

Domande avvenuti

- 1) Com'è fatto lo schema di un device driver dal punto di vista del programma
Che azioni deve fare il device driver?
Descrivere con precisione cosa succede dentro al device driver
 - 2) Descrivere un generico semaforo
Come si implementa un semaforo? Far vedere il codice
 - 3) caratteristiche degli algoritmi di schedulazione per sistemi in tempo reale
 - fattore di utilizzazione, definizione e cosa comporta
 - minimo comune multiplo dei periodi, come si ottiene e cosa significa nel sistema, a cosa ci serve concretamente
 - rate monotonic: condizione sufficiente per la schedulabilità, definizione e quanto vale.
 - condizione necessarie per EDF, è migliore EDF o RM?
 - può succedere che un algoritmo sia schedulabile con EDF e non RM?
 - quale tra i due è più potente?
 - cosa vuol dire che un algoritmo è ottimo?
 - 4) sistema di protezione di bell-lapadula
 - uso di bell-lapadula sulla matrice di accessi
 - esempio cavallo di troia per dimostrare l'utilità di bell-lapadula
 - come si vede un file protetto nella matrice di accessi? Cioè cosa viene mostrato
 - regole di bell-lapadula, modificare la matrice di accessi per far sì che includa tali regole
 - 5) algoritmo del banchiere:
 - a cosa serve
 - descrizione
 - quando viene chiamato, come funziona, cosa fa
 - 6) come viene gestito un page fault in un sistema operativo
 - dire precisamente l'ordine delle operazioni che vengono effettuati, gli eventi fondamentali del page fault avvenuti lo vuole passo per passo e nel dettaglio, come se stessi descrivendo un algoritmo
-
-

Domande Minici

- 1) Spiegare la fork
- 2) Spiegare la terminazione di un processo
- 3) Cosa restituisce una chiamata di una funzione della famiglia exec
- 4) Sincronizzazione tra i processi mediante l'utilizzo di segnali
- 5) Come possiamo usare un handler definito dall'utente?
- 6) System call alarm e sleep
- 7) Meccanismo dei permessi
 - Differenza dei permessi tra file e cartelle
 - Come vengono definiti i permessi speciali?
 - Comandi per cambiare i permessi
- 8) thread in linux
 - è necessario includere una libreria per usare i thread?
 - esiste il concetto nativo di thread in linux? Oppure è il programmatore a doverlo fare?
 - perché usiamo la libreria pthread invece del concetto nativo di thread di linux? Per la portabilità su sistemi non multithread nativi
 - come vengono identificati i thread usando la libreria
 - come si crea un thread usando pthread?
 - come si termina il thread?
- 9) comunicazione dei processi tramite il meccanismo delle pipe
 - è previsto un naming nello scambio del messaggio? È previsto che ci venga scritto il destinatario sul singolo messaggio
 - con che ordine vengono letti i messaggi?
 - quali sono i processi che possono comunicare tramite una pipe?
 - come fanno per permettere la comunicazione tramite pipe? Ordine dei comandi affinché il figlio possa comunicare con il padre attraverso la pipe
 - è necessario preoccuparsi di fare sincronizzazione sulla pipe? Bisogna scrivere il codice pensando alla sincronizzazione?
 - come si scrive all'interno della pipe?
 - cosa ritornano le funzioni per usare le pipe?