

**NOTE SULLO SVOLGIMENTO DELLA PROVA SCRITTA:**

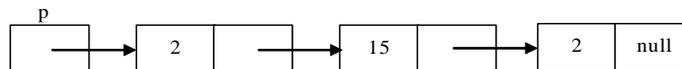
- SCRIVERE IL PROPRIO NOME, COGNOME, DATA E NUMERO DI MATRICOLA SU OGNI FOGLIO UTILIZZATO PER LO SVOLGIMENTO DELLA PROVA
- RICONSEGNARE TUTTI I FOGLI. NON SCRIVERE A MATITA.
- SPEGNERE I TELEFONINI
- NON È POSSIBILE UTILIZZARE CALCOLATRICI
- È POSSIBILE CONSULTARE SOLO LA DISPENSA SUL LINGUAGGIO ASSEMBLER 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 L'ESERCIZIO 4: 30 MIN.

**Esercizio 1**

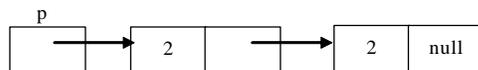
```
struct elem {int info; elem * pun; };
```

Scrivere una funzione che, data una lista di elementi di tipo elem, elimina dalla lista tutti gli elementi con campo informazione dispari.

Se chiamata con argomento la lista puntata da p,



la funzione modificherà la lista come segue:

**Esercizio 2**

Scrivere una funzione che, dati due vettori v1 e v2 di elementi di tipo int ordinati in ordine crescente, restituisca un vettore con tutti gli elementi di v1 e v2 ordinato in ordine decrescente. Il vettore deve anche essere salvato nel file di nome "output.txt". I vettori v1 e v2 possono contenere duplicati e possono avere dimensione diversa.

**Esercizio 3**

Scrivere una funzione ricorsiva che, data una matrice quadrata di elementi di tipo intero passata come argomento alla funzione, restituisca la somma di tutti gli elementi della matrice. Se chiamata con argomento la matrice mat seguente, la funzione restituirà 6.

```

mat =
  1  0  0
  0  2  0
  0  0  3
  
```

## Esercizio 4

### 4.1

Trovare la rappresentazione in virgola fissa del numero 0.75 su  $p = 6$  bit, usando  $f = 5$  bit per la parte frazionaria.

Inoltre, dato il numero -6.75, calcolarne la rappresentazione in virgola mobile usando la *half precision*.

### 4.2 Dire cosa stampa a video il seguente programma assembler.

```
str:      .ASCII "Tre"
_main:    MOV     $'a', %DH
          SUB     $'A', %DH
          MOV     %DH, %AL
          CALL    outbyte
          MOV     $2,%DL
          MOV     $str, %EBX
          MOV     (%EBX), %AL
          CALL    output
lab:      INC     %EBX
          MOV     (%EBX), %CL
          SUB     %DH, %CL
          MOV     %CL, %AL
          CALL    output
          DEC     %DL
          JNZ    lab
fine:     RET
          .INCLUDE "utility"
```

#### PROMEMORIA

CALL outbyte	stampa a video la coppia di caratteri ascii associati alla parte alta e bassa del naturale contenuto nei 4 bit piu' significativi e meno significativi di AL. Esempio: qualora in AL vi fosse 0011-0101 stamperebbe a video i caratteri "35"
CALL output	stampa a video il carattere ASCII la cui codifica si trova in AL. Esempio: qualora in AL si trovasse 0011-0101 stamperebbe a video il carattere '5'
JNZ label	salta a label specificata tutte le volte che il risultato dell'ultima operazione e' diverso da zero
SUB srg,dest	aggiorna il contenuto del registro destinazione con la differenza fra il suo contenuto precedente ed il contenuto del registro sorgente.