

# Fondamenti dell'Informatica

Compito scritto

28 settembre 2005

Cognome:

Nome:

Matricola:

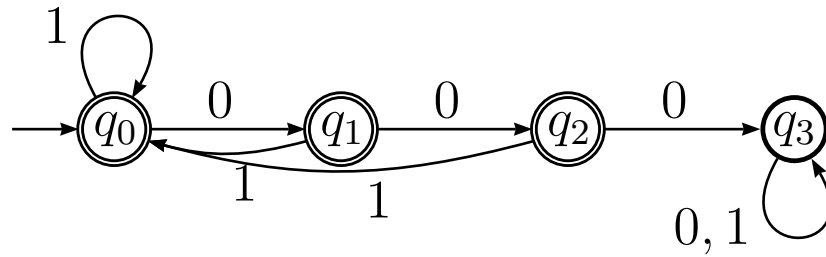
## Note

1. Per i quiz a risposta multipla, fare una croce sulla/e lettera/e che identifica/no la/e risposta/e desiderata/e.
2. Per i quiz a risposta multipla, c'è sempre almeno una risposta corretta. Talvolta ci sono più risposte corrette. Si richiede che siano marcate *tutte e sole* le risposte corrette. In altre parole, una crocetta in più o in meno invalida l'esercizio.
3. Per i quiz descrittivi e gli esercizi, la risposta va data sulla stessa facciata che contiene il testo dell'esercizio. Lo spazio lasciato a questo scopo è sempre sufficiente.
4. È possibile usare il retro dei fogli per eventuali calcoli e verifiche.
5. L'orario di consegna scritto alla lavagna è tassativo.
6. Non è consentita la consultazione di alcunché.
7. Gli esercizi verranno corretti solo se il numero di punti conseguiti nei quiz supera una certa soglia. In caso contrario il compito è insufficiente. Le soglie sono:
  - per i matematici: 11 punti su 20
  - per gli informatici: 16 punti su 30

## Quiz per tutti

1. (2 punti) Su un alfabeto di un numero finito di simboli si possono definire:
  - (A) Solo un numero finito di linguaggi finiti;
  - (B) Solo un numero finito di linguaggi regolari;
  - (C) Un numero infinito di linguaggi finiti;
  - (D) Un numero infinito di linguaggi regolari.
2. (2 punti) Se un  $\epsilon$ -NFA accetta un linguaggio  $L$ :
  - (A)  $L$  è un linguaggio regolare;
  - (B)  $L$  può non essere un linguaggio regolare;
  - (C)  $L$  contiene necessariamente la stringa vuota;
  - (D)  $L$  è necessariamente vuoto.
3. (2 punti) Se un NFA accetta un linguaggio  $L$ :
  - (A) esiste una espressione regolare che descrive  $L$ ;
  - (B) esiste un  $\epsilon$ -NFA che accetta  $L$ ;
  - (C) esiste un DFA che accetta  $L$ ;
  - (D) esiste un algoritmo per decidere se una stringa data appartiene a  $L$ .
4. (2 punti) I *programmi* di un qualche formalismo Turing-equivalente soddisfano tutti i requisiti degli *algoritmi* eccettuato quello/quelli di:
  - (A) lunghezza finita; (B) memoria illimitata; (C) sicura terminazione;
  - (D) *né (A) né (B) né (C)*.
5. (2 punti) Sono funzioni ricorsive di base:
  - (A) la funzione costante 0; (B) la funzione successore  $S$ ;
  - (C) la funzione  $i$ -esima proiezione; (D) la funzione di Ackermann.

6. (2 punti) Si descriva, usando la notazione insiemistica, il linguaggio accettato dall'automa  $M$  qui sotto:



$L(M) =$

7. (2 punti) Definire la somma come funzione primitiva ricorsiva partendo dalle funzioni ricorsive di base.

8. (3 punti) Dare un'espressione regolare  $e$  per il seguente linguaggio su  $\{0, 1\}$ :

$L = \{\text{tutte le stringhe che contengono un numero dispari di '0'}\}.$

$e =$

**9.** (3 punti) Si consideri la Macchina di Turing definita dal seguente schema:

Q	0	1	\$
$q_0$			$q_1$ \$ R
$q_1$	$q_2$ 1 R	$q_1$ 1 R	
$q_2$	$q_0$ 0 L	$q_1$ 0 R	

Si supponga che la macchina cominci la computazione nello stato  $q_0$ , avendo per input sul nastro la stringa 1010, con la testina posizionata sul primo simbolo \$ alla sinistra della stringa stessa.

Mostrare la computazione come sequenza di descrizioni istantanee.

## Quiz per gli “informatici”

**10.** (1 punto) Se l'intersezione  $L$  di due linguaggi  $L_1$  e  $L_2$  è un linguaggio libero dal contesto, allora  $L_1$  e  $L_2$  sono liberi dal contesto.

(A) vero; (B) falso

**11.** (2 punti) Data una grammatica libera dal contesto che genera un linguaggio  $L$  non vuoto e senza  $\epsilon$ , è sempre possibile determinare:

- (A) Una grammatica priva di simboli inutili che genera  $L$ ;
- (B) Una grammatica in forma normale di Chomsky che genera  $L$ ;
- (C) Una grammatica in forma normale di Greibach che genera  $L$ ;
- (D) *né (A) né (B) né (C).*

**12.** (2 punti) Se  $L_1$  e  $L_2$  sono linguaggi liberi dal contesto ma non regolari allora necessariamente:

- (A)  $L_1 \setminus L_2$  non è libero dal contesto;
- (B)  $L_1 \cup L_2$  non è libero dal contesto;
- (C)  $L_1 \cap L_2$  non è libero dal contesto;
- (D)  $L_1 \cap L_2$  è libero dal contesto ma non regolare;
- (E) *nessuna di queste.*

**13.** (3 punti) Si dia una grammatica che generi le stringhe del tipo `utente@dominio.ext` dove `utente` è una stringa alfanumerica di almeno due caratteri che comincia e finisce con una lettera, `dominio` è una stringa alfanumerica non vuota, `ext` è una stringa di due o tre simboli costituita di sole lettere. Si considerino date le produzioni per cifre e lettere  $C \mapsto 0 \mid 1 \mid \dots \mid 9$ ,  $L \mapsto a \mid b \mid \dots \mid z$ , e si considerino solo lettere minuscole.

14. (2 punti) Si consideri l'automa a pila

$$M = \langle \{q\}, \{a, b\}, \{S, A, B\}, \delta, q, S, \emptyset \rangle$$

dove

$$\delta(q, a, S) = \{(q, B)\},$$

$$\delta(q, a, A) = \{(q, \epsilon), (q, S)\},$$

$$\delta(q, a, B) = \{(q, BB)\},$$

$$\delta(q, b, S) = \{(q, A)\},$$

$$\delta(q, b, A) = \{(q, AA)\},$$

$$\delta(q, b, B) = \{(q, \epsilon), (q, S)\}.$$

Si mostri un'esecuzione dell'automa (sequenza di descrizioni istantanee) che porti all'accettazione per pila vuota della stringa *baba*.

## Esercizio 1 (per tutti)

Si supponga di estendere il metalinguaggio delle espressioni regolari con il costrutto ‘prefix’. Dato un linguaggio  $L$ ,  $\text{prefix}(L)$  denota il linguaggio di tutti i prefissi (non necessariamente propri e possibilmente vuoti) di stringhe di  $L$ . Si dica se l’aggiunta di ‘prefix’ porta ad un formalismo più potente delle espressioni regolari, dimostrando formalmente ogni affermazione. (Suggerimento: procedere per induzione strutturale).

## Esercizio 2 (per tutti)

Si dimostri che l'insieme delle stringhe accettate da un DFA  $M$  con  $n$  stati è:

1. non vuoto se e solo se  $M$  accetta una stringa di lunghezza inferiore a  $n$ ;
2. infinito se e solo se l'automa accetta una stringa di lunghezza  $\ell$  con  $n \leq \ell < 2n$ .



### Esercizio 3 (solo per gli “informatici”)

Si consideri il linguaggio

$$L = \{ ucw \mid u, w \in \{a, b\}^* \text{ hanno lo stesso numero di 'a' } \}.$$

Si dia una grammatica che lo generi e si progetti un automa a pila che lo accetti (per stato finale o per pila vuota, a scelta del candidato).