

Sistemi Operativi

15 giugno 2023

Programmazione di Sistema Unix

Si progetti un'applicazione concorrente in C che consenta all'amministrazione di una compagnia aerea di monitorare i voli effettuati da un determinato aereo. L'applicazione deve presentare la seguente interfaccia:

voli_effettuati dir

dove ***dir*** è una directory relativa.

Si supponga che le informazioni relative ai voli effettuati da un determinato aereo siano salvate in un unico file nella directory ***dir***. Ad esempio, nella directory ***dir*** il file V0001.txt contiene tutte le informazioni relative ai voli effettuati dal velivolo identificato dal codice univoco V0001. Ciascuna riga del file contiene le informazioni di un certo volo: durata del volo (in minuti), codice volo, città di partenza, città di arrivo, data del volo (nel formato AAAAMMGG). I campi di ciascuna riga sono separati da virgole.

L'applicazione concorrente deve essere composta da un processo **P0** che riceve dall'utente (via terminale) un identificativo di un velivolo, una data (nel formato AAAAMMGG) e il numero N (intero) di risultati da mostrare a video. Ad esempio, se l'utente inserisce "V123", "20221201" e "3", l'applicazione deve mostrare i 3 voli più lunghi effettuati dal velivolo "V123" in data 1 dicembre 2022.

Per ogni richiesta inserita, il processo **P0** crea tre processi figli: **P1**, **P2** e **P3**. **P1** seleziona i voli effettuati dall'aereo indicato e nella data specificata dall'utente e invia tali informazioni a **P2**. **P2** ordina i voli in modo crescente (sulla base della durata del volo) e invia tali informazioni a **P3**. **P3** mostra a video gli N voli più lunghi.

P3 deve inoltre mandare il segnale SIGUSR2 a **P0**, che tiene traccia del numero di segnali SIGUSR2 ricevuti e lo stampa a video ogniqualvolta riceve un segnale SIGUSR2. A tal fine si ricorda l'esistenza della funzione `pid_t getppid(void)`; che restituisce il PID del processo padre.

L'applicazione termina quando l'utente inserisce la stringa "fine".

Sistemi Operativi

15 giugno 2023

Programmazione Shell

Si scriva uno script shell (strutturato su più file con invocazione ricorsiva e senza l'uso del comando find) per aiutare il curatore di un museo a consultare le opere esposte per una determinata esposizione durante un anno specifico. In particolare, lo script deve presentare la seguente interfaccia:

museo ***esposizione*** ***anno***

dove ***esposizione*** è un stringa alfanumerica che identifica univocamente una certa esposizione organizzata dal museo e ***anno*** è un nome relativo di directory formato da soli caratteri numerici.

Si supponga che le informazioni sulle opere esposte nel corso dell'anno siano memorizzate all'interno della directory ***anno*** (o in una delle sue sottodirectory) in file di testo con estensione .txt. Ad esempio, le informazioni relative a tutte le esposizioni avvenute nel mese di maggio sono memorizzate in un file di testo maggio.txt. Ciascuna riga di tali file conterrà le informazioni relative a un'opera esposta in una certa esposizione con, in quest'ordine, la data di inizio dell'esposizione (nel formato AAAAMMGG), l'identificativo univoco dell'esposizione (E412, BA39, etc.), il titolo dell'opera e l'artista. Si noti che un'esposizione potrebbe essere riproposta più volte nel corso di un anno, con variazioni sulle opere proposte sulla base della disponibilità delle stesse.

Lo script ***museo*** deve quindi esplorare ricorsivamente la directory ***anno*** e tutte le sottodirectory per analizzare le informazioni presenti nei file .txt, selezionare esclusivamente le informazioni relative alle opere dell'esposizione di interesse (argomento ***esposizione***), estrarre le (sole) informazioni su data e titolo e quindi scrivere tali dati in un file di log (chiamato esposizione.txt) nella home directory dell'utente. Se il file di log esiste già, lo script deve sovrascriverne il contenuto.

Prima di terminare, il file comandi deve anche stampare a video il mese (ovverosia il nome del file corrispondente, comprensivo di suffisso .txt) in cui sono state esposte più opere, inteso come file

che ha il maggior numero di righe (a prescindere dal codice esposizione specificato).