

SISTEMI OPERATIVI E LAB.

(A.A. 11-12) – 01 GIUGNO 2012

IMPORTANTE:

- 1) Fare il login sui sistemi in modalità Linux usando il proprio **username** e **password**, attivare syncexam.sh e passare in modalità testuale.
- 2) I file prodotti devono essere collocati in un **sottodirettorio** (che deve essere nella directory studente_XXX) che deve essere creato e avere nome **ESAME08Giu12-1-1**. 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 **75 MINUTI**.
- 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 (già realizzata) e una parte in C.

La parte in C accetta un numero *variabile* di parametri (**maggiore o uguale a 3, da controllare**) che rappresentano le seguenti informazioni: i primi **N** parametri rappresentano nomi di file **F1...FN**, mentre l'ultimo parametro deve essere considerato un numero intero positivo (**H, da controllare**) che rappresenta la lunghezza media in linee dei file F1...FN.

Il processo padre deve generare **N processi figli (P0 ... PN-1)**: ogni processo figlio è associato al corrispondente file **Fi**. Ognuno di tali processi figli deve creare a sua volta un **processo nipote (PP0 ... PPN-1)**: ogni processo nipote **PPi** esegue concorrentemente calcolando la **lunghezza in linee** del file associato **Fi** usando in modo opportuno il comando *wc* di UNIX/Linux.

Ogni processo figlio **Pi**, terminato il nipote **PPi** corrispondente, deve **convertire** in termini di **valore intero (lunghezza)** quanto scritto dal processo nipote **PPi** in termini di caratteri sullo standard output; quindi deve controllare tale valore rispetto al valore **H**: se **lunghezza** è maggiore o uguale a **H**, deve comunicare al padre la stringa "**Sopra media**", altrimenti la stringa "**Sotto media**".

Per ogni figlio, il padre ha il compito di stampare su standard output, rispettando l'ordine dei file, la stringa ricevuta riportando sia il nome del file che il valore della stringa.

Al termine, ogni processo figlio **Pi** deve ritornare al padre il valore di ritorno del proprio processo nipote **PPi** e il padre deve stampare su standard output il PID di ogni figlio e il valore ritornato.

NOTA BENE: Chiaramente il testo assume che la **lunghezza in linee** di ogni possibile file passato come parametro sia rappresentabile con un intero del linguaggio C; quindi la corrispondente soluzione può fare riferimento a questa ipotesi senza bisogno di specificarlo.