

Principi di Sistemi Operativi – Esercitazione 6

1-Ambasciata (Esame 15/02/2012)

In un' **Ambasciata** lavorano N **addetti** ed un **ambasciatore**. In ambasciata si recano le persone che abitano fuori dal loro paese d'origine e devono richiedere un visto.

Una volta arrivate in ambasciata, le persone devono recarsi da uno degli addetti (il primo libero) per verificare la correttezza dei documenti portati (tempo random deciso dall'addetto). Finita la verifica con l'addetto, devono recarsi dall'ambasciatore per richiedere la vidimazione del visto.

L'ambasciatore, una volta ricevuta la persona, controlla nuovamente la richiesta (tempo random deciso dall'ambasciatore) e, se è corretta, firma il visto e la persona può tornare a casa¹. Se invece l'ambasciatore trova anomalie nella richiesta rimanda la persona da un addetto per ottenere i timbri mancanti prima di poter tornare da lui e richiedere nuovamente la firma del visto. Per l'accesso agli addetti, le persone che devono ottenere i timbri dagli addetti hanno la priorità rispetto a quelle che vengono dall'esterno.

Si implementi una soluzione usando il costrutto monitor per modellare l'**Ambasciata**, e i processi per modellare gli **addetti**, l'**ambasciatore** e le **persone**. 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.

2-Centro agopuntura (Esame 18/01/2012)

In un **Centro per l'Agopuntura** lavorano D **dottori** (con D multiplo di 2). Nel Centro si recano i **pazienti** (bambini o adulti) per fare una seduta di agopuntura.

Una volta trovato un dottore in grado di iniziare una seduta (si veda nel seguito le specifiche condizioni), il paziente attende il termine del trattamento per un tempo random (scelto dal paziente stesso). Nelle sedute di agopuntura, i bambini hanno la priorità rispetto agli adulti. Una volta finita la seduta, il paziente libera il dottore e torna a casa.

Dei D dottori che lavorano nel centro, $D/2$ sono primari e $D/2$ dottori non primari. I primari sono in grado di effettuare una seduta:

- a) se non hanno ancora effettuato N sedute di agopuntura;
- b) hanno effettuato N sedute di agopuntura e si sono fermati per un tempo X (noto) per sbrigare le pratiche burocratiche.

Quindi se un paziente trova un dottore primario libero che ha già effettuato N visite, deve anche verificare che abbia finito le pratiche per poter iniziare la seduta²; se un paziente non trova nessun dottore in grado di iniziare una seduta, il paziente deve attendere un dottore.

Si implementi una soluzione usando il costrutto monitor per modellare il **Centro per l'Agopuntura**, i processi per modellare i **pazienti** (adulti e bambini) e si utilizzino i **dottori** 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.

¹ Si modelli la scelta da parte dell'ambasciatore se firmare o no il visto, come una scelta random da parte dell'ambasciatore stesso.

² Suggerimento: Si memorizzi l'informazione relativa al tempo in cui un paziente libera un primario, ricavata usando il metodo `getTime()` della classe `java.util.Date`.

