

NOTE SULLO SVOLGIMENTO DELLA PROVA SCRITTA:

- SCRIVERE IL PROPRIO NOME, COGNOME, DATA E NUMERO DI MATRICOLA SU OGNI FOGLIO UTILIZZATO PER LO SVOLGIMENTO DELLA PROVA
- RICONSEGNARE TUTTI I FOGLI. NON SCRIVERE A MATITA.
- SPEGNERE I TELEFONINI
- NON È POSSIBILE UTILIZZARE CALCOLATRICI
- È POSSIBILE CONSULTARE SOLO LA DISPENSA SUL LINGUAGGIO ASSEMBLER DISPONIBILE SULLA CATTEDRA
- I PRIMI DUE ESERCIZI VALGONO 10 PUNTI; GLI ULTIMI 2 VALGONO 5 PUNTI
- TEMPO PER I PRIMI TRE ESERCIZI: 1 ORA E 45 MINUTI TEMPO PER ESERCIZI 4.1 E 4.2: 30 MIN.

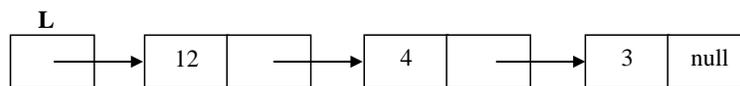
ESERCIZIO 1

1) Sia data la struttura elem seguente:

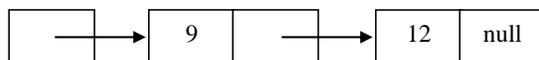
```
struct elem {int info; elem * pun; };
```

Scrivere una funzione che, data una lista L di elementi di tipo elem, restituisca una nuova lista ordinata per valori crescenti del campo info che contiene tutti gli elementi della lista L che sono multipli di 3. La lista L non deve essere modificata.

Per esempio, se la funzione viene chiamata con la lista L seguente:



la funzione restituisce la lista seguente:



ESERCIZIO 2

Scrivere una funzione che, data una matrice A di $n \times m$ interi ed un file, passati come argomento alla funzione, scriva nel file l'indice della colonna la cui somma degli elementi è la minima tra le m colonne. Nel caso in cui più colonne abbiano somma minima, la funzione restituisce l'indice più piccolo. Per esempio, se la funzione viene chiamata con la matrice A seguente:

$$A = \begin{bmatrix} 1 & 2 & 0 \\ 0 & 3 & 1 \\ 9 & 4 & 3 \\ 1 & 5 & 1 \end{bmatrix}$$

la funzione scrive 2 nel file.

ESERCIZIO 3

Sia data la struttura elem seguente:

```
struct elem {int info; elem * pun; };
```

Scrivere una funzione ricorsiva che, data una lista di elementi di tipo elem passata come argomento alla funzione, elimini tutti gli elementi dalla lista.

ESERCIZIO 4

4.1

Data la rappresentazione *in complemento a due* $A=100001$ di un intero in base 2 su $p = 6$ bit, dire di quale intero a è la rappresentazione. Qualora si fosse trattato di una *rappresentazione con Bias*, quale sarebbe stato invece l'intero corrispondente?

4.2

Dire cosa stampa a video il seguente programma assembler, qualora vengano fornite in ingresso le due cifre meno significative del proprio numero di matricola.

```
_main:  CALL  inbyte
        AND  $0x1F,%AL
        OR   $0x10,%AL
        CALL outbyte
        MOV  $5,%AH
        CALL s1
        MOV  %BL,%AL
        CALL outbyte
        RET

s1:     MOV  $0,%BL
lab:    SUB  %AH,%AL
        JC   ends1
        INC  %BL
        JMP  lab
ends1:  RET
```

```
.INCLUDE "utility"
```

PROMEMORIA

CALL inbyte: legge da tastiera due caratteri (purché compresi fra '0' e '9' oppure fra 'A' e 'F'), li interpreta come cifre esadecimali e li utilizza per inizializzare AL.

Esempio: qualora venissero inseriti '4' e 'B' in %AL ci finirebbe 0100-1011 (0x4B).

AND SRC,DST: calcola l'AND logico bit a bit fra SRC e DST e mette il risultato in DST.

OR SRC,DST: calcola l'OR logico bit a bit fra SRC e DST e mette il risultato in DST.

CALL outbyte: stampa a video le due cifre esadecimali corrispondenti al contenuto di AL.

Esempio: se in AL si trova 1010-0100 stamperà a video "A4".

SUB SRC,DST: Sottrae SRC a DST e memorizza il risultato in DST

INC REG: Incrementa il registro REG di una unità (equivale ad ADD \$1,REG).

JC label: Salta a 'label' solo se il carry flag vale 1.

JMP label: Salta a 'label' (sempre).
