

Informatica Teorica I (Informatica Teorica primo modulo)
Esame del 31 gennaio 2003

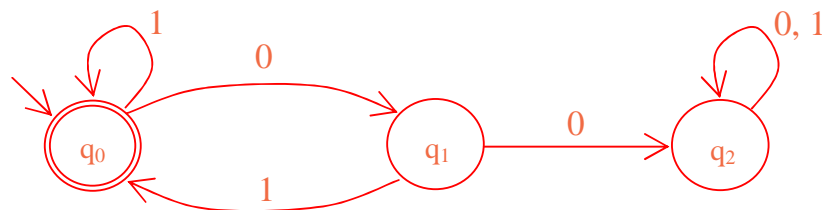
Tempo a disposizione: 90 minuti

Regole del gioco: Libri e quaderni chiusi, vietato scambiare informazioni con altri; indicare su tutti i fogli, con chiarezza, nome e numero di matricola; consegnare solo i fogli con le domande (questi).

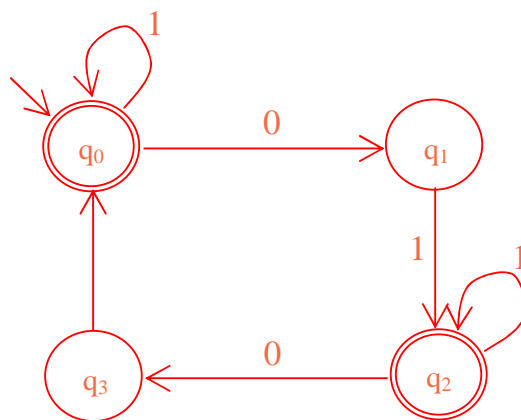
ogni esercizio viene valutato in decimi. Il voto totale in trentesimi si ottiene dalla media del voto degli esercizi moltiplicata per tre.

Esercizio 1 (20%) Costruire un ASF deterministico che riconosca il linguaggio delle stringhe di $(0+1)^*$ tali che ogni zero e' seguito da un uno. Esempi di stringhe del linguaggio: 011111, 1111111, 01010101, 0111011111.

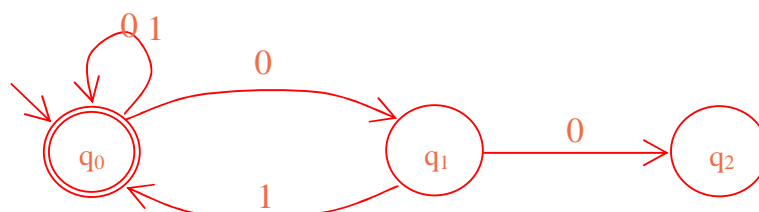
soluzione corretta (10 punti)



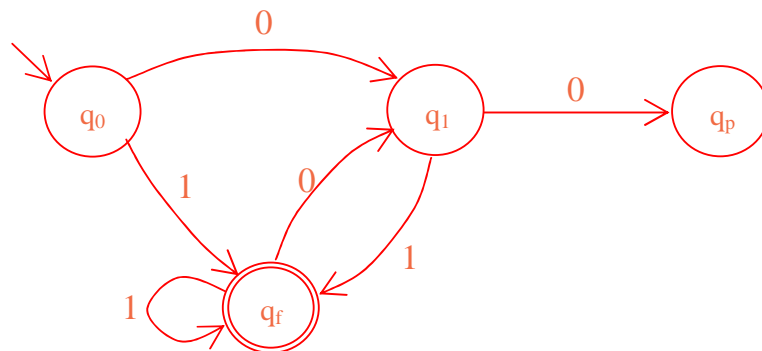
altra soluzione corretta (10 punti)



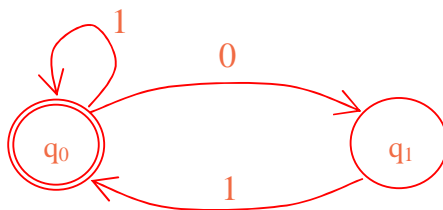
soluzione incompleta (sempre 10 punti)



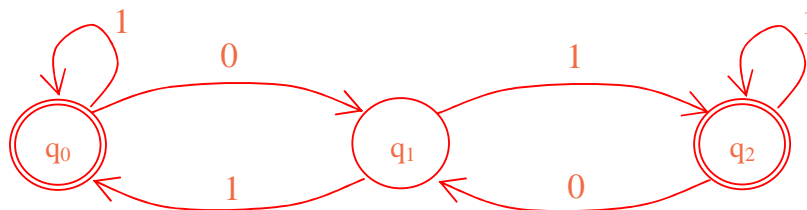
soluzione incompleta e che non riconosce stringa vuota (9,9999 punti)



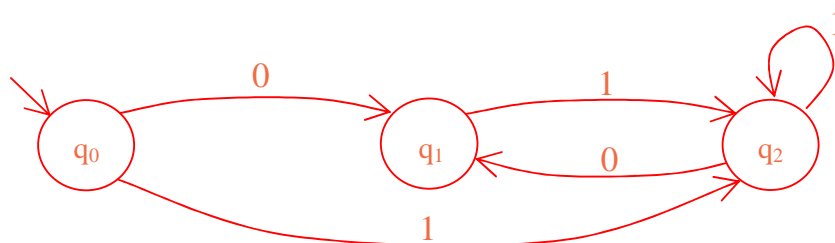
soluzione in cui lo stato iniziale non è specificato (9 punti)



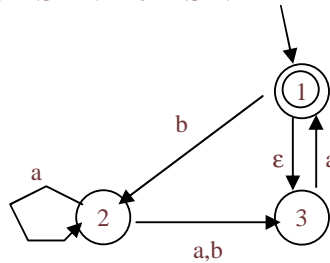
soluzione non deterministica (8 punti)



soluzione senza stati finali (8 punti)

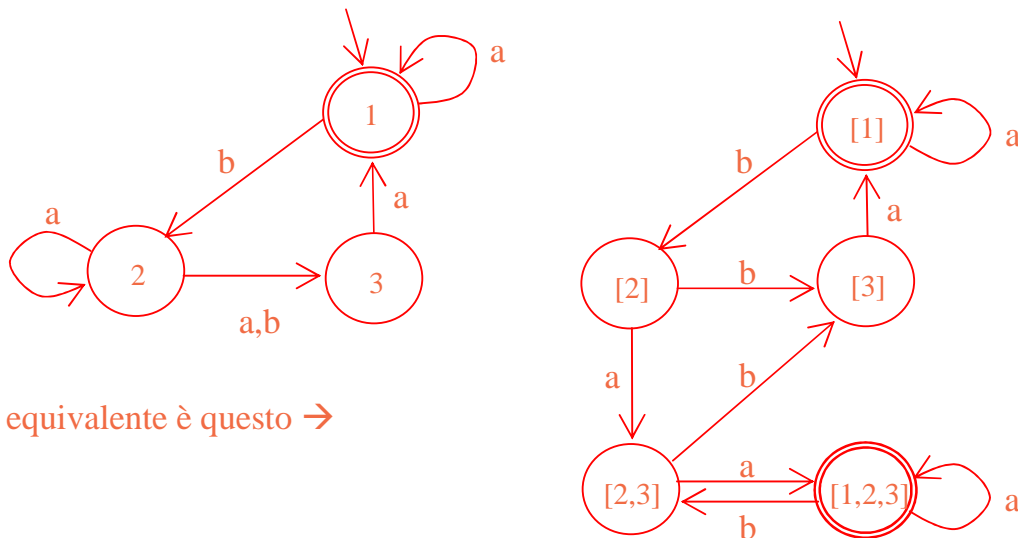


Esercizio 2 (20%) Dato il seguente ASFND disegnare l'ASF che riconosce lo stesso linguaggio. Nota: a lezione gli ASFND sono stati definiti senza le ϵ -transizioni. Non e' quindi possibile qui usare in modo completamente automatico la costruzione del teorema di corrispondenza fra ASFND e ASF.



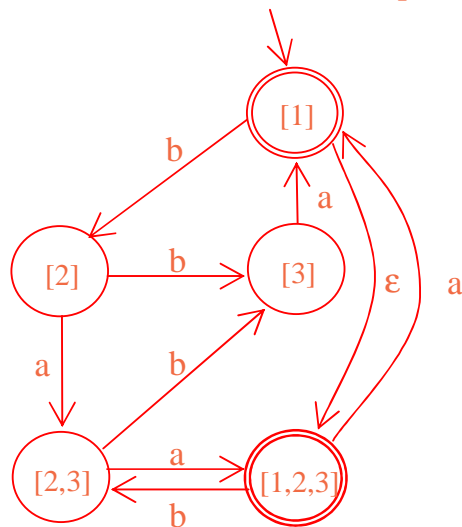
soluzione corretta (10 punti)

Un ASFND equivalente a quello dato è il seguente (la ϵ -transizione equivale ad un ramo percorribile senza 'bruciare' alcun carattere d'ingresso)



L'ASF equivalente è questo →

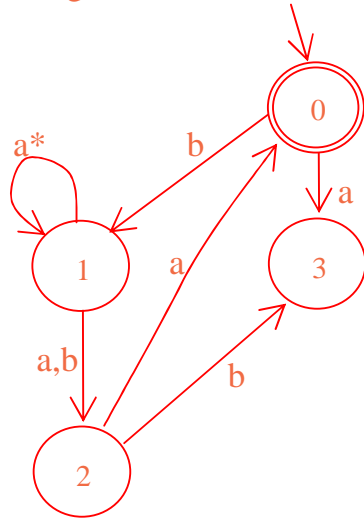
soluzione errata: non deterministica in quanto è rimasta una ϵ -transizione (4 punti)



soluzione errata (4 punti)

Il linguaggio riconosciuto dall'ASFND è $(ba^*(a+b)a)^*$ [affermazione falsa]

L'AFS è il seguente [non riconosce aaaa; non è deterministico]



Esercizio 3 (20%) Sia G una grammatica context free. Dimostra che esiste sempre una grammatica equivalente a G in forma normale di Greibach.

l'argomento di questo esercizio cade ora nel programma del secondo modulo

Esercizio 4 (20%) Mostrare una grammatica non contestuale per il linguaggio $a^n b^m a^m b^n$.

soluzione corretta (10 punti)

$S \rightarrow aXb$

$X \rightarrow aXb \mid bZa$

$Z \rightarrow ba \mid bZa$

soluzione errata [grammatica di tipo zero] (5 punti)

$S \rightarrow aBAb$

$BA \rightarrow aBAb \mid BXA$

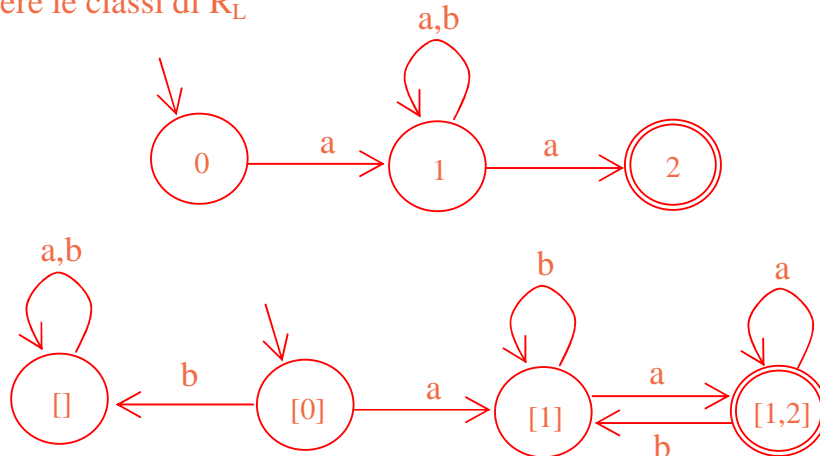
$BXA \rightarrow bBXAa \mid ba$

$B \rightarrow b$

$A \rightarrow a$

Esercizio 5 (20%) Si mostrino le classi di equivalenza di Myhill-Nerode per il linguaggio $a(a+b)^*a$.

soluzione corretta: trovare un AFSND, ottenere il corrispondente AFS, definire una classe di R_M per ogni stato, fondere le classi che ammettono gli stessi completamenti per ottenere le classi di R_L



$C[0] = \{\epsilon\} \rightarrow a(a+b)^*a$ (\rightarrow = “vado in uno stato finale con”)

$C[1] = \{a(b+aa^*b)^*\} \rightarrow (a+b)^*a$ (cioè qualsiasi cosa termini con ‘a’)

$C[1,2] = \{a(a+b)^*a\} \rightarrow (a+bb^*a)^*$

$C[] = \{\text{tutte i prefissi che, con qualsiasi suffisso, non danno una stringa } L\} \rightarrow \text{lambda}$

Le classi di R_M non possono essere fuse, dunque sono anche le classi di R_L

tipologie d’errore:

- scorda lo stato pozzo di AFS e definisce male la classe $C[]$ (-2 punti)
- cerca le classi di equivalenza di un AFSND (-3 punti)
- niente automi, solo le classi, manca proprio la classe di L (-3 punti)
- manca una transizione su AFS, quasi okay le classi (-1 punto)
- trova direttamente un AFS che è errato (-3 punti)
- descrive il teorema di Myhill-Nerode e poi ipotizza (da dove?) due classi (-8 punti)
- non è in grado di tradurre correttamente l’AFS in AFSND (-3 punti)