

ESERCITAZIONE 28 MARZO 2024 – A CASA

LN (seguito)

- 1) Creare una sottodirectory di nome `provaDirBis` nella propria HOME directory; verificare il numero di link della sottodirectory creata; creare con un editor almeno un file in tale sottodirectory; usare i nomi assoluti dei vari link della sottodirectory creata per visualizzarne il contenuto; cancellare la sottodirectory.
- 2) Creare un link SOFTWARE di nome `provaLinkSoftware` per un file della propria HOME directory nella sottodirectory `provaDir` (creata la scorsa esercitazione); verificare il numero di link, le proprietà del link software e il suo uso.
- 3) Cancellare (casomai dopo averne fatta una copia, se il contenuto è importante) il file della propria HOME directory usato come sorgente del link SOFTWARE; verificare cosa succede ad usare il link SOFTWARE.
- 4) Dopo aver ricreato il file sorgente del link SOFTWARE, cancellare a questo punto il link SOFTWARE e verificare che il file sorgente continui ad essere accessibile.
- 5) Provare a creare un link HARDWARE di nome `provaLinkDir`, nella propria HOME directory, per la sottodirectory `provaDir`: il comando ha successo?
- 6) Dopo aver creato una sottodirectory di nome `linkSw` nella propria HOME directory, provare a creare un link SOFTWARE di nome `provaLinkDir` per la sottodirectory `provaDir` nella nuova sottodirectory: il comando ha successo?

ESECUZIONE IN BACKGROUND

- 7) Mandare in esecuzione con le opportune ridirezioni di standard output e di standard error il comando `ls -lR` / usando l'esecuzione in background.
- 8) Verificare i processi attivi nella sessione prima che finisca il comando lanciato in background.

DATE

- 9) Verificare la data (e l'ora corrente).

DIFF

- 10) Creare, con un editor oppure con la ridirezione dello standard output usando il comando-filtro `cat`, un file che contenga almeno 5 o 6 linee; crearne una copia e quindi verificare il funzionamento del comando `diff` utilizzando i due file prodotti.
- 11) Modificare con un editor alcune parti di uno dei due file precedentemente creati; quindi, verificare di nuovo il funzionamento del comando `diff` utilizzando i due file prodotti.

PARTE SHELL DI TESTI DI ESAMI

OSSERVAZIONE: Per tutti i testi che NON sono di una prima prova in itinere, ma di un compito completo, al posto della invocazione della parte C, si inserisca un echo che illustri con quali parametri andrebbe invocata la parte C!

a) TESTO PARTE SHELL DELL'ESAME DEL 15 FEBBRAIO 2017

La parte in Shell deve prevedere due parametri: il primo deve essere il **nome assoluto di una directory** che identifica una gerarchia (**G**) all'interno del file system; il secondo parametro deve essere considerato un numero intero **X** strettamente positivo. Il programma deve cercare nella gerarchia **G** specificata tutte le directory che contengono almeno **un** file leggibile con lunghezza in linee pari **X** e il cui contenuto del file sia tale per cui **tutte** le sue linee contengano almeno un carattere numerico. Si riporti il nome assoluto di tali directory sullo standard output e quindi i nomi di tutti i file trovati (**F1, F2, ... FN**). Quindi in ogni directory trovata, chiesta conferma all'utente, si deve invocare la parte in C, passando come parametri i **nomi dei file trovati (F1, F2, ... FN)** e la loro lunghezza in linee **X**.

b) *TESTO PARTE SHELL DELL'ESAME DEL 9 FEBBRAIO 2015*

La parte in Shell deve prevedere **due** parametri: il primo deve essere il **nome assoluto di una directory** che identifica una gerarchia (**G**) all'interno del file system, mentre il secondo deve essere considerato un numero intero strettamente positivo (**K**). Il programma deve cercare nella gerarchia **G** specificata tutte le directory che contengono almeno **un** file leggibile con lunghezza in linee pari a **K**: si riporti il nome assoluto di tali directory sullo standard output. Al termine dell'intera esplorazione ricorsiva di G, per ogni file trovato **Fi** si deve richiedere all'utente un numero **Xi** intero strettamente positivo e minore o uguale a **K**: quindi si deve invocare la parte in C passando come parametri i nomi assoluti dei file trovati intervallati dal numero corrispondente chiesto all'utente (perciò i parametri saranno: **F0, X0, F1, X1, ... FN-1, XN-1**).

c) *TESTO DELLA PRIMA PROVA IN ITINERE DEL 17 APRILE 2015*

La parte in Shell deve prevedere un numero variabile **N+1** di parametri (con **N** maggiore o uguale a 2): il primo parametro deve essere considerato un numero intero **X** strettamente positivo, mentre gli altri **N** devono essere **nomi assoluti di directory** che identificano **N** gerarchie (**G1, G2, ... GN**) all'interno del file system. Il comportamento atteso dal programma, dopo il controllo dei parametri, è organizzato in **N** fasi, una per ogni gerarchia.

Il programma, per ognuna delle **N** fasi, deve esplorare la gerarchia **Gi** corrispondente - tramite un file comandi ricorsivo, **FCR.sh** - e deve contare **globalmente** tutti i file leggibili che soddisfano la seguente specifica: il contenuto del file deve essere tale per cui **almeno X** linee terminino con il carattere **'t'**. Al termine di tutte le **N** fasi, si deve riportare sullo standard output il numero totale di file trovati globalmente che soddisfano la specifica precedente (*file trovati*); quindi, si deve riportare sullo standard output il **nome assoluto** di ogni *file trovato* chiedendo contestualmente all'utente un numero **K** strettamente positivo e **strettamente minore di X**: tale numero deve essere usato per riportare sullo standard output la linea **K-esima** del *file trovato* corrente.

d) *TESTO PARTE SHELL DELL'ESAME DEL 14 LUGLIO 2021*

La parte in Shell deve prevedere un numero variabile di parametri **Q+2** (con **Q** maggiore o uguale a 2): i primi **Q** devono essere **nomi assoluti di directory** che identificano **Q** gerarchie (**G1, G2, ...**) all'interno del file system, mentre gli ultimi due parametri devono essere considerati numeri interi strettamente positivi (**H** e **M**). 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 cercare tutte le directory che contengono un **numero strettamente minore di H ma maggiore o uguale a 2** di file (**F1, F2, ...**) il cui nome sia costituito da esattamente **2** caratteri e con lunghezza in linee esattamente uguale a **M**. Si riporti il nome assoluto di tali directory sullo standard output. In ognuna di tali directory trovate, si deve invocare la parte in C, passando come parametri i nomi dei file trovati (**F1, F2, ...**) che soddisfano la condizione precedente e il numero **M**.

NOTA BENE NEI DUE FILE COMANDI SI USI OBBLIGATORIAMENTE:

- una variabile di nome H per contenere il penultimo parametro di FCP.sh;
- una variabile di nome M per contenere l'ultimo parametro di FCP.sh;
- una variabile di nome G per le singole gerarchie di ognuna delle Q fasi;
- una variabile di nome F per identificare, via via, i singoli file delle directory esplorate;
- una variabile di nome cont per contare i file che soddisfano la specifica