

FONDAMENTI DI PROGRAMMAZIONE - CdL in MATEMATICA

PROVA SCRITTA DEL 21/9/2011

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)

- Costruire due automi a stati finiti *deterministici* A_1 e A_2 che riconoscano rispettivamente i linguaggi L_1 e L_2 :

$$L_1 = \{w \in \{0,1\}^* \text{ t.c. } |w|_0 = 1 \pmod{3}\}$$

$$L_2 = \{w \in \{0,1\}^* \text{ t.c. } |w|_1 = 2 \pmod{3}\}$$

- Costruire quindi l'automa intersezione A che riconosca il linguaggio $L_1 \cap L_2$.
- Scrivere la grammatica corrispondente all'automa A_1 .

ESERCIZIO 2 (6 punti)

Si scriva una procedura **ricorsiva** che legge una sequenza di numeri interi terminati da 0 e stampa tutti e soli gli elementi della sequenza (esclusi gli estremi) che sono immediatamente preceduti da un numero inferiore e immediatamente seguiti da un numero superiore. Se ad esempio la sequenza fosse: 2 5 3 7 3 9 10 11 12 0 l'output atteso sarebbe il seguente: 9 10 11

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

ESERCIZIO 3 (7 punti)

Si scriva una funzione che riceva in input un array di caratteri e la sua dimensione, e che restituisca `true` se e solo se tutti i suoi caratteri sono distinti. Si ignorino le differenze tra minuscole e maiuscole.

ESERCIZIO 4 (11 punti)

Si rappresenti il *grafico* di una funzione definita sul piano cartesiano come una lista **ordinata** di punti, fatti di tre campi: x ascissa, y ordinata e un campo *succ* che punta al punto successivo, dove se FUN è la funzione rappresentata, il punto (x, y) è $FUN(x) = y$.

- (1 punti) Definire i tipi opportuni per la rappresentazione indicata.
- (5 punti) Scrivere una funzione *iterativa* che aggiunga un punto (x, y) . Se il punto con ascissa x è già presente, si modifichi il solo campo y .
- (5 punti) Scrivere una funzione **ricorsiva** che calcoli la *lunghezza* del grafico visto come una linea spezzata che passa per i punti del grafico. Usare `double sqrt(double)` presente nella libreria `math.h`.