
Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Operativni sistemi 2 (SI3OS2, IR3OS2)

Nastavnik: prof. dr Dragan Milićev

Odsek: Softversko inženjerstvo, Računarska tehnika i informatika

Kolokvijum: Drugi, novembar 2012.

Datum: 1.12.2012.

Drugi kolokvijum iz Operativnih sistema 2

Kandidat: _____

Broj indeksa: _____ *E-mail:* _____

Kolokvijum traje 1,5 sat. Dozvoljeno je korišćenje literature.

Zadatak 1 _____ /10 *Zadatak 3* _____ /10
Zadatak 2 _____ /10

Ukupno: _____ /30 = _____ %

Napomena: Ukoliko u zadatku nešto nije dovoljno precizno definisano, student treba da uvede razumnu pretpostavku, da je uokviri (da bi se lakše prepoznala prilikom ocenjivanja) i da nastavi da izgrađuje preostali deo svog odgovora na temeljima uvedene pretpostavke. Ocenjivanje unutar potpitanja je po sistemu "sve ili ništa", odnosno nema parcijalnih poena. Kod pitanja koja imaju ponuđene odgovore treba **samo zaokružiti** jedan odgovor. Na ostala pitanja odgovarati **čitko, kratko i precizno**.

1. (10 poena) Upravljanje deljenim resursima

U nekom sistemu svaki proces i svaki resurs ima svoj jedinstveni identifikator (tipa `unsigned int`), a zauzeće resursa od strane procesa prati se u matrici `resourceAlloc` u kojoj vrste označavaju procese, a kolone resurse. U toj matrici vrednost `-1` u ćeliji (p, r) označava da je proces sa identifikatorom p zauzeo resurs sa identifikatorom r , vrednost `1` označava da je proces p tražio, ali nije dobio resurs r (i čeka da ga dobije), a vrednost `0` označava da proces p nije ni zauzeo ni tražio resurs r . Implementirati operacije `allocate()` i `release()` koje procesi treba da pozivaju kada traže, odnosno oslobađaju resurse.

```
const unsigned MAXPROC = ...; // Maximum number of processes
const unsigned MAXRES = ...; // Maximum number of resources
extern unsigned numOfProc;    // Actual number of processes
extern unsigned numOfRes;    // Actual number of resources

int resourceAlloc[MAXPROC][MAXRES];

int allocate (unsigned pid, unsigned rid);
int release  (unsigned pid, unsigned rid);
```

Operacija `allocate()` vraća `1` ako je traženi resurs dodeljen datom procesu, a `0` ako nije. Operacija `release()` dodeljuje oslobođeni resurs nekom drugom (bilo kom) procesu koji je čekao na taj resurs i vraća njegov identifikator, ako takav proces postoji; u suprotnom, samo oslobađa resurs i vraća `-1`. Ne primenjuje se nikakav algoritam sprečavanja, izbegavanja, ili detekcije mrtve blokade.

Rešenje:

2. (10 poena) Upravljanje memorijom

Neki sistem podržava memorijski preslikane fajlove (*memory mapped files*) i nudi sistemski poziv `mmapfile` koji preslikava sadržaj fajla zadatog imenom u prvi deo adresnog prostora pozivajućeg procesa koji operativni sistem pronade kao nealociran, a koji je dovoljno veliki da se u njega preslika sadržaj veličine datog fajla. Ovaj poziv vraća pokazivač na alocirani segment virtuelne memorije u koji je preslikan dati fajl, odnosno *null* ukoliko poziv nije uspeo.

U binarnom fajlu `log.bin` snimljeni su podaci u sledećem binarnom formatu:

- na početku je snimljen ceo broj n (tipa `int`);
- u nastavku je tačno n zapisa tipa `DailySales`, u čijem polju `quantity` tipa `double` je upisana količina nekog artikla prodana u jednom danu.

Napisati funkciju koja tehnikom memorijski preslikanog fajla izračunava ukupnu prodatu količinu datog artikla. Smatrati da je ulazni fajl sigurno korektno zapisan u navedenom formatu.

Rešenje:

3. (10 poena) Upravljanje diskovima

U redu zahteva za pristup disku nalaze se zahtevi za pristup sledećim cilindrima (po redosledu pristizanja):

62, 43, 95, 130, 70, 41, 51

Prethodno opsluženi zahtev je bio na cilindru 46, a glava se kreće prema cilindrima sa većim brojevima. Napisati redosled opsluživanja ovih zahteva ukoliko je algoritam raspoređivanja:

a)(5) *Shortest-Seek-Time-First*

Odgovor: _____

b)(5) *C-Scan*

Odgovor: _____