

SISTEMI OPERATIVI E LAB.

(A.A. 21-22) – 15 GIUGNO 2022

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 un numero variabile di parametri **Q** (con **Q** maggiore o uguale a **3**): che devono essere **nomi assoluti di directory** che identificano **Q** gerarchie (**G1, G2, ...**) all'interno del file system. Il comportamento atteso dal programma, dopo il controllo dei parametri, è organizzato in **Q** fasi, una per ogni gerarchia.

Il programma, per ognuna delle **Q** fasi, deve esplorare la gerarchia **G** corrispondente - tramite un file comandi ricorsivo, **FCR.sh** - e deve contare **globalmente** tutti i file che saranno cercati secondo quanto di seguito specificato. Il file comandi ricorsivo **FCR.sh** deve cercare in ogni gerarchia **G** che esista almeno un file (**F**) **di lunghezza non nulla e che** contenga (nel suo contenuto) solo caratteri di tipo **alfabetico minuscolo**: per ogni file trovato **F**, si deve riportare, contestualmente, il suo nome assoluto sullo standard output. Al termine di tutte le Q fasi, si deve riportare sullo standard output il numero N di file trovati e si deve invocare la parte in C, passando come parametri i nomi dei file *trovati* (**F1, F2, ... FN**).

NOTA BENE NEI DUE FILE COMANDI SI USI OBBLIGATORIAMENTE:

- una variabile di nome **G** per le singole gerarchie di ognuna delle **Q** fasi;
- il nome **/tmp/nomiAssoluti** per il file temporaneo;
- una variabile di nome **N** per contenere il numero dei file trovati globalmente;
- una variabile di nome **F** per identificare, via via, i singoli file delle directory esplorate in FCR.sh.

La parte in C accetta un numero variabile di parametri **N** (con **N** maggiore o uguale a **2**) che rappresentano nomi di file (**F1, ...FN**). Il processo padre deve generare **N** processi figli (**P0 ... PN-1**): i processi figli **Pi** (con **i** variabile da **0 a N-1**) sono associati agli **N** file **Ff** (con $f = i + 1$). Ogni processo figlio **Pi** deve leggere tutti i caratteri del file associato **Ff** e appena viene trovata una occorrenza di un **carattere alfabetico minuscolo**, il processo figlio **Pi** deve comunicare al padre tale carattere e deve ricevere dal padre l'indicazione di stampare o meno su standard output delle informazioni (*vedi dopo* *). Il padre deve ricevere, rispettando l'ordine dei file **Ff**, da ogni figlio via via i caratteri di tipo alfabetico minuscolo letti dai figli. Quindi, al processo padre deve arrivare **un insieme di char** (che via via potrebbe diminuire in quantità in dipendenza della terminazione dei figli): sicuramente il primo insieme (questo viene garantito dalla parte Shell) è costituito dal primo carattere alfabetico minuscolo inviato dal figlio **P0**, dal primo carattere alfabetico minuscolo inviato dal figlio **P1**, etc., dal primo carattere alfabetico minuscolo inviato dal figlio **PN-1**. Per ogni insieme ricevuto, il padre deve determinare il valore **massimo** e, SOLO AL PROCESSO FIGLIO CHE HA INVIATO TALE CARATTERE, deve indicare (*) di stampare su standard output **il carattere trovato, assieme all'indice d'ordine del processo, il suo pid, la posizione all'interno del file e il nome del file associato**, mentre a tutti gli altri processi figli deve indicare di non stampare[®].

Al termine, ogni processo figlio **Pi** deve ritornare al padre il numero di stampe effettuate (supposto strettamente minore di 255); il padre deve stampare su standard output il PID di ogni figlio e il valore ritornato.

NOTA BENE NEL FILE C main.c SI USI OBBLIGATORIAMENTE:

- una variabile di nome **N** per il numero di file;
- una variabile di nome **i** per l'indice dei processi figli;
- una variabile di nome **ch** per il carattere letto dai file dai figli;
- una variabile di nome **cur** per la posizione all'interno del file (di tipo long int).

OSSERVAZIONE PER CHI SVOLGE IL COMPITO COMPLETO: Per quanto specificato nei criteri di ricerca della parte SHELL, la soluzione della parte C si semplifica dato che i figli devono inviare al padre. Indistintamente, tutti i caratteri letti dal file associato.

[®] Per questo tipo di interazione, volendo si possono usare i segnali. Nel caso si usino le pipe, fare attenzione che il padre deve inviare l'indicazione SOLO ai figli che non sono terminati!