

Nome: «Nome» Cognome: «Cognome» Compito: «Numero» Turno: «Turno»

# PRINCIPI DI SISTEMI OPERATIVI

## (A.A. 14-15)

### 09 Settembre 2015

#### IMPORTANTE:

1. 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**.
2. Il tempo a disposizione è di 90 minuti.
3. Il compito deve essere svolto solamente nel linguaggio Java, usando le classi del package **monitor** e lavorando con l'ambiente di sviluppo **IBM Eclipse**.
4. Seguire le seguenti regole per lo svolgimento dell'esame al laboratorio LINFA:
  - Fare il login in Linux con il proprio account (numero di tesserino e password di posta elettronica)
  - Aprire un browser sulla pagina `ftp://lica02.lab.unimo.it/README` e copiare il comando presente in un terminale: sul Desktop, viene creata automaticamente una directory `studente_XXX` al cui interno viene creato un file denominato `student_data.csv` che non va eliminato.
  - Aprire Eclipse (comando "eclipse" sempre da shell)
  - Utilizzare come workspace la cartella "studente\_xxxxx"
  - Creare un progetto Java con nome "ESAME090915-«Turno»-«Numero»" e scrivere le classi Java della soluzione nel package di default (senza nome) di tale progetto. Fare attenzione a scrivere correttamente il nome del progetto, con maiuscole e minuscole a posto!
  - Installare le classi del monitor Java e gli eventuali template
  - Finito il vostro esame (o allo scadere del tempo), dovete salvare tutto (si consiglia di salvare spesso per non perdere il proprio lavoro), chiudere Eclipse, fare il logout, lasciare il vostro PC e procedere alla consegna del testo.

In un **Centro per Eventi**, sono presenti **S stanze** e **O organizzatori** (con  $O \ll S$ ) all'apertura/chiusura delle stanze. Al centro si possono fare diversi **eventi**: cerimonie (battesimi, comunioni, etc..) e compleanni, ognuno dei quali deve far riferimento ad un **festeggiato**. Per poter fare la propria festa, il festeggiato deve prenotare una stanza libera. Una volta che la stanza prenotata è libera, il festeggiato deve attendere un organizzatore per l'allestimento della sala (tempo random). A questo punto la festa può avere luogo (sempre per una durata random dipendente da ogni festa e decisa dal festeggiato). Una volta finita la festa, la stanza deve essere riordinata e per questo è necessario che il festeggiato attenda un organizzatore che sistemi la stanza e la chiuda. Solo dopo la sua chiusura, la stanza potrà essere utilizzata per un'altra festa. Nella prenotazione e assegnazione delle stanze, le cerimonie hanno priorità rispetto ai compleanni.

Si implementi una soluzione usando il costrutto monitor per modellare il **Centro per eventi**, i processi per modellare i **festeggiati** e gli **organizzatori**, si utilizzino le stanze come risorse. 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.