

**PROVA SCRITTA DEL CORSO DI
FONDAMENTI DI INFORMATICA
CORSI DI LAUREA IN INGEGNERIA CHIMICA ED INGEGNERIA MECCANICA
18 luglio 2018**

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

ESERCIZIO 1 (4 punti)

Si consideri il valore 57 in base 10.

- (1) (2 punti) Convertire il valore dato in base 2.
- (2) (2 punti) Moltiplicare per 2 il valore dato in base 10, convertire il risultato in binario e dedurre, dal confronto del valore testé convertito con quello convertito al punto 1, a cosa equivale la moltiplicazione per 2 quando effettuata su valori binari.

ESERCIZIO 2 (4 punti)

| | |
|---|---|
| <p>Relation Studenti (Matricola: char(5) PRIMARY KEY NOT NULL, Nome: char(20) NOT NULL, Data-N: date, Anno-Corso: integer, Corso-Laurea: char(1));</p> <p>Relation Corsi (Cod-Corso: char(6) PRIMARY KEY NOT NULL, Titolo: char(50), Docente: char(20));</p> <p>Relation Esami (Cod-Corso: char(5) NOT NULL, Matricola: char(5) NOT NULL, Data-E: date NOT NULL, Voto: integer NOT NULL, PRIMARY KEY (Cod-Corso, Matricola));</p> | <p>Facendo riferimento allo schema logico a sinistra, scrivere in SQL la seguente interrogazione, precisandone il tipo: "Estrarre nome e matricola degli studenti che hanno sostenuto il corso con codice pari a 1 prendendo un voto superiore a 25."</p> |
|---|---|

ESERCIZIO 3 (25 punti)

(7 punti) Nel laboratorio del Prof. Marcialis lo staff ha realizzato un robot in grado di stimare, tramite due videocamere poste in corrispondenza degli occhi, la forma delle persone in una scena, in modo tale da calcolare il *Body Mass Index* (BMI – Indice di Massa Corporea) e suggerire la dieta maggiormente rispondente al loro stato.

Durante la fase di osservazione, il robot scrive in un file "persone.txt" i dati riferiti a peso in kg ed altezza in m. Tuttavia, gli manca l'intelligenza per calcolare il BMI attraverso la formula: $BMI = peso / altezza^2$.

Come membri dello staff, di questo v'incaricate voi, predisponendo un programma Python che legga il file "persone.txt", calcoli il BMI e scriva su un file "dieta.txt", per ogni persona, la classificazione fornita dalla figura a lato. Più tardi il robot leggerà a ciascuna persona i dati memorizzati nel file creato in seguito all'elaborazione scritta da voi.

| BMI | CONDIZIONE |
|------------|------------------------|
| < 18,50 | SOTTOPESO |
| 18,50 - 25 | NORMOPESO |
| 25 - 30 | SOVRAPPESO |
| 30 - 40 | OBESITÀ DI MEDIO GRADO |
| > 40 | OBESITÀ DI ALTO GRADO |

Il file "persone.txt" è formattato, per esempio, come segue:

```
64.2 1.7  
80.0 1.5  
40.0 1.8
```

Dove ciascuna coppia di valori per riga rappresenta il peso e l'altezza di un soggetto osservato dal robot.

Nel file "dieta.txt" andrà invece scritto il relativo BMI calcolato secondo la formula di cui sopra e la classificazione associata, come dall'esempio:

```
22.2 Normopeso
35.6 Obesità di medio grado
12.3 Sottopeso
1 0
```

A fine file andrà scritta infine la coppia di valori corrispondenti al numero di soggetti sottopeso (primo valore) e affetti da obesità di alto grado (secondo valore).

Nell'implementare il programma, utilizzate una lista di liste per memorizzare le coppie peso-altezza per ogni individuo letto. Corrispondentemente, utilizzate un'altra lista di liste per memorizzare le coppie BMI-classificazione.

Infine, decidete di scrivere le seguenti funzioni:

(6 punti) Funzione `leggiDati(nomefile)`:

Legge da file di nome `nomefile` la lista dei valori di peso ed altezza associati dal robot ad ogni soggetto visto e restituisce una lista di liste di corrispondenti coppie di valori numerici letti per riga.

(8 punti) Funzione `calcolaBMI(peso, altezza)`:

Riceve in ingresso il peso e l'altezza **di un dato soggetto** e restituisce la coppia BMI-classificazione in base alla tabella fornita nel testo del problema in forma di lista `[BMI, classificazione]`.

(4 punti) Funzione `scriviRisultati(lista, ns, no)`:

Riceve in ingresso la lista di coppie BMI-classificazione e scrive su file "dieta.txt" le coppie di valori secondo il formato indicato nell'esempio, incluse la coppia di fine file relativa a numero di soggetti sottopeso (parametro in ingresso `ns`) e con obesità di alto grado (parametro in ingresso `no`).

NOTA: E' proibito alterare i requisiti e l'elenco dei parametri forniti dal testo. Si legga attentamente il testo dell'esercizio per non incorrere in tale errore.

Soluzioni

ESERCIZIO 1 (4 punti)

Con l'**algoritmo delle divisioni successive**, si ottiene subito la conversione $57_{10} \rightarrow 111001_2$.

Il doppio di 57 è 114. Riconvertendo si ottiene 1110010. Confrontando questo con il valore binario precedente si osserva che i due valori differiscono per la presenza di uno zero nella posizione meno significativa: $57_{10} \rightarrow 111001_2$ e $114_{10} \rightarrow 1110010_2$. In altre parole, la moltiplicazione per 2 di un valore binario corrisponde a far scorrere ciascuna cifra verso la posizione più significativa adiacente (per esempio la cifra più significativa 111001 dalla posizione 5 passa alla posizione 6 nel numero moltiplicato per 2), mentre viene assegnato il valore 0 alla cifra nella posizione 0. In aritmetica dei calcolatori, questa operazione si chiama "shift a sinistra": 111001 a 6 bit diventa $\rightarrow 1110010$ a 7 bit.

ESERCIZIO 2 (4 punti)

```
SELECT NOME, MATRICOLA
FROM STUDENTI, ESAMI
WHERE STUDENTI.MATRICOLA=ESAMI.MATRICOLA
AND COD-CORSO=1
AND VOTO>25
```

Si tratta senza dubbio di una join, perché richiede un sottoinsieme del prodotto cartesiano delle relazioni `Studenti` ed `Esami`.

ESERCIZIO 3 (25 punti)

```
def leggiDati(nomefile):
    f=open(nomefile,"r")
    l=[]
    r=f.readline()
    while r!="":
        p,a=r.split()
        l=l+[[float(p), float(a)]]
        r=f.readline()
    f.close()
    return l

def calcolaBMI(peso,altezza):

    bmi=peso/(altezza**2)

    if bmi<=18.5:
        return bmi, "Sottopeso"
    elif bmi<=25.:
        return bmi, "Normopeso"
    elif bmi<=30.:
        return bmi, "Sovrappeso"
    elif bmi<=40.:
        return bmi, "Obesita di medio grado"
    else:
        return bmi, "Obesita di alto grado"

def scriviRisultati(lista, ns, no):
    s=""
    for el in lista:
        s=s+str(round(el[0],1))+ " "+el[1]+"\\n"
    s=s+str(ns)+" "+str(no)+"\\n"

    f=open("dieta.txt","w")
    f.write(s)
    f.close()

#Programma principale
dati=leggiDati("persone.txt")

no=0
ns=0
outlista=[]
for d in dati:
    outd=calcolaBMI(d[0],d[1])
    outlista=outlista+[outd]
    ns=ns+int(outd[1]=="Sottopeso")
    no=no+int(outd[1]=="Obesita di alto grado")

scriviRisultati(outlista,ns,no)
```