
Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Operativni sistemi 2 (IR3OS2)
Nastavnik: prof. dr Dragan Milićev
Odsek: Računarska tehnika i informatika
Kolokvijum: Prvi, decembar 2016.
Datum: 24.12.2016.

Prvi 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) Raspoređivanje procesa

U nekom sistemu koristi se *Multilevel Feedback-Queue Scheduling* (MFQS) na sledeći način:

- Postoje tri reda spremnih procesa: HP (*High Priority*), MP (*Medium Priority*) i LP (*Low Priority*).
- Globalni algoritam raspoređivanja je po prioritetu, s tim da HP ima najviši, a LP najniži prioritet.
- Raspoređivanje u svim redovima je *Round-Robin* (RR), samo sa različitim vremenskim kvantomom koji se dodeljuje procesima.
- Procesima koji se uzimaju iz HP dodeljuje se vremenski kvantum 4, onima koji se uzimaju iz MP vremenski kvantum 8, a onima iz LP kvantum 16.
- Proces koji je tek postao spreman (bio je blokiran ili je tek kreiran) smešta se u red prema proceni dužine svog sledećeg naleta izvršavanja (*CPU burst*) na sledeći način: ako je procena $\tau \leq 4$, stavlja se u red HP; ako je procena $4 < \tau \leq 8$, stavlja se u red MP; inače se stavlja u red LP.
- Procena dužine sledećeg naleta izvršavanja vrši se eksponencijalnim usrednjavanjem sa koeficijentom $\alpha = 1/2$ i pamti se kao ceo broj (vrši se odsecanje pri izračunavanju).
- Proces kome je istekao vremenski kvantum smešta se u MP ako je prethodno bio uzet iz HP, odnosno u LP ako je prethodno bio uzet iz MP ili LP.

Posmatra se jedan proces koji ima sledeće nalete izvršavanja (označeni sa C i dužinom trajanja naleta) i ulazno/izlazne operacije (označene sa I/O), i sa početnom procenom $\tau = 3$:

C13, I/O, C2, I/O, C9, I/O, C12, I/O, C3

a)(7) Dati oznake redova spremnih procesa (HP, MP, LP) u koje je ovaj proces redom stavljan, i to za svako stavljanje procesa u neki od redova spremnih (odgovor dati u obliku npr. HP, MP, LP, LP, LP, ...)

Odgovor: _____

b)(3) Koja je vrednost procene dužine sledećeg naleta izvršavanja τ nakon ove sekvence?

Odgovor: $\tau =$ _____

2. (10 poena) Međuprocesna komunikacija pomoću deljene promenljive

Na programskom jeziku Java implementirati monitor `Computer` čiji je interfejs dat dole. Ovaj monitor treba da prihvati vrednost za dva operanda, x i y , koje uporedni procesi mogu da dostavljaju pozivima procedura `writeX` i `writeY`, respektivno. Monitor treba da primi tačno po jednu vrednost svakog od ta dva operanda, ali najpre x , pa tek onda y (baš u tom redosledu), i tek nakon što ih primi oba, može da prihvati zahtev za čitanje rezultata pozivom funkcije `read` koja treba da vrati zbir ova dva poslednje primljena operanda. Tek kada neki klijent tako pročita taj zbir, monitor može ponovo da prihvati novi par operanada x i y , tim redom, i tako ciklično.

```
public class Computer {
    public void writeX (double val);
    public void writeY (double val);
    public double read ();
}
```

Rešenje:

3. (10 poena) Međuprocena komunikacija razmenom poruka

Na programskom jeziku Java implementirati proces koji služi kao posrednik između klijenata i servera. Klijent komunicira sa tačno jednim serverom. Posrednik od klijenta dobija adresu i port servera sa kojim želi da komunicira. Nakon toga, posrednik prosleđuje sve poruke od klijenta ka zadatom serveru i sve odgovore od servera prosleđuje klijentu. Za svaku poruku, server će poslati tačno jedan odgovor. Format poruke i odgovora je isti za sve klijente i sve servere (recimo jedna linija teksta). Obezbediti da komunikacija između jednog para klijenta i servera bude nezavisna od ostalih parova. Po pokretanju, posrednik treba da osluškuje port 6000 kako bi prihvatao zahteve od klijenata. Nije potrebno implementirati klijente i servere.

Rešenje: