

Studente: {{Nome}} {{Cognome}}, Matr. {{Matricola}}, Email: {{Email}}

Compito: {{Num}} **Turno:** {{Dsati}}

PRINCIPI DI SISTEMI OPERATIVI

A.A. 2015/16

18 Dicembre 2015

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 con il proprio account (numero e password usate per la e-mail)
 - 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) e selezionare la cartella **studente_XXXX**, appena creata, come workspace.
 - d. Creare un progetto Java all'interno del workspace.
 - e. Finito l'esame: salvare, chiudere Eclipse, fare il logout, lasciare il vostro PC e procedere alla consegna del testo.

TESTO:

Nella città di Los Angeles è presente un **quartiere fieristico** in cui è in corso un'esposizione di cimeli di Star Wars in cui sono presenti **N_PADIGLIONI** (>2) padiglioni che hanno attirato tantissimi visitatori.

Per garantirne la sicurezza, ogni padiglione ha un numero massimo **MAX_VISITATORI** di visitatori che può contenere in un determinato momento: quelli che in un certo istante non possono entrare devono dunque mettersi in coda.

I biglietti acquistabili dai visitatori per l'accesso al quartiere fieristico sono di due tipi: **semplice** o **premium**. I visitatori con biglietto premium hanno priorità nelle code all'ingresso dei padiglioni rispetto a quelli con biglietto semplice.

Al fine di evitare code troppo lunghe, la direzione del quartiere fieristico ha inoltre stabilito che ogni visitatore può visitare un numero massimo **MAX_PADIGLIONI** di padiglioni ($MAX_PADIGLIONI \ll N_PADIGLIONI$). Ciascun visitatore stabilisce inizialmente quanti (**X**) e quali padiglioni visitare (con X da 1 a **MAX_PADIGLIONI** padiglioni): la visita di ogni padiglione ha una certa durata di tempo (scelta in modo random); una volta terminata la visita degli X padiglioni, il visitatore lascia il quartiere fieristico.

Si implementi una soluzione per modellare o il **quartiere fieristico**, con gli **N_PADIGLIONI** padiglioni come risorse oppure direttamente le risorse **N_PADIGLIONI** padiglioni, e i processi **visitatori**, assicurandosi che la gestione delle code sia sempre FIFO e garantendo la priorità indicata. Nella soluzione si massimizzi l'utilizzo delle risorse. Si discuta se la soluzione proposta può presentare starvation e in caso positivo per quali processi, e si propongano modifiche e/o aggiunte per evitare la starvation (nello spazio bianco qui sotto).

May the force be with you