

**PROVA SCRITTA DEL CORSO DI**  
**FONDAMENTI DI INFORMATICA 1**  
**CORSI DI LAUREA IN INGEGNERIA CHIMICA E INGEGNERIA MECCANICA**  
**4/7/2018**

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

**ESERCIZIO 1 (4 punti)**

E' data la rappresentazione per i valori interi in complemento a 2 con cinque bit. Indicare a quale valore decimale corrisponde la sequenza binaria 1011 e calcolare il suo opposto seguendo l'algoritmo di calcolo impartito a lezione.

**ESERCIZIO 2 (4 punti)**

Si esponga in modo chiaro e sintetico in cosa consiste l'architettura di un moderno calcolatore elettronico.

**ESERCIZIO 3 (25 punti)**

(3 punti) Questa volta gli studenti di Fondamenti di Informatica si sono organizzati a puntino per superare l'esame. Hanno collezionato i compiti degli ultimi anni accademici e hanno classificato le domande su Python fatte dal professore in tre categorie: "If:else", dove la soluzione imponesse l'uso dei costrutti decisionali, "Loop", dove la soluzione imponesse l'uso dei cicli, "I/O", dove la soluzione imponesse la lettura/scrittura di file. Gli studenti hanno inoltre monitorato i fogli degli esiti di ciascun esame ed hanno associato, per ogni scritto, il voto preso dagli esaminandi a quell'appello su quella determinata categoria. Queste due informazioni (categoria e voto) sono state inserite in un file formattato, per esempio, come segue:

```
I/O 5.0
I/O 3.0
Loop 6.0
If:else 4.0
If:else 5.0
```

Da queste informazioni gli studenti vogliono ricavare: (1) la o le categorie di costrutto Python richieste più frequentemente, (2) il voto medio complessivo per ciascuna delle categorie. Questi dati saranno scritti in file "output.txt" così formattato:

```
I/O If:else
I/O 4.0
Loop 6.0
If:else 4.5
```

Come si può notare il file di uscita presenta nella prima riga le due categorie più frequenti (If:else, I/O), nelle successive le categorie con il voto medio associato. Poiché il file originale è lungo oltre mille righe, gli studenti decidono di automatizzare il processo di estrazione di questi dati e chiedono a voi di scrivere il programma Python che calcoli le statistiche desiderate.

Nello scrivere la vostra soluzione, decidete che la cosa migliore sia conservare i dati letti da file in un dizionario le cui chiavi sono ciascuna delle categorie e i valori sono le liste di punteggi via via letti. Seguendo il file fornito da esempio si otterrebbe:  
`d={"I/O": [5.0, 3.0], "Loop": [6.0], "If:else": [4.0, 5.0]}`

Inoltre decidete di scrivere le seguenti funzioni:

(4 punti) `leggiDati(nomefile)`

- Riceve in ingresso il nome del file da leggere e produce in uscita il dizionario come indicato.

(8 punti) `calcolaStatistiche(d, cat)`

- Riceve in ingresso il dizionario con i dati sulle categorie, l'elenco delle categorie come lista `cat=["I/O", "Loop", "If:else"]` e produce in uscita due liste: nella prima ci sono le categorie più frequenti, nella seconda le medie associate a ciascuna categoria nell'ordine indicato da `cat`.

(4 punti) `calcolaMedia(l)`

- Riceve in ingresso una lista di valori numerici, calcola e restituisce la media aritmetica dei valori presenti nella lista.

(6 punti) `scriviRisultati(lf, lm, cat)`

- Riceve in ingresso la lista delle categorie più frequenti, la lista delle medie e la corrispondente lista di categorie associata (come nella funzione `calcolaStatistiche`) e le stampa su file `"output.txt"` secondo il formato fornito nell'esempio.

NOTA: E' proibito alterare i requisiti e l'elenco dei parametri forniti dal testo. Si guardino e comprendano gli esempi per non incorrere in tale errore.

### Soluzione es. 1

Il valore fornito è a quattro bit, quindi per rappresentarlo a cinque bit occorre aggiungere uno zero nella posizione più significativa ottenendo: 01011. Il valore fornito è sicuramente positivo proprio per via dello 0 in posizione più significativa e corrisponde, applicando l'algoritmo di conversione binario-decimale per il complemento a due, a:  $-0 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 0 + 8 + 0 + 2 + 1 = 11$ .

Per ottenere il suo opposto è sufficiente invertire tutti i bit della stringa completa, ottenendo 10100 ed aggiungere 1, da cui 10101. Facile verificare che questo valore soddisfa la definizione di complemento a 2 e può essere utilizzato per rappresentare l'opposto del valore numerico inizialmente dato.

### Soluzione es. 2

V. dispense del corso.

### Soluzione es. 3

```
def leggiDati(nomefile):
    d={"I/O":[], "Loop":[], "If:else": []}

    f=open(nomefile,"r")
    linea=f.readline()
    while linea!="":
        categoria, voto=linea.split() #ogni linea contiene due stringhe
        d[categoria]=d[categoria]+[float(voto)]
        linea=f.readline()
    f.close()

    return d

def calcolaStatistiche(d,cat):
    datif=0
    datim=[0.0,0.0,0.0] #lista con le medie
    cpf=[] #lista delle categorie più frequenti
    i=0
    for c in cat:
        f=len(d[c]) #questa parte di codice trova la/e
        if f>datif: #categoria/e più frequente/i
            cpf=[c]
            datif=f
        elif f==datif:
            cpf=cpf+[c]
        datim[i]=calcolaMedia(d[c]) #media per ogni categoria
        i=i+1

    return cpf, datim
```

```

def calcolaMedia(l):
    s=0.0
    for x in l:      #somma cumulativa dei valori in l
        s=s+x

    return s/float(len(l)) #divido per il loro numero

def scriviRisultati(lf, lm, cat):
    s="" #prima riga è la sequenza delle categorie più frequenti
    for c in lf:
        s=s+c+" "
    s=s+"\n"

    i=0
    while i<3: #le categorie sono solo tre (note a priori)
        s=s+cat[i]+" "+str(lm[i])+"\n" #stampa categoria e media
        i=i+1

    f=open("output.txt","w") #devo solo aprire il file e scrivere
    f.write(s)               #la stringa precedentemente generata
    f.close()

#Programma principale
cat=["I/O", "Loop", "If:else"] #cat è "costante"
d=leggiDati("input.txt")
lf, lm=calcolaStatistiche(d,cat)
scriviRisultati(lf,lm,cat)

```