

Sistemi Operativi L-A
Compito di Martedì 16 Settembre 2008
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:

unicaPipeA fileIn fileOut

dove **fileIn** è un nome assoluto di file esistente nel file system e **fileOut** è il nome assoluto del file che deve essere creato come risultato dell'esecuzione del programma concorrente. Dopo aver effettuato gli opportuni controlli sui parametri di invocazione, il processo iniziale **P0** deve generare un processo figlio **P1** che a sua volta genererà un processo nipote **P2**.

fileIn contiene una sequenza di caratteri organizzati in righe (numero di righe non noto a priori, lunghezza massima di una riga pari a 80 caratteri), con ciascuna riga che termina con il carattere '+' oppure con il carattere '-'. I processi figlio e nipote devono leggere concorrentemente l'intero **fileIn**, **P1** processando le sole righe che terminano con '+', **P2** occupandosi delle righe che terminano per '-'. In particolare, per ciascuna riga di pertinenza, **P1** e **P2** dovranno inviare un messaggio a **P0** contenente i soli caratteri maiuscoli trovati nella riga. I processi figlio devono comunicare con il padre facendo uso di una unica pipe condivisa, utilizzata quindi sia da **P1** che da **P2** per scrivere messaggi a **P0** (due processi scrittori e uno lettore). **P0** deve leggere i messaggi dalla pipe, tenere traccia del numero totale di caratteri maiuscoli trovati dai processi e scrivere il contenuto dei messaggi su **fileOut**.

Una volta terminata la lettura del file e la ricezione di messaggi da parte di **P0**, **P0** deve comunicare l'esito del conteggio sia a **P1** che a **P2** (l'unica pipe condivisa è utilizzata questa volta in verso opposto) e terminare immediatamente il processo con conteggio più basso. L'altro processo dovrà invece scrivere su **fileOut** "Sono il processo `pid` e ho realizzato il conteggio `x` più alto".

In ogni istante, deve essere inoltre possibile per l'utente forzare la modifica del comportamento del programma concorrente premendo la combinazione di tasti <CTRL-C>. In particolare, se <CTRL-C> viene premuto due volte in meno di 1 secondo, allora **P2** dovrà essere terminato e **P1** dovrà occuparsi di tutte le righe rimanenti del file, anche quelle che terminano con '-'; ruoli scambiati fra i due processi, invece, nel caso di singola pressione di <CTRL-C>.

Si facciano le ipotesi semplificative desiderate in termini di modello affidabile dei segnali e di innestamento dell'esecuzione dei gestori associati. Come sempre, si facciano le scelte di sincronizzazione dei processi ritenute più opportune, cercando di sequenzializzare il meno possibile le varie operazioni richieste.