

Studente: , Matricola: , Email:

Compito: Turno:

PRINCIPI DI SISTEMI OPERATIVI

A.A. 2015/16

09 Giugno 2016

IMPORTANTE:

- Si considerano parte integrante delle soluzioni i COMMENTI significativi introdotti per facilitare la lettura del codice: come tali, essi influenzano la votazione finale. Tuttavia, i messaggi di debug (ad es. le println()) del programma NON SONO CONSIDERATI E QUINDI NON INFLUENZANO LA VOTAZIONE FINALE.
- Il tempo a disposizione è di 2 ore.
- Il compito deve essere svolto in Java, lavorando con l'ambiente di sviluppo Eclipse, usando le classi del package monitor o i costrutti di sincronizzazione nativi di Java.
- Seguire le seguenti regole per lo svolgimento dell'esame::
 - a. Fare il login in Linux
 - b. Aprire un terminale e digitare:
 - i. `$ wget ftp://lica02.lab.unimo.it/README` e seguire le istruzioni contenute. Lo script scaricato e lanciato, dopo aver risposto alle varie domande, crea una cartella sul Desktop chiamata **studente_XXXX**.
 - c. Aprire Eclipse (comando "eclipse" sempre da shell) ed usare come workspace la cartella **studente_XXXX** appena creata. ATTENZIONE: Saltando questo passaggio non riceveremo il vostro codice e non sarà quindi possibile correggere il compito.
 - d. Creare un progetto Java all'interno del workspace e nominarlo **nome_cognome_numTessera**.
 - e. Finito l'esame: salvare, attendere un ultimo upload, chiudere Eclipse, fare il logout, lasciare il vostro PC e procedere alla consegna del testo.

Studente: , **Matricola:** , **Email:**

Compito: **Turno:**

TESTO:

Il comune della città di Metropolis dispone di più uffici (**N**) per poter svolgere diverse tipologie di pratiche. I cittadini della città arrivano all'anagrafe e si prenotano ad uno degli **N** uffici per svolgere le proprie pratiche. Una volta entrati nell'ufficio i cittadini impiegano un certo tempo (random) per completare la pratica, ed in seguito lasciano l'ufficio. Il comune ha diviso i cittadini in 2 categorie: **GIOVANI** e **ANZIANI**. I cittadini **ANZIANI** hanno priorità sui cittadini **GIOVANI** nell'essere serviti agli uffici. Talvolta, però (si realizzi un meccanismo che selezioni in modo random), un cittadino che si presenta in un ufficio non può tornare direttamente a casa, ma viene invece rimandato ad un altro ufficio (scelto ancora in modo random). In questo caso, indipendentemente dalla categoria degli altri cittadini in attesa (**GIOVANE** o **ANZIANO**), il cittadino ha priorità su tutti gli altri.

Si implementi una soluzione modellando i cittadini come Thread e gli uffici come risorse, assicurandosi che la gestione delle code sia sempre FIFO e garantendo le priorità indicate. Nella soluzione si massimizzi l'utilizzo delle risorse. Si discuta se la soluzione proposta può presentare starvation e, in caso positivo, si indichi per quali processi e si proponga una possibile soluzione per evitare la starvation (nello spazio bianco qui sotto).

*"Believe in your potential even
if you haven't seen the results"*
Gaby Natale