

SISTEMI OPERATIVI e LABORATORIO DI SISTEMI OPERATIVI (A.A. 07-08) – 4 APRILE 2008

IMPORTANTE:

- 1) Fare il login sui sistemi in modalità Linux usando il proprio **username** e **password**.
- 2) I file prodotti devono essere collocati in un **sottodirettorio** della propria **HOME** directory che deve essere creato e avere nome **ESAME04Apr08-1-1**. FARE ATTENZIONE AL NOME DEL DIRETTORIO, in particolare alle maiuscole e ai trattini indicati. Verrà penalizzata l'assenza del direttorio con il nome indicato e/o l'assenza dei file nel direttorio specificato, al momento della copia automatica del direttorio e dei file. **ALLA SCADENZA DEL TEMPO A DISPOSIZIONE VERRÀ INFATTI ATTIVATA UNA PROCEDURA AUTOMATICA DI COPIA, PER OGNI STUDENTE DEL TURNO, DEI FILE CONTENUTI NEL DIRETTORIO SPECIFICATO.**
- 3) Il tempo a disposizione per la prova è di **120 MINUTI** per lo svolgimento di tutto il compito e di **75 minuti** per lo svolgimento della sola parte C.
- 4) Non è ammesso **nessun tipo di scambio di informazioni** né verbale né elettronico, pena la invalidazione della verifica.
- 5) L'assenza di commenti significativi verrà penalizzata.
- 6) **AL TERMINE DELLA PROVA È INDISPENSABILE CONSEGNARE IL TESTO DEL COMPITO (ANCHE IN CASO CHE UNO STUDENTE SI RITIRI): IN CASO CONTRARIO, NON POTRÀ ESSERE EFFETTUATA LA CORREZIONE DEL COMPITO MANCANDO IL TESTO DI RIFERIMENTO.**

Esercizio

Si realizzi un programma **concorrente** per UNIX che deve avere una parte in **Bourne Shell** e una parte in **C**.

La parte in Shell deve prevedere **3 parametri**: il primo deve essere il nome assoluto di un direttorio che identifica una gerarchia (G) all'interno del file system, il secondo deve essere il nome relativo semplice di un direttorio (D), mentre il terzo parametro deve essere considerato un numero intero strettamente positivo (K).

Il programma deve cercare nella gerarchia **G** specificata tutti i direttori di nome **D** che contengono almeno un file che abbiano dimensione in byte uguale a **K**. Si riporti il nome assoluto di tali direttori sullo standard output. In ognuno di tali direttori trovati, si deve invocare la parte in C, passando come parametri i nomi dei file trovati (**F1, F2, ... FN**) che soddisfano la condizione precedente.

La parte in C accetta un numero variabile **N** di parametri che rappresentano **N** nomi di file (**F1, F2, ... FN**): la lunghezza in caratteri dei file è la stessa (questo viene garantito dalla parte shell e NON deve essere controllato).

Il processo padre deve generare **N processi figli (P0 ... PN-1)** ognuno dei quali è associato ad uno dei file **Fi**. Ogni processo figlio deve leggere i singoli carattere del file associato **Fi sempre** fino alla fine. I processi figli e il processo padre devono attenersi a questo **schema di comunicazione a pipeline**: il figlio **P0** comunica con il figlio **P1** che comunica con il figlio **P2** etc. fino al figlio **PN-1** che comunica con il **padre**; questo schema a pipeline deve essere ripetuto per ogni carattere dei file associati **Fi**. In particolare, il figlio **P0** passa in avanti (cioè comunica) via via tutti i caratteri che legge dal proprio file, il figlio seguente **P1** se verifica l'uguaglianza del carattere ricevuto con il proprio carattere corrente (cioè quello di uguale posizione rispetto all'inizio del file) passa avanti questo carattere 'normale' al figlio seguente **P2**, etc. fino al figlio **PN-1** che lo passa al **padre**; nel caso in cui, in un qualunque punto della pipeline, non si verifichi l'uguaglianza del carattere ricevuto con il proprio carattere corrente deve essere passato avanti invece che il carattere 'normale' il valore -1 come *char*. In ogni modo, una comunicazione relativa ad ogni carattere partito dal processo P0 arriva al processo padre. Il padre può ricevere:

- un carattere 'normale' e in questo caso lo deve stampare su standard output con l'indicazione della posizione (in termini di **long int**) in cui si trova il carattere trovato uguale in tutti i file analizzati dai figli;
- un carattere di valore -1 e in questo caso deve stampare su standard output l'indicazione della posizione (in termini di **long int**) in cui non è stata verificata l'uguaglianza in tutti i file analizzati dai figli.

Al termine, ogni processo figlio **Pi** (a parte il primo che torna sempre zero), deve ritornare al padre il numero uguaglianze verificate (*supposto minore di 255*) e il padre deve stampare su standard output il PID di ogni figlio e il valore ritornato.