

# COMPITO DI LINGUAGGI DI PROGRAMMAZIONE E LABORATORIO

**21 febbraio 2023 (Tot. 29 punti) Tempo: 3,5h**

## **Esercizio su Java e Python (13 punti per Java, 13 punti per Python)**

Si scriva un programma in Java e uno in Python per la gestione di uno studio medico.

I programmi devono leggere il file operatori.txt contenente l'elenco degli operatori dello studio con le seguenti informazioni (una per riga):

- codice (intero), uno spazio, tipo ("medico" o "terapista"), a capo,
- nome dell'operatore (eventualmente contenente spazi), a capo,
- costo della visita (double), a capo,
- nel caso di medico: specialità (stringa eventualmente contenente spazi), a capo, n. visite nel mese precedente (intero), uno spazio, ambulatorio (stringa priva di spazi), a capo,
- nel caso di terapista: terapia (stringa eventualmente contenente spazi), a capo, studio (stringa priva di spazi), uno spazio, n. sedute della terapia (intero), a capo.

I programmi devono poi leggere un secondo file pazienti.txt contenente le informazioni sui pazienti:

- codice del paziente (intero), a capo,
- nome del paziente (stringa eventualmente contenente spazi), a capo,
- visite allo studio nella forma di un elenco:
  - codice dell'operatore (intero), uno spazio, numero di sedute (intero), uno spazio.

1. I programmi devono leggere il seguente file operatori.txt:

```
1 medico
Giovanni Verdi
75.0
otorinolaringoiatra
10 a10
2 terapista
Alberto Pifferi
62.0
agopuntura
b15 5
3 medico
90.0
Stefania Giovannini
gastroenterologo
20 a12
4 terapista
Monica Freddi
76.0
riabilitazione
b4 4
5 medico
Roberta Rossi
100.0
Internista
30 a2
```

e memorizzare gli operatori.

2. I programmi devono leggere il seguente file pazienti.txt:

```
1
Roberto Ferilli
1 3 5 4 2 7
2
```

Andrea Guerra  
3 2 4 3  
3  
Stefano Fantoni  
5 4 3 2 1 5  
4  
Valeria Verdi  
4 3 2 5 3 6

e memorizzare i pazienti.

1. I programmi devono stampare a video l'elenco di tutti gli operatori in una tabella con queste intestazioni:  
nome operatore, codice, costo, specialità, visite,  
ambulatorio, terapia, studio, sedute, tipo  
Per gli attributi che non si applicano ad un operatore (terapia, studio, sedute per i medici e specialità, visite, ambulatorio per i terapisti) si stampi "-".  
Per il tipo si stampi medico o terapeuta (punti 4 per Java, punti 4 per Python).
2. I programmi devono stampare a video l'elenco dei pazienti in una tabella con questa intestazione  
codice, nome, numero operatori, visite  
dove numero operatori è il numero di operatori presso cui hanno fatto visite, e  
visite è l'elenco delle visite nella forma di una lista di coppie (codice, numero)  
(punti 4 per Java, punti 4 per Python).
3. I programmi devono leggere da riga di comando il codice di un paziente e stampare il nome del paziente e il numero totale di sedute da lui sostenute (somma del numero di sedute con ogni operatore) (punti 5 per Java, punti 5 per Python).

## I programmi devono stampare qualcosa di simile a

```
java Gestione 2
nome operatore, codice, costo, specialità, visite, ambulatorio, terapia, studio, sedute,
tipo
nome operatore, codice, costo, specialità, visite, ambulatorio, terapia, studio, sedute,
tipo
Giovanni Verdi      1      75.0  otorinolaringoiatra 10      a10      -      -      -
      medico
Alberto Pifferi     2      62.0  -      -      -      agopuntura      b15      5      terapeuta
Stefania Giovannini 3      90.0  gastroenterologo      20      a12      -      -      -
      medico
Monica Freddi4      76.0  -      -      -      riabilitazione      b4      4      terapeuta
Roberta Rossi5      100.0 Internista      30      a2      -      -      -      medico
codice, nome, numero operatori
1      Roberto Ferilli      3      [(1,3), (5,4), (2,7)]
2      Andrea Guerra2      [(3,2), (4,3)]
3      Stefano Fantoni      3      [(5,4), (3,2), (1,5)]
4      Valeria Verdi3      [(4,3), (2,5), (3,6)]
```

Numero di visite del paziente Andrea Guerra: 5

```
python gestione.py 3
nome operatore, codice, costo, specialità, visite, ambulatorio, terapia, studio, sedute,
tipo
Giovanni Verdi      1      75.0  otorinolaringoiatra 10      a10      -      -      -
      medico
Alberto Pifferi     2      62.0  -      -      -      agopuntura      b15      5      calcio
Stefania Giovannini 3      90.0  gastroenterologo      20      a12      -      -      -
      medico
Monica Freddi4      76.0  -      -      -      riabilitazione      b4      4      calcio
Roberta Rossi5      100.0 Internista      30      a2      -      -      -      medico
codice, nome, visite
1      Roberto Ferilli      3      [(1,3) (5,4) (2,7)]
2      Andrea Guerra2      [(3,2) (4,3)]
3      Stefano Fantoni      3      [(5,4) (3,2) (1,5)]
4      Valeria Verdi3      [(4,3) (2,5) (3,6)]
```

Numero sedute del paziente Stefano Fantoni: 11

I programmi devono sfruttare incapsulamento e astrazione al massimo grado.

I programmi devono avere una interfaccia testuale che usi la console.

Se il codice non si compila il voto sarà insufficiente.

Si può accedere alla documentazione su Java a

<https://ml.unife.it/java11-api>

## **Per chi ha seguito nell'AA 2022/23**

### **Esercizio su programmazione funzionale (punti 3)**

Scrivere una funzione Haskell che calcola la somma dei primi n numeri interi

`sumN :: Int -> Int`

Ad esempio

`Prelude> sumN 5`

`15`

## **Per chi ha seguito nell'AA 2021/22 o precedenti**

### **Esercizio su programmazione logica (punti 3)**

Dato il programma

```
pl([], Acc, Acc) .
```

```
pl([H|T], Acc0, Acc) :-  
    Acc1 is Acc0*H,  
    pl(T, Acc1, Acc) .
```

Indicare tutte le risposte alle query

```
pl([3,4], 1, S) .
```

```
pl([1,2,3,4], 1, S) .
```

```
pl([1,2], 1, S) .
```