

Corso di Sistemi Operativi e Reti

Prova scritta di GENNAIO 2019

**ISTRUZIONI**

1. **Rinomina** la cartella chiamata "CognomeNomeMatricola" che hai trovato sul Desktop e in cui hai trovato questa traccia, sostituendo "Cognome" "Nome" e "Matricola" con i tuoi dati personali;
2. **Carica** tutto il materiale didattico che vorrai usare sul Desktop; puoi farlo solo nei primi 5 minuti della prova;
3. **Svolgi** il compito; lascia tutto il sorgente che hai prodotto nella cartella di cui al punto 1;
4. Quando hai finito lascia la postazione facendo logout,

**senza spegnere il PC.**

**SALVA SPESSO il tuo lavoro**

## ESERCIZIO 1 (Programmazione multithread. Punti: 0-20)

Si implementino le classi `Cittadino` e `Popolo` secondo lo scheletro di codice sorgente fornito. Lo scheletro è fornito rispettivamente nel file `cittadino.py` per chi lavorerà in Python e `Cittadino.java` per chi lavorerà in Java.

Tutti i metodi richiesti per entrambe le classi devono essere implementati garantendo la necessaria thread safety, e dunque assumendo che più di un thread possa invocare tali metodi contemporaneamente; non sono ammesse situazioni di deadlock; è opportuno migliorare l'accessibilità concorrente alle strutture dati ed evitare potenziali situazioni di starvation.

# NON SPEGNERE IL PC A FINE ESAME

## CI SONO DEI PUNTI AMBIGUI NELLA TRACCIA? **COMPLETA TU**

È parte integrante di questo esercizio completare le specifiche date nei punti non esplicitamente definiti, introducendo o estendendo tutte le strutture dati laddove si ritenga necessario, e risolvendo eventuali ambiguità.

## POSSO CAMBIARE IL PROTOTIPO DEI METODI RICHIESTI? **NO**

*Non è consentito modificare il prototipo dei metodi se questo è stato fornito. Potete aggiungere qualsivoglia campo e metodo di servizio, e qualsivoglia classe ausiliaria, ma NON variare l'interfaccia dei metodi pubblici già specificati.*

## CHE LINGUAGGIO POSSO USARE? **PYTHON 3.X; oppure JAVA 7 o successivo**

Il linguaggio da utilizzare per l'implementazione è Python 3, o in alternativa, Java. È consentito usare qualsiasi funzione di libreria di Python 3.X o di Java 7 e versioni successive.

## MA IL MAIN() LO DEVO SCRIVERE? E I THREAD DI PROVA? **SI**

Sebbene non saranno oggetto di valutazione, è obbligatorio scrivere un `main()` e implementare esplicitamente del codice di prova, che è comunque necessario per testare il proprio codice prima della consegna.

**NON SPEGNERE IL PC A FINE ESAME**

# NON SPEGNERE IL PC A FINE ESAME

## ESERCIZIO 2 (Linguaggi di scripting. Punti 0-10)

Si scriva uno script Perl dal nome `rubrica.pl`. Lo script dovrà modellare l'utilizzo e la gestione dei contatti salvati in rubrica dall'utente. In particolare lo script metterà a disposizione 4 diverse funzionalità, ovvero l'aggiunta, la rimozione, la ricerca e l'esportazione dei contatti in rubrica.

In particolare il codice da progettare dovrà funzionare nel seguente modo:

una volta avviato lo script (`./rubrica.pl` o `perl rubrica.pl`) all'utente sarà richiesto di effettuare una delle operazioni messe a disposizione dallo script (aggiunta --> opzione `-a`, rimozione --> opzione `-d`, ricerca --> opzione `-s` o esportazione --> opzione `-e`) tramite l'immissione da tastiera del parametro opportuno. L'utente effettuerà quindi l'operazione e lo script continuerà a girare finché l'utente non deciderà di fermarlo (kill --> opzione `-k`)

**N.B. Non sono accettati argomenti iniziali su linea di comando**

**N.B. Per semplicità si assume che il nome e il cognome dei contatti non contengano il carattere ",".**

I metodi richiesti sono descritti nel dettaglio di seguito.

**##### Opzione 1 -a #####**

Aggiunge contatti alla rubrica.

Viene richiesto di inserire la stringa "Nome, Cognome, Numero\_di\_telefono".

Se il contatto esiste già in rubrica allora questo va aggiornato con il nuovo numero di telefono, altrimenti si deve creare una nuova voce.

Il numero di telefono deve essere controllato: non possono essere presenti caratteri o simboli e deve essere composto esattamente da 10 cifre.

**##### Opzione 2 -d #####**

Rimuove un contatto dalla rubrica.

Viene richiesto di inserire la stringa "Nome, Cognome"

Il contatto dovrà essere cancellato dalla rubrica. Se il contatto non esiste è necessario emettere un messaggio di errore.

**NON SPEGNERE IL PC A FINE ESAME**

# NON SPEGNERE IL PC A FINE ESAME

##### Opzione 3 -s #####

Ricerca un contatto in rubrica.

Chiedere di inserire la stringa "Nome" o "Nome, Cognome" e mostrare in output tutti i contatti corrispondenti nel formato:

Nome, Cognome, Numero\_di\_telefono

##### Opzione 4 -e #####

Esporta la lista dei contatti su un file dal nome "contatti.vcf" ordinati alfabeticamente.

Il formato del file dovrà essere il seguente:

```
BEGIN:VCARD
VERSION:2.1
FN:Nome Cognome
N:Cognome;Nome;;;
TEL;CELL: Numero_di_telefono
END:VCARD
```

##### Opzione 5 -k #####

Termina lo script.

NON SPEGNERE IL PC A FINE ESAME