
Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Operativni sistemi 2
Nastavnik: prof. dr Dragan Milićev
Odsek: Softversko inženjerstvo
Kolokvijum: Prvi, novembar 2018.
Datum: 02. 11. 2018.

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 kvantom koji se dodeljuje procesima.
- Procesima koji se uzimaju iz HP dodeljuje se vremenski kvantum 2, onima koji se uzimaju iz MP vremenski kvantum 4, a onima iz LP kvantum 8.
- Proces koji je tek postao spreman (bio je blokiran ili je tek kreiran) smešta se u prvi viši red od onoga iz koga je otišao u stanje blokade (u HP ako je otišao iz HP ili MP, u MP ako je otišao iz LP), odnosno u HP ako je prvi put aktiviran.
- Proces kome je istekao vremenski kvantum smešta se u prvi niži red od onog iz kog je uzet na izvršavanje (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):

C5, I/O, C3, I/O, C1, I/O, C7, I/O, C3

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; pored svake oznake reda dati i ceo broj koji pokazuje koliko mu je jedinica vremena preostalo u tom naletu izvršavanja (odgovor dati u obliku npr. HP5, MP4, LP3, LP2, LP1, ...).

Odgovor: _____

2. (10 poena) Međuprocesna komunikacija pomoću deljene promenljive

Korišćenjem klasičnih uslovnih promenljivih implementirati monitor koji ima dve procedure, *tick* i *tuck*, i prihvata pozive na njima na sledeći način: neki proces može izvršiti najpre proceduru *tick*, i to samo jednom, a nakon toga neki proces ili procesi mogu izvršiti proceduru *tuck* tačno dva puta, i tako ciklično (*tick-tuck-tuck-tick-tuck-tuck-tuck...*).

Rešenje:

3. (10 poena) Međuprocena komunikacija razmenom poruka

Na programskom jeziku *Java* napisati kod servera koji može da opslužuje ograničen broj klijenata. Broj klijenata koji može da opslužuje je N , gde je N konstanta. Kada se klijent konektuje na server, ukoliko je broj klijenata koje trenutno opslužuje manji od N , server šalje poruku klijentu da može da krene sa komunikacijom. Ukoliko je broj klijenata koje server opslužuje N , server klijentu šalje poruku da sačeka i šalje mu broj klijenata koji pored njega trenutno čekaju da ih server opsluži. Kada se broj klijenata koje server opslužuje smanji, server počinje da opslužuje klijente koji čekaju, po redu po kome su uspostavljali konekcije, tako što mu pošalje poruku da krene sa komunikacijom. Server osluškuje na portu 5555. Server za klijenta radi neko izračunavanje (samo navesti gde to izračunavanje treba da se nalazi).

Rešenje: