

Compito di Sistemi Operativi

21 Novembre 2017

Esercizio 1 (5 punti)

Si consideri il seguente programma concorrente:

```
#include <unistd.h>

int main(void) {
    int i;
    for (i = 0; i < 4; i++) {
        if (fork() && (i == 2)) {
            break;
        }
    }
}
```

Disegnare l'albero dei processi, giustificando la risposta fornita.

Esercizio 2 (8 punti)

Si considerino i due processi P_1 e P_2 :

<code>wait(s1);</code>	<code>wait(s2);</code>
<code>x = x + 1;</code>	<code>y = y * 2;</code>
<code>wait(s2);</code>	<code>wait(s3);</code>
<code>y = y - x;</code>	<code>z = z - y;</code>
<code>wait(s3);</code>	<code>signal(s2);</code>
<code>z = x + 2 * y - z;</code>	<code>wait(s1);</code>
<code>signal(s3);</code>	<code>x = x + 2;</code>
<code>signal(s2);</code>	<code>signal(s1);</code>
<code>signal(s1);</code>	<code>signal(s3);</code>

con i semafori s_1 , s_2 e s_3 inizializzati ad 1. Può verificarsi uno stallo? In caso affermativo, descrivere una successione di esecuzione tale da provocarlo; viceversa si riportino tutte le possibili combinazioni di valori assunti dalle variabili x , y , z a fronte dell'esecuzione concorrente dei processi P_1 e P_2 .

Esercizio 3 (8 punti)

Si consideri un sistema con scheduling a priorità con tre code, A, B, C, di priorità decrescente e con prelazione tra code. Le code A e B sono servite con modalità RR, con time slice di 8 e 16 millisecondi; la coda C è FCFS. Se un processo nella coda A o B consuma il suo quanto di tempo, viene spostato in fondo alla coda B o C, rispettivamente. Si supponga che i processi P_1 , P_2 , P_3 , P_4 arrivino tutti al tempo 0 (nell'ordine indicato), nelle code A, A, C, B, e con CPU burst pari a 30, 28, 4 e 20 millisecondi. Si tracci il diagramma di Gantt relativo all'esecuzione dei quattro processi e si calcoli il tempo di risposta relativo a ciascun processo ed il tempo medio di attesa.

Esercizio 4 (9 punti)

Si dica se le seguenti affermazioni sono vere o false, motivando la risposta data.

- 1) Il sistema operativo è un esempio di software applicativo.
- 2) Il SO fornisce un'interfaccia agevole fra l'uomo e la macchina.
- 3) Un computer può avere più SO installati, ma solo un sistema operativo ne ha il controllo in un dato momento.
- 4) La multiprogrammazione è la tecnica impiegata per utilizzare più CPU per eseguire i programmi.
- 5) L'elaborazione batch implica un alto livello di interazione tra l'utente e il programma.
- 6) Il multitasking non presuppone l'interazione dell'utente con un programma in esecuzione.
- 7) Un SO è portatile se è in grado di far eseguire più processi utente contemporaneamente.
- 8) In un SO multiprogrammato è possibile il passaggio dallo stato RUN allo stato READY.
- 9) Il time sharing, gestito con la politica round robin, favorisce la condivisione delle periferiche.
- 10) Lo scheduler della CPU determina quali e quanti programmi si trovano in memoria ad un certo istante.
- 11) Con RR, il "quanto" è il tempo di CPU assegnato ad ogni processo, prima della prelazione da parte del SO.
- 12) L'algoritmo di scheduling FCFS è ottimo.
- 13) Un indirizzo logico specifica una posizione effettiva nella memoria principale.
- 14) Il registro limite contiene l'ultimo indirizzo di una partizione.
- 15) La prima pagina, in un sistema di memoria gestito con paginazione su richiesta, è la pagina 0.
- 16) Il PCB è una struttura dati che memorizza tutte le informazioni su un processo.
- 17) La FAT può considerarsi un modo per realizzare l'allocazione indicizzata dei file.
- 18) La directory in Linux è un file speciale, che contiene informazioni sugli altri file in essa contenuti.