

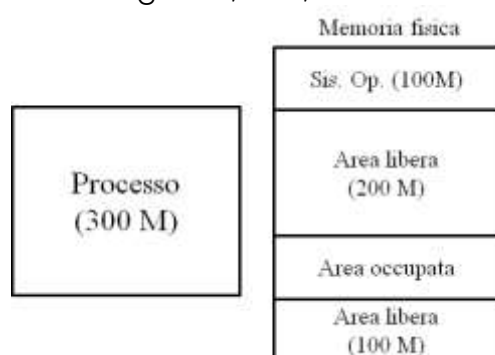
**PROVA SCRITTA DEL CORSO DI
FONDAMENTI DI INFORMATICA 1
CORSI DI LAUREA IN INGEGNERIA CHIMICA ED INGEGNERIA MECCANICA
19/7/2016**

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

NOME: _____ **COGNOME:** _____ **MATRICOLA:** _____

ESERCIZIO 1 (5 punti)

- 1) (3 punti) Lo stato della memoria è illustrata nella figura a fianco (i valori tra parentesi tonde indicano la dimensioni in byte delle relative aree e del processo). Dire se sia possibile allocare il processo in figura e, se sì, indicare una possibile metodologia.



- 2) (2 punti) Scrivere in modo chiaro e sintetico cos'è una shell dei comandi.

ESERCIZIO 2 (5 punti)

Si hanno a disposizione 8 bit per rappresentare dei valori numerici in complemento a due.

- (2 punti) Quali sono le configurazioni corrispondenti al valore massimo e minimo rappresentabili, e qual è il loro valore espresso in notazione decimale?
- (3 punti) Dati i valori 01100001 e 10001111, sommarli e verificare la corrispondenza del risultato con i relativi valori decimali.

ESERCIZIO 3 (5 punti)

- (2 punti) Indicare grado e cardinalità delle relazioni in figura.
- (3 punti) Dato lo schema logico in figura, scrivere la query: "Estrarre il nome dei corsi i cui esami sono stati sostenuti da almeno tre studenti".

Relation Studenti

(Matricola: char(5) PRIMARY KEY NOT NULL,
Nome: char(20) NOT NULL,
Data-N: date,
Anno-Corso: integer,
Corso-Laurea: char (1));

Relation Corsi

(Cod-Corso: char(6) PRIMARY KEY NOT NULL,
Titolo: char(50),
Docente: char(20));

Relation Esami

(Cod-Corso: char(6) NOT NULL,
Matricola: char(5) NOT NULL,
Data-E: date NOT NULL,
Voto: integer NOT NULL,
PRIMARY KEY (Cod-Corso, Matricola));

ESERCIZIO 4 (18 punti)

Un robot è dotato di un "occhio", ovvero una telecamera in grado di identificare i volti circostanti associandoli a delle persone. Il riconoscimento avviene attraverso un numero reale compreso tra 0 ed 1 e chiamato punteggio di somiglianza: un punteggio pari a 0 significa totale assenza di somiglianza, un punteggio pari a 1 significa totale somiglianza. Ogni volta che il robot si "guarda intorno", rileva i volti e scrive in una riga di un file "volti.txt" il punteggio calcolato e il nome della persona a cui è associato quel punteggio. Tuttavia, non tutte le associazioni fatte dal robot sono affidabili. Sono considerate affidabili solo quelle associazioni il cui punteggio è *strettamente* maggiore del valore 0.5.

A questo punto vi viene richiesto di scrivere un programma Python che permetta di rilevare, dato il file prodotto dal robot, il numero di associazioni affidabili e di indicare a quali persone corrispondano queste associazioni.

Per questo scopo, decidete di memorizzare le due informazioni memorizzate in ogni riga del file dato in un dizionario, dove la chiave è data dal nome della persona e il valore da una lista di punteggi associati alla persona stessa.

Vanno poi implementate le seguenti funzioni:

- 1) (3 punti) Funzione `leggi(nomeFile)`: legge da file, il cui nome è passato come parametro, i dati relativi ad i punteggi ed alla persona associate e restituisce un dizionario secondo la specifica di cui sopra.
- 2) (4 punti) Funzione `estrai_affidabili(lista, punteggio)`, che ricevendo in ingresso una lista di punteggi ed un punteggio, estrae il numero di valori superiori al punteggio dato.
- 3) (5 punti) Funzione `estrai_soggetti(dizionario, punteggio)`, che ricevendo in ingresso un dizionario secondo specifica ed un punteggio, restituisce la percentuale complessiva di punteggi superiori a quello dato e una lista contenente i nomi dei soggetti con almeno un punteggio.
- 4) (3 punti) Funzione `stampa_risultati(soggetti, p)` che ricevendo in ingresso una lista di soggetti e la percentuale di associazioni "affidabili", inteso com, stampa a video il punteggio e la sequenza dei soggetti separati da spazio.

Utilizzando le predette funzioni, si scriva infine la sequenza di istruzioni Python che compia l'elaborazione richiesta a partire dal file prodotto dal robot (3 punti).

ESERCIZIO 1 (5 punti)

- 1) L'area libera in memoria è pari alla dimensione del processo. E' possibile quindi allocare il processo utilizzando il metodo di paginazione con pagine di dimensione massima pari a 100M. In questo modo, ad esempio, due pagine verrebbero allocate nell'area libera di 200M e una pagina nell'area da 100M.
- 2) Vedi dispense del corso (Cap. 3 pp. 25-29).

ESERCIZIO 2 (5 punti)

1. I valori massimi e minimi sono dati dalle configurazioni 01111111, pari a 2^7-1 , e 10000000, pari a -2^7 .
2. Il primo valore corrisponde a $2^6 + 2^5 + 1 = 64 + 32 + 1 = 97$, il secondo a $-2^7 + 2^3 + 2^2 + 2 + 1 = -128 + 8 + 4 + 2 + 1 = -113$. La loro somma dovrebbe dunque corrispondere a -16. Vediamo se è vero:

$$\begin{array}{r} 01100001 \\ 10001111 \\ \hline 11110000 \end{array}$$

Questo valore è pari a $-128 + 64 + 32 + 16 = -16$.

ESERCIZIO 3 (5 punti)

1. Il grado delle tre relazioni è pari rispettivamente a 5, 3, 5, dato dal numero di attributi. Non è possibile fornire la cardinalità in quanto è dato solo lo schema delle relazioni.
- 2.

```
SELECT TITOLO
FROM ESAMI, CORSI, STUDENTI
WHERE STUDENTI.MATRICOLA=ESAMI.MATRICOLA
AND CORSI.COD-CORSO=ESAMI.COD-CORSO
GROUP BY TITOLO
HAVING COUNT(*) >=3
```

ESERCIZIO 4 (18 punti)

```
def leggi(nomeFile):
    idf=open(nomeFile,"r")
    linea=idf.readline()

    d={}
    while linea!="":
        valori=linea.split()
        k=valori[1]
        if k in d:
            d[k]=d[k]+[float(valori[0])]
        else:
            d[k]=[float(valori[0])]
        linea=idf.readline()

    idf.close()
    return d
```

```

def estrai_affidabili(lista, punteggio):
    naff=0
    for elemento in lista:
        naff=naff+int(elemento>punteggio)
    return naff

def estrai_soggetti(dizionario,punteggio):
    soggetti=[]
    naff=0
    n=0
    for k in dizionario:
        naff=naff+estrai_affidabili(dizionario[k],punteggio)
        n=n+len(dizionario[k])
    if naff:
        soggetti=soggetti+[k]
    return 100.*float(naff)/float(n), soggetti

def stampa_risultati(soggetti, p):
    print "Percentuale di riconoscimenti affidabili:",p
    print "Soggetti con almeno un riconoscimento affidabile:\n"
    for s in soggetti:
        print s

d=leggi("volti.txt")
p, soggetti=estrai_soggetti(d,0.5)
stampa_risultati(soggetti,p)

```