
cui soluzione è $X = A^*B$. Partiamo dall’equazione $X_2 = aX_2 + bX_2$, allora la sua soluzione è
$X_2 = \emptyset$. Inserendo tale valore nell’equazione due si ottiene $X_1 = a\emptyset + bX_2$ Ora sostituisco il
valore trovato di X_1 nell’equazione quattro, si ha $X_3 = aX_3 + bbX_0 + \varepsilon$,
da cui $X_3 = a^*(bbX_0 + \varepsilon)$.
Ora possiamo sostituire il valore di X_3 nell’equazione uno:

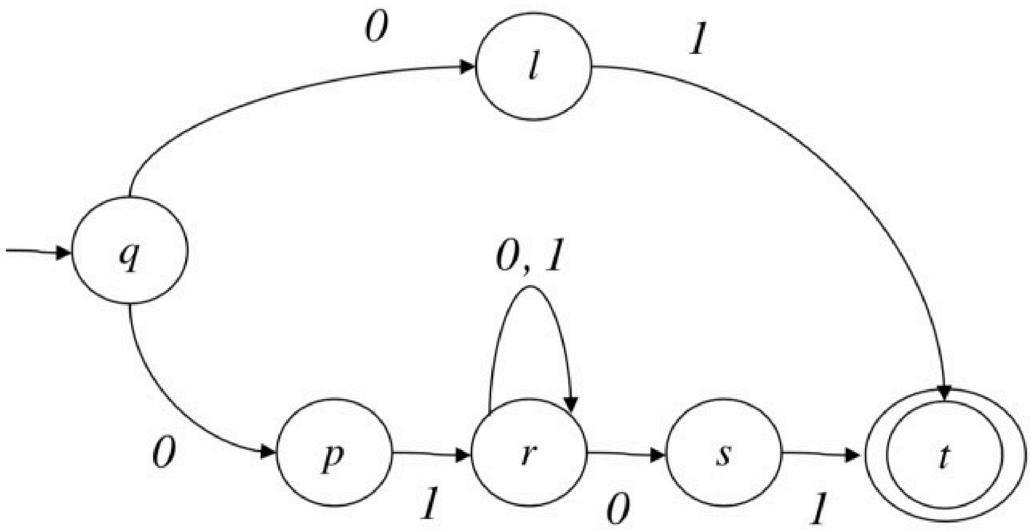
$$X_0 = a(b^+bX_0 + \varepsilon) + b\emptyset = abbX_0^+ + a^+$$

.
Applicando ancora una volta l’equazione tipo si ottiene:

$$X_0 = (a^+bb)^*a^+ = L(A).$$

5. Disegnare un automa a stati finiti che riconosce le parole su {0, 1} con prefisso e suffisso 01.
Non è richiesto che l’automa sia deterministico.

Soluzione: La scelta di un automa nondeterministico semplifica l'esercizio. Prima di dare il di-
agramma degli stati dell'automa richiesto, si noti che oltre alle parole denotate dall'espressione
regolare $01\{0, 1\}^*01$, l'automa deve riconoscere anche la semplice parola 01. Infatti, per
definizione di prefisso e suffisso anche 01 ha prefisso e suffisso 01. L'automa richiesto è il
seguente:



14 settembre 2015

1. Sia L un linguaggio ricorsivo. L è ricorsivamente numerabile? Se sì, dare una procedura che dimostri l'asserto. Se no, dare un controesempio.

2. Siano i seguenti linguaggi binari denotati da espressioni regolari estese:

$$A = 10^* \quad B = 10^* \quad C = 1^+ \quad D = 0^+ .$$

   a. Che linguaggi sono $A \cap B$ e $C \cap D$?

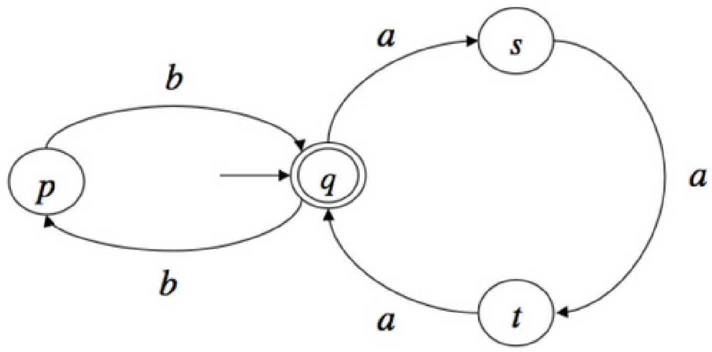
   b. Che linguaggio è $(A \cup B) \cap (C \cup D)$?

   c. Vale $C^* = C$? Giustificare la risposta.

   d. Dare un'espressione regolare per il linguaggio $(A \cdot B) \cap (C \cdot D)$?

3. Dare una grammatica di tipo 2 per il linguaggio
   $\{a^n\{0, 1\}^* b^n \mid n \geq 0\}$.

4. Dare la definizione formale di riconoscitore a pila (prestare particolare attenzione a δ, la funzione di evoluzione della pila).
5. Sia A il seguente automa a stati finiti non completamente specificato:



a. Ricavare un’espressione regolare per L(A).
b. Disegnare un automa equivalente ad A che sia completamente specificato.

13 Febbraio 2017

1. Dati i Linguaggi A={a, b}*b e B=(ab)*b
a) Quante volte la parola $(ab)^k b$ con k>0 contiene il fattore “ba”?
b) Dare la definizione formale di due linguaggi generici
c) Vale $A \cdot B \subseteq A$? Giustificare la risposta
2. Sia R_k la classe dei linguaggi generati da grammatiche di tipo K.
a) Enunciare il teorema di inclusione degli R_k
b) Commentare il teorema giustificando perché valgono le inclusioni
3. Data l’equazione $X = A \cdot X + B$, con A, B due linguaggi e X il linguaggio incognita
a) Dare la soluzione dell’equazione
b) Provare formalmente che la soluzione soddisfa l’equazione
4. Data la grammatica di tipo 3: G=(T={a,b}, V={S}, S, P={S→ abS, S→ baS, S→ ε})
a) Costruire un automa A a stati finiti equivalente a G, evidenziando il metodo
utilizzato.?
b) Dare un’espressione regolare per L(A), evidenziando il metodo utilizzato

5. Sia A=(Σ={a,b}, K={S,O}, S, δ) un riconoscitore a pila dove δ è così definita:

δ	a	b
S	{OS, SO, O}	-
O	-	ε

a) Dare il grafo di computazione di A per la parola "abaab"
b) Tale parola è accettata da A?

14 Giugno 2017

1. Elencare e definire formalmente le operazioni insiemistiche e tipiche dei linguaggi formali
2. a) Elencare il Teorema di Kleene
   b) Siano G1=(T_1, V_1, S_1, P_1) e G2=(T_2, V_2, S_2, P_2) due grammatiche di tipo 3 con regole
   del tipo A→ σB e → ε dove A,B sono variabili e σ simbolo terminale. Costruire una
   grammatica G3 di tipo 3 per il linguaggio L(G1) ⠂L(G2).
3. a) Dare le definizioni formali di automa a stati finiti deterministico e non deterministico
   b) È noto che per ogni NFA A=(Σ, Q, q, R, F) esiste un DFA A'=(Σ, Q', q', δ, F')
   equivalente. Indicare ogni componente di A' in funzione di quelle di A.
4. Siano A e B linguaggi ricorsivi. Il linguaggio A ∪ B è ricorsivo? Se sì, dare un programma
   che dimostri l'asserto. Se no, un controesempio.
5. Sia A=(Σ={a,b}, K={S,O}, S, δ) un riconoscitore a pila. δ definita così:

δ	a	b
S	{OS, SO, O}	-
O	-	ε

A è non deterministico? ☐Sì ☐No
Ricavare una grammatica di tipo 2 equivalente ad A.

3 Luglio 2017

1. Sia A un linguaggio ricorsivo e B un linguaggio ricorsivamente numerabile. Dare unprogramma che dimostri che $A \cap B$ è ricorsivamente numerabile

2. Dati i seguenti linguaggi binari:

   $A = (ab)^{+}a$ $B = (ab)^{+}b$

   a) Che linguaggio è $A \cap B$?

   b) L'espressione regolare $(ab)^{+} \cdot (a + b)$ denota il linguaggio: ....

   d) Vale $A \cdot B \subseteq B$? Giustificare la risposta

   c) Sia $w \in A^k$ e $v \in B^h$. Rispondere alle seguenti domande:

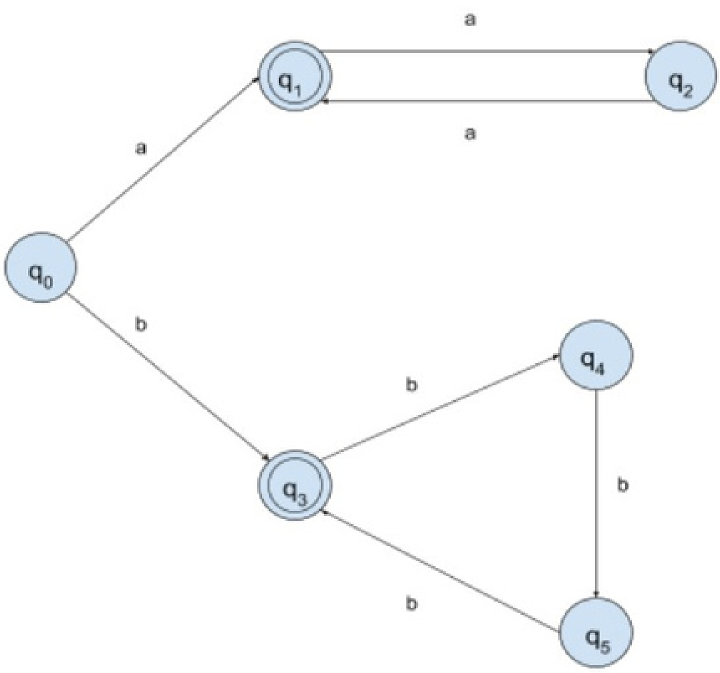
   - Quante volte il fattore "aa" è presente in w?
   - Quante volte il fattore "bb" è presente in v?

3. Sia A=(Σ, Q, q0, δ, F) un automa a stati finiti .

a) Definire formalmente la relazione "≈" di indistinguibilità tra stati

b) Definire formalmente l'automa A≈ equivalente ad A

4. Sia A il seguente automa a stati finiti deterministico non completamente specificato:



   Ricavare un'espressione regolare per L(A) evidenziando il metodo utilizzato

5. Sia G una grammatica di tipo 2.

a) Quando una parola di L(G) si dice ambigua?

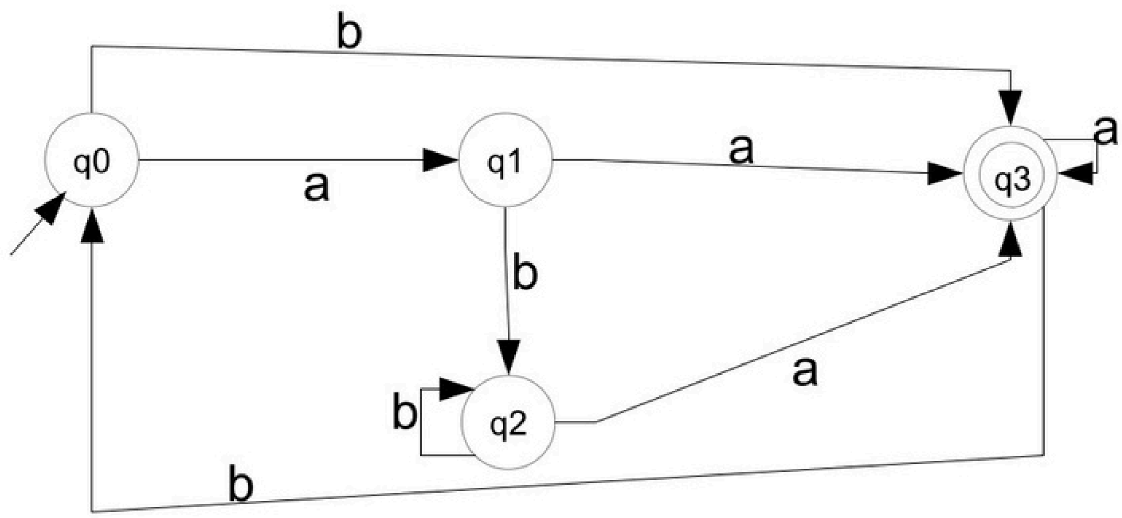
b) Quando G si dice ambigua?

c) Sia G = (T={a, ◯, ⬤}, V={E}, E, P={E➡️(E◯E, E⬤E,a)}). Mostrare che G è ambigua

Luglio 2021

1. Sia A un linguaggio ricorsivo e B un linguaggio ricorsivamente numerabile. Dare un programma
   che dimostri AUB è ricorsivamente numerabile.

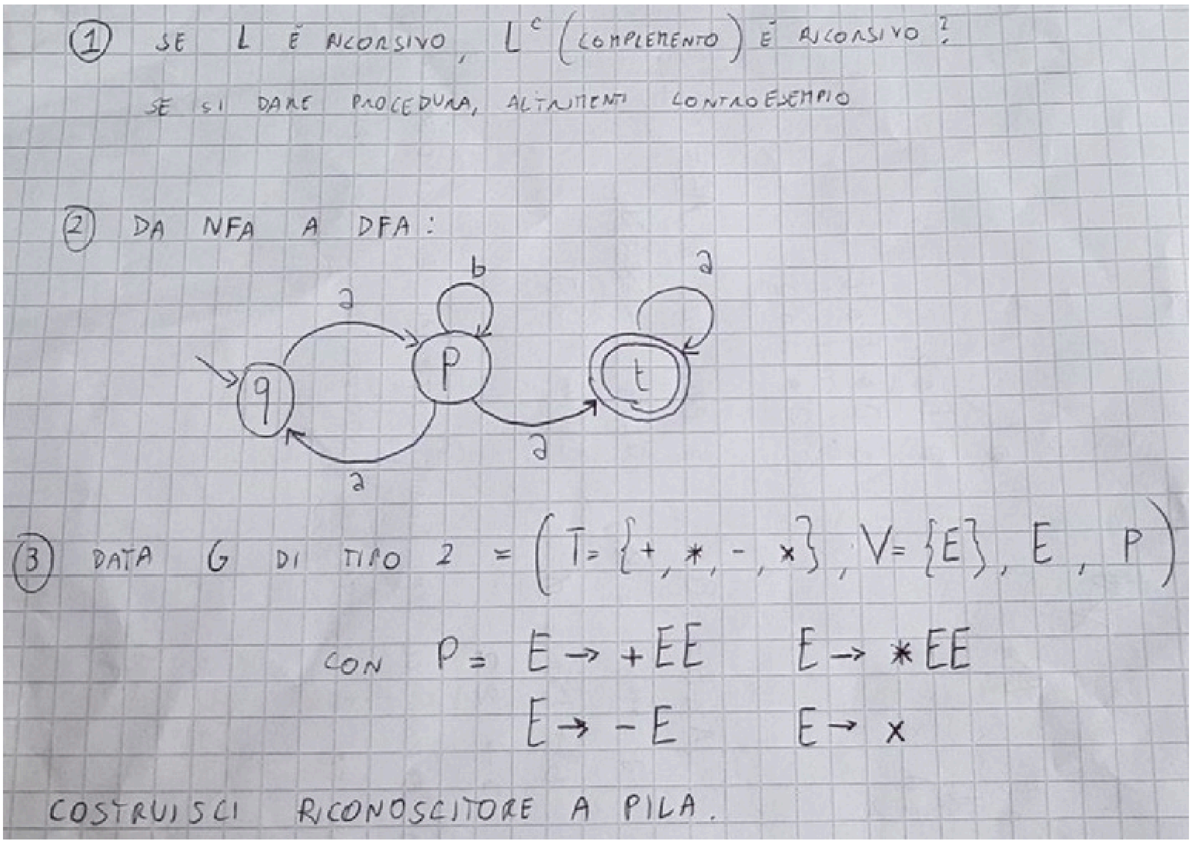
2. Sia due linguaggio binari $A=a\{a,b\}^*$ e $B=(bb)a^*$
   a. $A \cap B$ = ...
   c. Nel linguaggio $A \ast B$ ci sono parole palindrome della forma xx^r,
   dove x^r reverse di x. (scrivine tre di queste parole)
   b. siano $w \in A^k$, $v \in B^h$, $z \in (A \ast^k B)^h$, per interi positivi (>0) k,h.
   Allora:
   - numero minimo di “a” in w:
   - numero minimo di “b” in v:
   - numero minimo di “a” in z:
3. Dare la definizione formale di espressione regolare.
4. Sia A il seguente automa a stati finiti



   a. L'automa A è deterministico ? Giustifica la risposta.
   b. Ricavare una grammatica G equivalente ad A.
   c. che tipo è G? [optizioni] 0 1 2 3

5. a. Dare enuciato del Pumping Lemma per i linguaggi liberi dal contesto.
   b. Il linguaggio $\{ a^n b^n c^n \mid n>0 \}$ soddisfa il pumping lemma ? SI NO

6 settembre 2021



1) SE L È RICORSIVO, L^c (COMPLEMENTO) È RICORSIVO?

SE SI DARE PROCEDURA, ALTRIMENTI CONTROESEMPIO

2) DA NFA A DFA:

3) DATA G DI TIPO 2 = $(T=\{+, *, -, x\}, V=\{E\}, E, P)$

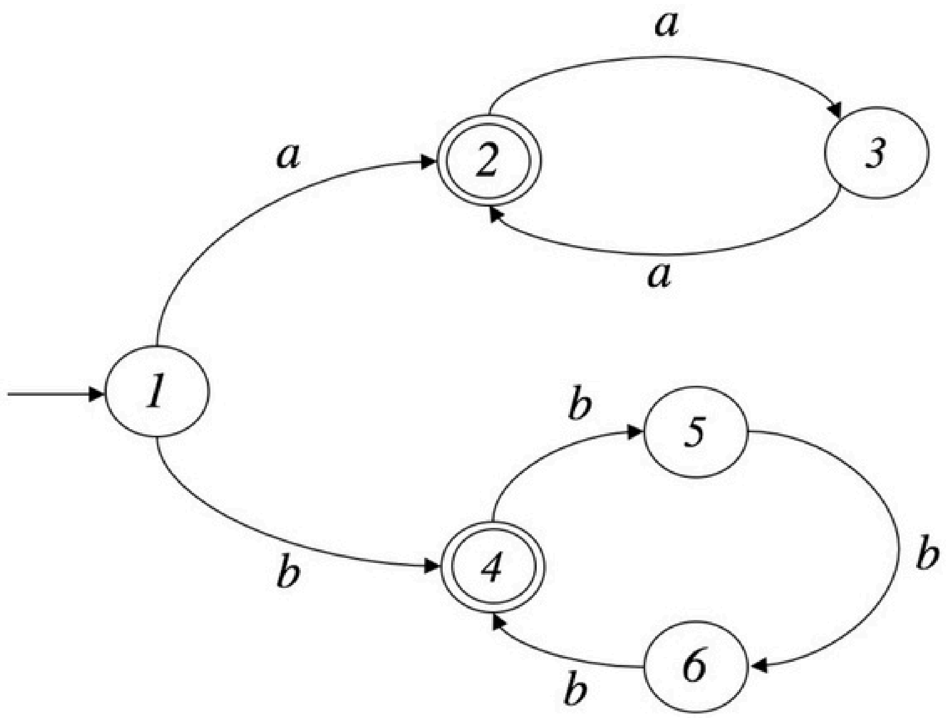
CON $P = E \rightarrow +EE \quad E \rightarrow *EE$

$E \rightarrow -E \quad E \rightarrow x$

COSTRUISCI RICONOSCITORE A PILA.

26 gennaio 2022

1. Siano A e B due linguaggi ricorsivi. Il linguaggio A ∪ B è ricorsivamente enumerabile? Se sì,
   dare un programma che dimostri l’asserto. Se no, dare un controesempio.
2. Sia A il seguente automa a stati finiti determistico non completamente specificato:



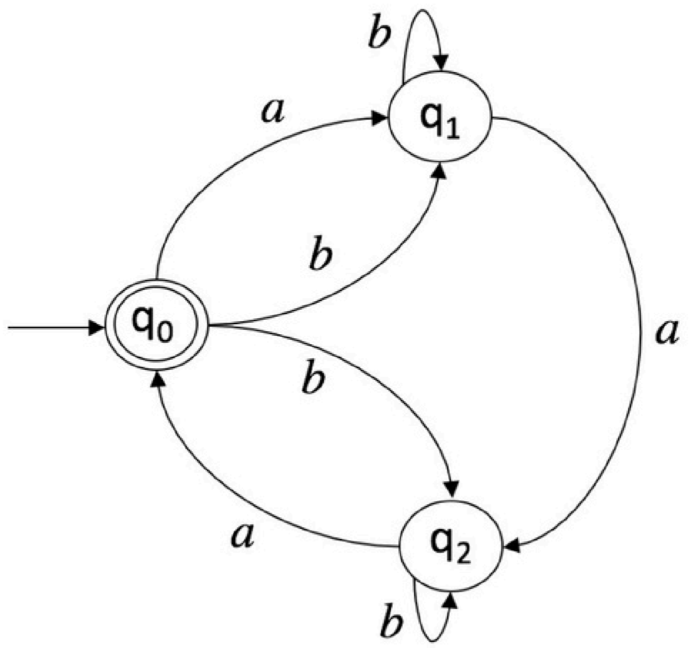
Ricavare un’espressione regolare per L(A). Verrà valutato il metodo utilizzato.

3. Sia $E \subseteq \{0, 1, (,), \wedge, \vee, \neg\}$ il linguaggio delle espressioni booleane definito dalle seguenti regole:
   • $0, 1 \in E$,
   • se $e, r \in E$ allora anche $(e \wedge r)$, $(e \vee r)$, $\neg e \in E$,
   • nient’altro appartiene ad E.
   (a) Dare una grammatica G di tipo 2 che generi E.
   (b) Ricavare un riconoscitore a pila per il linguaggio E. Verrà valutato il metodo utilizzato.
   (c) La grammatica G data al punto (a) è ambigua? Giustificare la risposta.

9 febbraio 2022

1. (a) Sia D il linguaggio dell’arresto ristretto. Definire formalmente il linguaggio D.
   (b) Dimostrare che D è ricorsivamente enumerabile implementando una procedura in pseu-
   docodice.

2. Sia A il seguente automa a stati finiti:



   (a) L’automa A è nondeterministico. Perchè?

   (b) Ricavare un automa deterministico equivalente ad A (verrà valutata la tecnica applicata).

3. Data una parola w = w1 w2 · · · wn con wi ∈ {0, 1}, si definisce $w^R = w_n w_{n-1} \cdots w_n$. Definiamo inoltre

   L = $\{ww^R \mid w \in \{0, 1\}^+\}$.

   (a) Dare una grammatica G di tipo 2 che generi L.

   (b) Dare un riconoscitore a pila per il linguaggio L.

Settembre 2023

1. Stabilire se un linguaggio ricorsivo è ricorsivamente enumerabile
2. Le due forme normali Chomsky e Greibach
3. Trovare l'automa equivalente con l'indistinguibilità degli stati

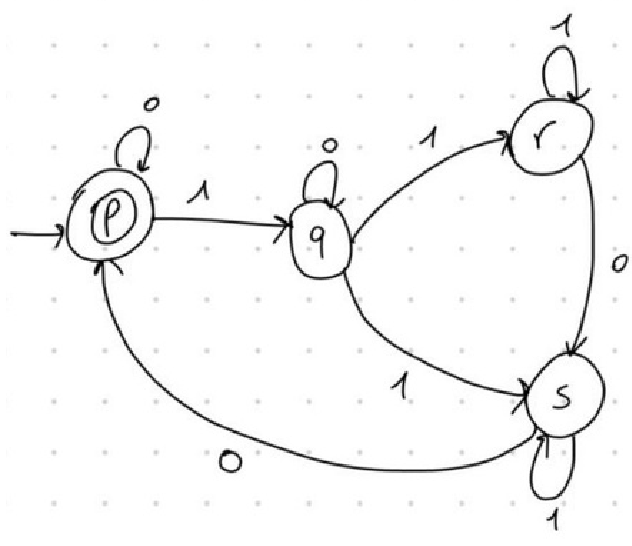
Ottobre 2023

1. operazioni sui linguaggi e operazioni tipiche sui linguaggi
2. date 2 grammatiche di tipo 3 e fare G3 = L(G1)*L(G2)
3. data la parola, fare grafo di computazione della pila

Gennaio 2024

1. Dato L ricorsivo, L^c è ricorsivo? Se si scrivere un programma che lo dimostri altrimenti controesempio.

2. Con l'automa A con input binario



espressione regolare linguaggio L(A).

3. Data la parola w= $w_1 w_2 \dots w_n$ con wi ∈{0,1} $w^r = w_n w_{n-1} \dots w_1$

   Definiamo inoltre L = {w w^r| w ∈ $\{0,1\}^+$}

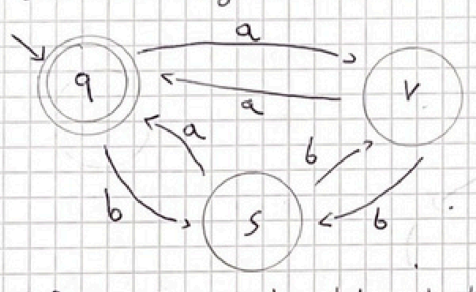
   a) Definire una grammatica G di tipo 2 che generi L

   b) Dare un riconoscitore a pila per il linguaggio L

Febbraio 2024

1. Sia D il linguaggio dell'arresto ristretto.
   (a) Dare la definizione formale di D.
   (b) Dimostrare che D è ricorsivamente enumerabile

2. Sia A il seguente automa a stati finiti:



   (a) Ci sono coppie di stati indistinguibili? Se sì, quali?
   (b) Definire formalmente l'automa $A_{\approx}$ equivalente ad A.
   (c) Disegnare l'automa $A_{\approx}$ appena definito

3. Sia la grammatica
   $G = (T = \{a, b, (,), +, \times\}, V = \{E\}, E,$
   $P = \{E \to (E+E), E \to (E \times E), E \to a, E \to b\})$

   (a) Disegnare un albero di derivazione per la parola $(((a \times b)+a) \times b)$
   (b) Dare la definizione formale di grammatica ambigua
   (c) La grammatica G è ambigua? ○ sì ○ no

21 Giugno 2024

LFA – 21/06/2024

1. Sia L ricorsivo. L^c è ricorsivo? Programma o controesempio

2. Dato l'automa deterministico a stati finiti
$A = (\Sigma, Q, q_0, \delta, F)$

   a) Definire formalmente $A_{\approx}$ in funzione dell'indistinguibilità tra stati $\approx$

   b) Enunciare formalmente il corollario su $A_{\approx}$

3. Data $w = w_1 w_2 \dots w_n$, $w^R = w_n w_{n-1} \dots w_1$

   $L = \{ww^R \mid w \in \{0,1\}^*\}$

   a) Dare G di tipo 2 per L

   b) Dare riconoscitore a pila per L

6 Settembre 2024

1) Sia L il linguaggio dell'arresto ristretto. Dare la definizione formale di L e dimostrare che L è ricorsivamente numerabile

2) Sia dato un automa a stati finiti A=(Σ, Q, q0, δ, F)
a) Dimostrare come ricavare una grammatica G di tipo 3 equivalente ad A
b) Dimostrare che L(A) = L(G) supponendo vera la proprietà seguente su A e su G

$$\delta^*(q_0, w) = p \quad \Leftrightarrow \quad q_0 \Rightarrow_G^* wp.$$

3) Sia E ⊆ {0, 1, (,), ∧, ∨, ¬}* il linguaggio delle espressioni booleane definito dalle seguenti regole:
- 0, 1 ∈ E
- Se e, r ∈ E allora anche (e ∧ r), (e ∨ r), ¬e ∈ E
- Nient'altro appartiene ad E

a) Dare la grammatica di tipo 2 in forma normale di Greibach che generi E
b) Ricavare un riconoscitore a pila per il linguaggio E

18 Settembre 2024

1) Dare la definizione formale di linguaggio ricorsivamente enumerabile.
Se L è un linguaggio ricorsivo, L è ricorsivamente enumerabile?

2) Sia A = (Σ, Q, q0, R, F) un'automa a stati finiti non deterministico. Ricavare un'automa A' deterministico equivalente ad A (Definizione formale, no esempi)

3) Sia A = (Σ = {a, b}, k = {S, O}, S, δ) un riconoscitore a pila, dove δ è così definita:

δ	a	b
S	{OS, SO, O}	-
O	-	ε

a) A è nondeterministico? SI/NO (Crocetta)
b) Ricavare una grammatica G di tipo 2 equivalente ad A