# Algoritmi e Principi dell'Informatica
# Prima Prova in Itinere
25 Novembre 2013

**Avvisi importanti**
**Il tempo a disposizione è di 1 ora e 30 minuti.**
**Se non verranno risolti in maniera soddisfacente gli esercizi 1, 2, 3 (ossia ottenendo almeno 2 punti in *ognuno* di essi) non si procederà alla correzione dell'esercizio 4.**
**Chi otterrà una valutazione inferiore a 6/15-esimi non potrà sostenere la seconda prova e potrà iscriversi soltanto al secondo appello della sessione invernale.**

## Esercizio 1 (punti 3/15-esimi)
Dire se esiste un automa a stati finiti che riconosce il seguente linguaggio:
$L = \{a^n b^n c^n \mid n > 2\} \cup \{a^*b^*c^*\}$
Spiegare brevemente la risposta.

## Esercizio 2 (punti 3/15-esimi)

Quale delle seguenti formule del prim'ordine definisce correttamente i numeri di Fibonacci?

(si ricorda che l'n-esimo numero di Fibonacci è la somma dell'(n-1)-esimo e dell'(n-2)-esimo, per n > 1, mentre per n = 0 o 1 ha in valore convenzionale 0 e 1, rispettivamente.)

$\alpha$) $\forall n((n=0 \rightarrow fib(n) = 0) \wedge (n=1 \rightarrow fib(n) = 1) \wedge (n > 1 \rightarrow fib(n) = fib(n-1) + fib(n-2)))$

$\beta$) $\forall n((n=0 \rightarrow fib(n) = 0) \vee (n=1 \rightarrow fib(n) = 1) \vee (n > 1 \rightarrow fib(n) = fib(n-1) + fib(n-2)))$

Spiegare brevemente la risposta, preferibilmente fornendo esempi opportuni di valori di n e fib(n) che soddisfino o non soddisfino le formule $\alpha$) e $\beta$): per esempio, se m è l'n-esimo numero di Fibonacci e $\gamma$) è la formula corretta, la coppia n, m deve soddisfare $\gamma$) ma non l'altra formula e viceversa.

## Esercizio 3 (punti 3/15-esimi)
Si consideri il seguente (frammento) di programma:

```
while (x >= 0) {
	x = x*x - 2*x;
}
```

E' decidibile il problema di stabilire se la sua esecuzione termina *qualsiasi sia il valore iniziale della variabile x* all'inizio del ciclo? Spiegare brevemente la risposta.

## Esercizio 4  (punti 8/15-esimi)

Si consideri la seguente macchina astratta MA, descritta informalmente nel modo seguente:

- MA è dotata di un nastro di ingresso e uno di uscita;
- è dotata inoltre di un organo di controllo a stati finiti;
- e di una memoria ausiliaria consistente in una sequenza lineare di celle (come per altre classiche macchine astratte, ogni cella può contenere un carattere di un alfabeto finito) illimitata da ambo le parti; MA può accedere in lettura e scrittura solo ai due estremi della memoria ausiliaria.

Una mossa di MA consiste nel:

- Leggere il simbolo in corrispondenza della testina del nastro di ingresso, oppure effettuare una ε-mossa nella quale non viene letto nessun carattere;
- Leggere i due simboli alle estremità della memoria ausiliaria;

In base ai valori acquisiti e allo stato dell'organo di controllo, la MA può:

- Cambiare stato dell'organo di controllo;
- Scrivere, in sostituzione dei caratteri alle estremità della memoria ausiliaria, due rispettive stringhe;
- Scrivere una stringa, eventualmente nulla, sul nastro di uscita;
- Spostare a destra di una posizione o lasciare ferma la testina del nastro di lettura, a seconda che venga eseguita una mossa di lettura o una ε-mossa;
- Spostare a destra la testina del nastro di scrittura di un numero di posizioni corrispondente alla lunghezza della stringa scritta.

1. Si formalizzi completamente la descrizione di MA, specificandone i suoi elementi, le sue configurazioni e relative transizioni, il criterio di accettazione.
2. Si confronti la sua potenza con quella delle altre macchine astratte conosciute. Per semplicità il confronto può essere limitato alla capacità di riconoscimento di linguaggi, tralasciando la capacità di traduzione.

**NB:** la precedente descrizione informale, trascura volutamente di precisare alcuni dettagli (ad esempio, il criterio di accettazione, possibili situazioni anomale o di errore): la formalizzazione dovrà quindi risolvere eventuali ambiguità garantendo un funzionamento di MA utile a riconoscere una classe significativa di linguaggi.

## Tracce di soluzioni

**Esercizio 1**
Siccome $\{a^n b^n c^n \mid n > 2\}$ è contenuto in $\{a^*b^*c^*\}$, il linguaggio è ovviamente regolare.

**Esercizio 2**
La formula corretta è la $\alpha$), che è soddisfatta (solo) dalla serie {0, 1, 1, 2, 3, 5, 8, …}; la $\beta$) invece è soddisfatta da qualsiasi coppia n, fib(n) perché qualsiasi valore di n falsifica almeno uno degli antecedenti delle formule in disgiunzione, garantendo quindi che la formula complessiva sia sempre vera.

**Esercizio 3**
Il problema è "chiuso" perché il programma cui ci si riferisce è fissato e si chiede di stabilirne la terminazione *per ogni* valore iniziale della variabile x. E' quindi necessariamente *decidibile*.
Inoltre è anche *deciso*: infatti si può verificare che per x = 0 il programma non termina, pertanto non è vero che il programma termina *qualsiasi sia il valore iniziale della variabile x* all'inizio del ciclo.

**Esercizio 4**

**1.**
Una possibile formalizzazione della MA (in versione deterministica) è la seguente.

- Una MA è una tupla $(Q, I, \Gamma, O, q_0, Z_0, F, \delta, \eta)$, dove i vari simboli hanno il tradizionale significato della teoria degli automi –stati dell'organo di controllo, alfabeti, ecc.-. Le funzioni $\delta$, $\eta$ sono hanno rispettivamente i domini e codomini seguenti:
  - $\delta: Q \times (I \cup \{\varepsilon\}) \times \Gamma^2 \to Q \times (\Gamma^*)^2$
    /* dato lo stato dell'OC, il simbolo in lettura o eventuale $\varepsilon$-mossa, i due simboli alle estremità della memoria ausiliaria, la macchina si porta in un nuovo stato e riscrive alle due estremità della memoria le corrispondenti stringhe di caratteri in $\Gamma^*$*/.
    Le usuali restrizioni sulla definizione di $\delta$ vanno applicate per garantire il determinismo.
  - $\eta: Q \times (I \cup \{\varepsilon\}) \times \Gamma^2 \to O^*$
    /* in corrispondenza della mossa definita dalla $\delta$ viene scritta la stringa in $O^*$ sul nastro di uscita.*/
- Una *configurazione* di MA è una tupla $c = \langle q, x, \alpha, z\rangle$ , con $q \in Q$, $x, \in I^*$, $\alpha \in \Gamma^*$, $z \in O^*$. La relazione di transizione tra configurazioni
  $c = \langle q, x, \alpha, z\rangle \mid\text{--}\ c' = \langle q', x', \alpha', z'\rangle$ è definita nel modo tradizionale, con il seguente aspetto significativo:
  - se $x = i.y$ , $\alpha = A\beta B$ e $\delta(q, i, A, B) = (q', \zeta, \theta)$, allora
    $x' = y$, $\alpha' = \zeta\beta\theta$;
    z' è definita analogamente sulla base di $\eta$.
  - l'effetto di una $\varepsilon$-mossa è definito analogamente.
- Una stringa x è accettata se $\langle q_0, x, Z_0, \varepsilon\rangle \mid\text{--}^* \langle q, \varepsilon, \alpha, z\rangle$, con $q \in F$. In tal caso z è l'eventuale traduzione $\tau(x)$ definita dal trasduttore. (Si noti però che, anche in caso di MA deterministica la traduzione da essa definita potrebbe essere multipla a causa di possibili $\varepsilon$-mosse a scansione dell'input completata). Si noti che, a causa di quanto

verrà precisato in seguito a proposito della potenza della MA, non è decidibile se una tale MA per un generico ingresso x termini o meno la computazione.

- **NB**. la definizione precedente espone al rischio di conflitti nel caso $|\alpha| = 1$, ossia $\alpha$ = A: in tal caso si potrebbe verificare $\delta$(q, i, A, A) = (q', $\zeta$, $\theta$) con $\zeta \neq \theta$ e la mossa della macchina non potrebbe avvenire in modo coerente. Una possibile soluzione consiste nel "bloccare" la MA qualora si verifichi la condizione suddetta; l'alternativa di imporre $\zeta = \theta$ quando $\delta$(q, i, A, A) = (q', $\zeta$, $\theta$) sarebbe invece troppo restrittiva perché limiterebbe il comportamento della MA anche in casi in cui $\alpha$ = A$\beta$A. Dovrebbe però fare eccezione il caso iniziale $\delta(q_0$, i (o $\varepsilon$) , $Z_0$, $Z_0$) per permettere alla macchina di eseguire la prima mossa senza bloccarsi subito; in alternativa si potrebbe definire come configurazione iniziale $<q_0, x, Z_0Z_0, \varepsilon>$.

2.
La MA può facilmente simulare sia un automa a doppia pila (tenendo un separatore $Z_0$ tra la parte sinistra e la parte destra della memoria in modo che esse possano simulare separatamente le due pile) sia un automa a coda. In entrambi i casi è nota l'equipotenza di queste macchine con la macchina di Turing.