

Nume și grupă:

Sisteme de Operare

9 septembrie 2013

Timp de lucru: 100 de minute

Notă: Toate răspunsurile trebuie justificate

1. (1 punct) De ce **NU** este posibil ca un planificator să fie, simultan, echitabil (fair) și productiv (să ofere un throughput mare)?
2. (1 punct) De ce este afectat sistemul gazdă (host) în cazul apariției unei erori fatale la nivelul nucleului unui container OpenVZ?
3. (1 punct) Funcția `malloc` alocă memorie. Funcția `calloc` alocă memorie și completează spațiul cu zero-uri. De ce un apel `calloc` generează, de obicei, mai multe page fault-uri decât `malloc`?
4. (1 punct) De ce este preferată folosirea apelurilor asincrone, acolo unde există, în locul celor sincrone?
5. (1 punct) Trei thread-uri afișează continuu respectiv mesajul "red", "green", "blue". Descrieți, ținând cont de sincronizare, cele trei funcții aferente thread-urilor astfel încât să se afișeze "red green blue red green blue red green blue ...".
6. (1 punct) De ce este indicat, din punct de vedere al securității, să fie folosit apelul `printf("%s", argv[1])` în locul apelului `printf(argv[1])`?
7. (1 punct) Creați un paragraf adevărat, informativ și argumentat din domeniul sistemelor de operare care să cuprindă ca subiecte principale noțiunile de *fișier mapat în memorie* și *apel de sistem*.
8. (1 punct) Creați un paragraf adevărat, informativ și argumentat din domeniul sistemelor de operare care să cuprindă ca subiecte principale noțiunile de *hard link*, *symbolic link* și *inode*.
9. (1 punct) Care sunt asocierile dintre clasele asincron, zero-copying, sincron blocant și apelurile/tehnologiile `read`, `readv`, `ReadFile`, `mmap`, `sendfile`, `Completion Ports`, `io_submit`?
10. (1 punct) Care sunt asocierile dintre perechile de mai jos (unul la unul, unul la mai multe, mai multe la unul, mai multe la mai multe)?
 - (procesor, registru)
 - (spațiu de adresă, stivă)
 - (procesor, proces)
 - (hard link, inode)
 - (mutex, thread-uri în așteptare)
11. (2.5 puncte) Se dă un server TCP pe care trebuie să îl caracterizați din punct de vedere al performanței. În general, asta înseamnă să măsurați, de exemplu:
 - numărul de cereri servite pe secundă
 - numărul maxim de clienți serviți simultan

- throughput de date

și cum variază aceste trei mărimi una în funcție de cealaltă.

Descrieți cum realizați o arhitectură de sistem (topologie, legături, scenarii de folosire) care să poată măsura acești parametri. Elaborați având în vedere faptul că software-ul de server rulează pe un calculator server mult mai performant decât calculatoarele pe care le aveți la dispoziție pentru testare.

12. (2.5 puncte) Pentru aceeași situație ca la punctul anterior, descrieți arhitectura software a software-ului de măsură: single threaded, multi-threaded, multi-proces? Ce alți parametri ați mai putea măsura?

În conformitate cu ghidul de etică al Departamentului de Calculatoare, declar că nu am copiat și nu voi copia la această lucrare. De asemenea, nu am ajutat și nu voi ajuta pe nimeni să copieze la această lucrare.

Nume și grupă:

Semnătură:.....