

VERIFICA DI SISTEMI E RETI
CLASSE 4°C – A.S. 2019/20 – ITIS RIVOIRA

Mulino Convention Centre è il centro congressi di Verzuolo gestito dall'azienda "Amici del Mulino", leader nella gestione di spazi congressuali ed eventi sul territorio.

Il MulinoCC è sede ogni anno dei più importanti congressi ed eventi internazionali dedicati alle nuove tecnologie emergenti nel campo dell'informatica ed è formato da 5 sale interne. Ognuna delle 5 sale interne è numerata da 1 a 5. Ogni giorno, il MulinoCC ospita 5 eventi differenti che si svolgono in contemporanea, ciascuno in una delle 5 sale interne.

Il personale dell'azienda Amici del Mulino, si occupa dell'iscrizione dei partecipanti a ciascuna giornata.

Ogni partecipante, una volta raggiunto il MulinoCC dovrà registrarsi ad uno degli eventi, ricevere il badge di ingresso per l'evento a cui si è registrato ed al termine dell'evento uscire dal centro congressi.

Il MulinoCC, mantiene giornalmente un registro per ogni sala interna in cui memorizza i dati delle persone che partecipano a ciascun evento.

Implementare, utilizzando il linguaggio C e le tecniche viste a lezione di gestione dei processi, memoria condivisa e semafori, una soluzione per simulare una giornata di congressi al MulinoCC.

Si richiede di:

1. Implementare la soluzione su due file distinti **MulinoCC.c** e **AmiciDelMulino.c**
2. MulinoCC dovrà gestire le 5 sale interne e la fornitura dei badge. I badge saranno messi a disposizione in numero limitato di 5 unità per ogni sala; una volta terminati i badge, MulinoCC provvederà a reincrementare il numero di badge per quella determinata sala. Per semplicità supponiamo che giornalmente si svolga un solo evento per ogni sala interna. Inoltre, per ogni sala interna, viene mantenuto un registro giornaliero (ad esempio sala1.txt, sala2.txt, ecc...) che contiene il nome dell'evento (es. "evento1", "evento2", ecc...) e le credenziali fornite in fase di registrazione da ogni partecipante all'evento. Per quanto riguarda le credenziali, scegliere le informazioni ritenute più opportune (ad. Esempio PID, nome, cognome, ecc...). Al termine della giornata, quando tutti i partecipanti avranno abbandonato le sale, il MulinoCC verrà chiuso.
3. AmiciDelMulino dovrà simulare l'iscrizione dei partecipanti (max 50 unità) agli eventi giornalieri e la distribuzione dei badge per ogni sala.
4. Ogni partecipante, una volta raggiunto il MulinoCC, dovrà scegliere uno degli eventi a cui partecipare, al quale potrà accedere solo dopo aver ottenuto il badge. Ciascun partecipante dovrà, infine, uscire dalla sala ed abbandonare il MulinoCC.

GESTIONE FILE DI TESTO

```
FILE *fp

fp=fopen("esempio.txt","w|a")
fprintf(fp,"Scrivo nel file\n")
fclose(fp)
```

FUNZIONI DI UTILITY PER L'USO DELLA SHARED MEMORY

```
IDMEMORIA = shmget(KEY, SIZE, PERMESSI)
PUNTATOREMEMORIA = (TIPO*) shmat(IDMEMORIA, NULL, 0)
VAR=shmctl(IDMEMORIA, IPC_RMID, NULL)
```

FUNZIONI DI UTILITY PER L'USO DEI SEMAFORI

```
IDSEMAFORO = semget(KEY, SIZE, PERMESSI)
VAR=semctl(IDSEMAFORO, NSEMAFORO, SETVAL, 1)
VAR=semctl(IDSEMAFORO, NSEMAFORO, GETVAL)
VAR=semctl(IDSEMAFORO, NSEMAFORO, IPC_RMID, 0)

void sem_Lock (int sem_id, int id){
    buffer.sem_num=id;
    buffer.sem_flg=0;
    buffer.sem_op=-1;
    if(semop(sem_id,&buffer,1)==-1)
        printf("Errore blocco semaforo\n");
}

void sem_Unlock (int sem_id, int id){
    buffer.sem_num=id;
    buffer.sem_flg=0;
    buffer.sem_op=1;
    if(semop(sem_id, &buffer, 1)==-1)
        printf("Errore sblocco semaforo\n");
}
```