Automi, calcolabilità e complessità
Prova di esame del 7/11/2016
Prof.ssa E. Fachini

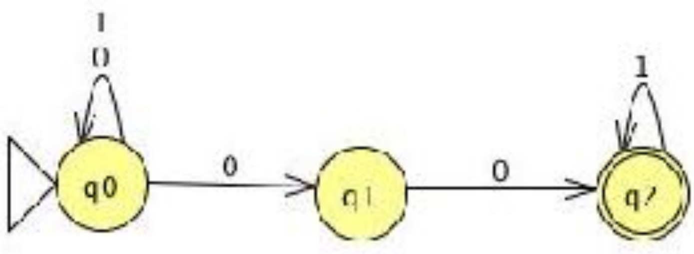
Parte prima

1) Si illustri la costruzione di un'espressione regolare che denota il linguaggio accettato da un DFA dato in input.
   Sol. Si veda libro di testo.

2) Se L è regolare su {0,1} possiamo dire che
   1. L_1 = {wy | w è in L e y in {0,1}*} è regolare?
   2. L_2 = {wrev(w) | w in L} è regolare?
      Si motivi le risposte.

Sol. L_1 è regolare perché è il prodotto di due linguaggi regolari, L e Σ*.
L_2 non è regolare, infatti preso L=Σ* è un linguaggio che si è dimostrato in aula non essere regolare, utilizzando il pumping lemma.
Riportiamo qui la prova. Basta per ogni n, prendere la parola w = 1^n001^n che è in L_2 e di lunghezza maggiore o uguale di n. Comunque prendiamo x,y,z tali che w = xyz, con |xy| ≤ n e |y| ≥ 1, facciamo vedere che possiamo prendere un valore di i tale xy^iz non è in L.
Poiché y è di soli 1 prendendo i=0, la parola xz è della forma 1^m001^n con m<n, quindi xz non è in L_2 .

3) Si costruisca un NFA che accetta le stringhe binarie che terminano con almeno 2 0 seguiti da un numero qualunque di 1.
Sol. Dallo stato iniziale q_0 leggendo 0 non deterministicamente la macchina rimane nello stato q_0 oppure va nello stato q1, perchè "indovina" che quello 0 è il primo dei due finali della parola, quindi in q1, leggendo 0 si va nello stato finale, dove si rimane leggendo solo 1.



Automi, calcolabilità e complessità
Prova di esame del 7/11/2016
Prof.ssa E. Fachini

Parte seconda

1) Si dimostri che se A $\leq_p$ B e A è NP-completo allora se B è in NP possiamo concludere che B è NP-completo.
Sol. Si veda il libro di testo.

2) Preso un linguaggio L non Turing riconoscibile il suo complemento può essere decidibile o Turing riconoscibile o nessuno dei due? Si motivi la risposta.
Sol. Poiché se un linguaggio è decidibile anche il suo complemento è decidibile, il complemento di L non può essere decidibile. Potrebbe essere Turing riconoscibile, infatti per esempio A_{TM} è un linguaggio Turing riconoscibile il cui complemento non è Turing riconoscibile.

3) Si consideri il linguaggio L = { <M> | M è una TM e L(M)={xx | x è in {0,1}*}
Si costruisca una riduzione da A_{TM} a L.
L è indecidibile? L è Turing riconoscibile? Si motivi le risposte.

Sol. La riduzione deve mandare un'istanza di A_{TM} , <T,w>, in un'istanza di L, M, in modo tale che <T,w> è in A_{TM} sse L è in L.
La riduzione è calcolata dalla seguente TM:
A
input <T,w>
output: la codifica della TM M di seguito descritta.

```
M
input x
if x non è della forma yy, per una qualche y in {0,1}* then rifiuta
else
        esegui T su w
        if T accetta w then accetta
        if  T rifiuta then rifiuta
```

Se T accetta w allora L(M)= {xx | x è in {0,1}*} altrimenti L(M)=∅ e quindi la riduzione è corretta.
In virtù della riduzione L è indecidibile.

Per la Turing riconoscibilità dovremmo costruire una TM che riconosce L. Osserviamo che la riduzione $A_{TM} \leq L$ già mi dice che complemento di L non è Turing riconoscibile, perchè equivale a $\neg A_{TM} \leq \neg L$ e sappiamo che $\neg A_{TM}$ non è Turing riconoscibile. Se si riduce $A_{TM} \leq \neg L$, si dimostra che anche L non è Turing riconoscibile, perchè equivale a dimostrare $\neg A_{TM} \leq L$.

```
R= input <M,w>
    output: la codifica della  TM T di seguito descritta.
        T = input x
        if x è della forma yy, per una qualche y in {0,1}* then accetta
        else
                esegui M su w
                if M accetta w then T accetta x
                if  M rifiuta w then T rifiuta x
```

Se M accetta w allora L(T) = {0,1}*, mentre se M non accetta w allora L(T) = {xx | x è in {0,1}*}. Quindi a un'istanza sì di A_{TM} si è associata un'istanza sì di ¬L, e a un'istanza no di ATM un'istanza no di ¬L, cioè una TM T in L.