

**PROVA SCRITTA DEL CORSO DI  
FONDAMENTI DI INFORMATICA 1  
CORSO DI LAUREA IN INGEGNERIA CHIMICA ED INGEGNERIA MECCANICA  
23/2/2016**

**MOTIVARE IN MANIERA CHIARA LE SOLUZIONI PROPOSTE A CIASCUNO DEGLI ESERCIZI SVOLTI**

**ESERCIZIO 1 (5 punti)**

Convertire in binario il valore, espresso in decimale, 320.25. Si usi la rappresentazione in virgola fissa. Si calcoli poi il complemento a due della parte intera usando 8 bit.

**ESERCIZIO 2 (solo 5 crediti: 5 punti)**

Semplificare, indicando le proprietà od i teoremi dell'algebra booleana da applicare, la seguente espressione:  $AB' + ABC + A'$ .

**ESERCIZIO 2 (solo 6 crediti: 5 punti)**

Siano date le istanze di relazione:

**Docenti**

Nome_D	Cognome_D	Matricola	Corsi_tenuti
Gian Luca	Marcialis	1	2
Giulio	Concas	2	4
Michele	Marchesi	3	5
Giuliano	Armano	4	3
Giorgio	Fumera	5	2
Fabio	Roli	6	5

**Allievi**

Matricola	Nome	Cognome	Max_voto	Docente
1	Giancarla	Ferrai	30	3
2	Marco	Pistis	30	2
3	Maria	Desogus	18	1
4	Silvia	Ferrai	30	1
5	Gianguido	Pistis	25	2

(2 punti) Indicare la cardinalità ed il grado di ognuna delle due relazioni.

(3 punti) Scrivere in SQL l'interrogazione: "Si trovi il nome e cognome dei professori che hanno registrato esami a più di uno studente".

**ESERCIZIO 3 (5 punti)**

Spiegare in modo chiaro e sintetico la differenza tra rete di calcolatori e sistema distribuito.

**ESERCIZIO 4 (18 punti)**

- (4 punti) Scrivere una funzione Python `separa(x)` che a partire da un numero reale  $x$  restituisce in una lista la parte intera e la parte frazionaria separate.
- (5 punti) Scrivere una funzione Python che, a partire da un file di testo `nomefile` contenente una sequenza di valori reali, restituisca una lista di liste contenente per ciascun valore letto: il valore, la sua parte intera e la sua parte frazionaria. Si usi la funzione `separa(x)` sviluppata in precedenza.
- (3 punti) Scrivere una funzione Python `divisibile(x)` che restituisce `True` se  $x$  è divisibile per 2, `False` altrimenti.
- (6 punti) Scrivere uno script Python che, a partire da un file `nomefile` composto da una sequenza di valori reali, scriva su file "output.txt" quei valori la cui parte intera risulta divisibile per 2. Utilizzare le funzioni sviluppate nei punti precedenti.

### Soluzione dell'esercizio 1

Per tradurre in binario il valore espresso in virgola fissa 320.25, dobbiamo applicare l'algoritmo delle divisioni successive per la parte intera e quello delle moltiplicazioni successive per la parte frazionaria, ottenendo:

Parte intera	Parte frazionaria
$320/2 = 160$ , resto <b>0</b>	$0.25*2 = 0.50$
$160/2 = 80$ , resto <b>0</b>	$0.5*2 = 1.00$
$80/2 = 40$ , resto <b>0</b>	
$40/2 = 20$ , resto <b>0</b>	
$20/2 = 10$ , resto <b>0</b>	
$10/2 = 5$ , resto <b>0</b>	
$5/2 = 2$ , resto <b>1</b>	
$2/2 = 1$ , resto <b>0</b>	
$1/2 = 0$ , resto <b>1</b>	

Quindi:  $320.25_{10} = 101000000.01_2$

Ovviamente 8 bit non sufficienti per rappresentare 320, tantomeno è possibile calcolarne il complemento a due.

### Soluzione dell'esercizio 2

$AB' + ABC + A' =$   
 $= A * (B' + BC) + A' =$  (prop. Distributiva della somma)  
 $= A * (B' + B) * (B' + C) + A' =$  (prop. Distributiva del prodotto)  
 $= A * (B' + C) + A' =$  (prop. Elemento complementare)  
 $= (A' + A) * (A' + (B' + C)) =$  (prop. Distributiva del prodotto)  
 $= A' + B' + C$  (el. Complementare, elemento neutro, prop. Associativa)

### Soluzione dell'esercizio 2 (6 crediti)

Cardinalità e grado di una relazione corrispondono rispettivamente al numero degli attributi ed al numero delle tuple. Per cui sono uguali a 4 e 6 per la prima relazione, a 5 e 5 per la seconda.

```
SELECT DISTINCT NOME_D, COGNOME_D
FROM DOCENTI, ALLIEVI
WHERE ALLIEVI.DOCENTE=DOCENTI.MATRICOLA
GROUP BY ALLIEVI.DOCENTE
HAVING COUNT(*) >=2
```

### Soluzione dell'esercizio 3.

Vedi dispense, Cap. 2, pagg. 21-33.

#### Soluzione dell'esercizio 4.

```
#Esercizio 1
def separa(x):
    i=int(x)
    f=x-i
    return [i,f]

#Esercizio 2
def leggi(nomefile):
    l=[]
    infile=open(nomefile,"r")
    line=infile.readline()
    while line!="":
        x=float(line)
        v=separa(x)
        v=[x] + v
        l = l + [v]
        line=infile.readline()
    infile.close()
    return l

#Esercizio 3
def divisibile(x):
    return not x%2

#Esercizio 4
nomefile="input.txt"
l=leggi(nomefile)
outfile=open("output.txt","w")
for v in l:
    if divisibile(v[1]):
        outfile.write(str(v[0])+"\n")
outfile.close()
```