

FONDAMENTI DI PROGRAMMAZIONE - CdL in MATEMATICA

PROVA SCRITTA DEL 14/9/2012

Scrivere **in stampatello** COGNOME, NOME e MATRICOLA su ogni foglio consegnato

N.B.: Negli esercizi di programmazione, viene valutata anche la leggibilità del codice proposto. Inoltre, non è consentito l'uso di istruzioni che alterino il normale flusso dell'esecuzione (come, ad esempio, **continue**, **break** e istruzioni di **return** all'interno di cicli che ne provochino l'uscita forzata). Infine non è consentito l'uso di variabili statiche.

Laddove è utilizzato, il tipo **boolean** è definito da **typedef enum {false, true} boolean;**

ESERCIZIO 1 (6 punti)

Si consideri il linguaggio $L = \{x3y \mid x \in \{2, 01\}^*, y \in \{1, 3\}^*\}$ sull'alfabeto $\{0, 1, 2, 3\}$.

- definire un automa deterministico che accetti tutte e sole le stringhe di L ;
- scrivere una grammatica regolare equivalente all'automa dato.

ESERCIZIO 2 (6 punti)

Si scriva una funzione **ricorsiva** che legga una sequenza di caratteri in ingresso terminati dal carattere **z** e che restituisca il prodotto delle cifre presenti nella sequenza. Ad es. sulla sequenza $\{'3' \ 'd' \ '5' \ 'A' \ '7' \ 'z' \}$ la funzione dovrebbe restituire $3 \times 5 \times 7 = 105$.

Si risolva il problema, senza utilizzare strutture di appoggio, tipo liste o vettori.

ESERCIZIO 3 (7 punti)

Si scriva una funzione **iterativa** che, dato un array di interi **a**, di dimensione **dim**, e un valore **x** intero e maggiore di 1,

- restituisca **true** se contiene i primi **dim** elementi $a(i) = \sum_{h=0}^i x^h$ della serie $\sum_{i=0}^{\infty} x^i$ nell'ordine in cui vengono generati dalla serie, e che restituisca **false** altrimenti;
- la funzione deve inoltre e comunque stampare l'ultimo numero pari tra quelli che appaiono nell'array.

ESERCIZIO 4 (11 punti)

In una lista *circolare* tutti i nodi vengono collegati in un cerchio continuo. L'ultimo elemento non punta a **NULL**, ma al primo nodo della lista.

- (1 punti) Definire i tipi opportuni per la rappresentazione di nodi **elem** con il campo informazione dato da un intero.
- (5 punti) Scrivere una procedura **ricorsiva** che data una lista circolare, un elemento **elem** e un valore **k**, inserisca **elem** prima della prima occorrenza del nodo che contiene il valore **k**, che supponiamo esistere sempre.
- (5 punti) Scrivere una funzione **iterativa** che data una lista circolare e un valore **k** calcoli tutte le occorrenze del valore **k**.