

Sistemi Operativi L-A
Compito di Mercoledì 19 Dicembre 2007
CdS in Ingegneria Informatica - Prof. Paolo Bellavista

Compito A - Parte di Programmazione di Sistema (16 punti)

Si scriva un programma C che, utilizzando le System Call del sistema operativo UNIX, abbia un'interfaccia del tipo:

`giocoDiCarte mosse vincitore`

dove **mosse** è un nome assoluto di file esistente nel file system mentre **vincitore** è il nome relativo del file che deve essere creato dal programma concorrente nel direttorio corrente. Dopo aver effettuato gli opportuni controlli sui parametri di invocazione, il processo iniziale **P0** (coordinatore del gioco di carte) deve generare altri 2 processi giocatori **P1** (Player1) e **P2** (Player2), rispettivamente figlio e nipote di **P0**.

Il file **mosse** contiene 39 righe che rappresentano, ciascuna una carta da gioco secondo il seguente formato: <numCarta (da 1 a 10), seme (danari, coppe, ...)>. Le prime 13 righe costituiscono le 13 carte che, turno dopo turno, saranno giocate da **P0**; il secondo slot di 13 righe rappresenta le 13 mosse che saranno giocate da **P1**; infine le ultime 13 righe costituiscono le 13 giocate di **P2**. Si scelga opportunamente la modalità di rappresentazione delle righe per rendere più semplice la lettura del file **mosse** da parte dei processi.

Ad ogni turno, ciascuno dei tre processi gioca una carta (secondo lo schema indicato sopra) e si aggiudica il turno il giocatore che ha giocato la carta con NumCarta più alto, indipendentemente dal seme. Si preveda che **P1** e **P2** comunichino a **P0** la carta giocata (sia numero che seme, come rappresentata nel file); non è invece necessario che **P0** comunichi agli altri la sua carta. Ricevute le informazioni sulle carte giocate, **P0** aggiudica il turno a uno dei processi e gli comunica chi "ha preso" in quel turno. Ogni processo autonomamente mantiene un proprio contatore con il numero delle prese effettuate. Al termine delle carte (dopo 13 turni), il processo che ha effettuato il maggior numero di prese deve scrivere sul file **vincitore** "Sono il processo PIDx e ho vinto", mentre gli altri due processi dovranno scrivere sullo stesso file "Sono il processo PIDy e sono stato battuto"; non ci sono obblighi sull'ordine con cui le righe sono scritte su **vincitore**.

In ogni istante, deve essere possibile per l'utente forzare l'uscita dal gioco di **P1** premendo la combinazione di tasti <CTRL-C> o di **P2** tramite la combinazione di tasti <CTRL-Z>. In entrambi i casi la partita deve essere considerata terminata, avrà vinto il processo che in quel momento ha il numero più alto di prese, e i tre processi devono scrivere su **vincitore** l'esito della partita come nel caso normale di terminazione di tutte le mosse.

Si facciano le scelte di sincronizzazione dei processi ritenute più opportune, cercando di sequenzializzare il meno possibile le varie operazioni richieste.