

Principi di Sistemi Operativi – Esercitazione 4

1-Biblioteca (Esame 16/06/2000)

Una piccola **biblioteca** ha a disposizione **L** libri e **P** ($< L$) posti a sedere per la consultazione. Ogni **utente** richiede 3 libri, che consulta contemporaneamente; gli utenti prendono posto a sedere solo se sono disponibili tutti i libri che hanno richiesto e c'è posto a sedere. Terminata la consultazione, ogni utente deve restituire tutti i libri che ha avuto in prestito.

Si implementi una soluzione usando il costrutto monitor per modellare la **biblioteca** e i processi per modellare gli **utenti**, e si descriva la sincronizzazione tra gli utenti. 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 starvation.

2-Terremoto (Esame 20/06/2012)

In una grande **Città** si avverte una potente scossa di terremoto. Gli **N cittadini** che abitano in quella città, una volta passata la paura, contattano o i **VVFF** o la **Protezione Civile** a seconda della classificazione della zona in cui abitano (zona rossa, gialla o verde scelta in maniera random alla creazione del processo).

I cittadini che abitano nella zona rossa contattano i VVFF che devono mandare una delle loro **S** squadre ($S \ll N$) a verificare l'agilità dell'abitazione. La prima squadra dei VVFF libera, si reca da un cittadino, prendendo la sua chiamata in ordine di arrivo. Se la casa viene dichiarata dai VVFF "inagibile" (per simulare tale decisione si usi una scelta random), il cittadino deve chiedere aiuto alla Protezione Civile (v. dopo), mentre se viene dichiarata "agibile" il cittadino può tornarsene a casa.

I cittadini che abitano nella zona gialla contattano la Protezione Civile, per ottenere un controllo strutturale dell'abitazione, mentre i cittadini che abitano nella zona rossa la cui casa è stata dichiarata "inagibile" chiedono, sempre alla Protezione Civile, ospitalità nelle tendopoli.

La Protezione Civile ha **C** squadre ($C < N$). Ogni squadra può effettuare una verifica strutturale per i cittadini della zona gialla (verifica che dura un tempo t scelto random), oppure può aprire le pratiche per l'ospitalità nella tendopoli per i cittadini della zona rossa (tempo per le pratiche z , scelto ancora in modo random). La Protezione Civile dà la priorità ai cittadini che abitano nella zona rossa.

I cittadini che invece abitano nella zona verde, devono attendere che sia finita l'emergenza (ovvero che non ci siano cittadini della zona rossa che chiedono l'intervento dei VVFF), per chiedere ai VVFF di aprire una pratica per verificare eventuali crepe (tempo per l'apertura della pratica scelto random).

Si implementi una soluzione usando il costrutto monitor per modellare la **Città**, i processi per modellare i **cittadini**, i **VVFF** e la **Protezione Civile**. 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.