
Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Operativni sistemi 2
Nastavnik: prof. dr Dragan Milićev
Odsek: Računarska tehnika i informatika
Kolokvijum: Prvi, januar 2026.
Datum: 17. 1. 2026.

Prvi kolokvijum iz Operativnih sistema 2

Kandidat: _____

Broj indeksa: _____ *E-mail:* _____

Kolokvijum traje 90 minuta. 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

Neki sistem raspoređuje procese po prioritetima na sledeći način:

- Manja celobrojna vrednost označava viši prioritet.
- Proces ima svoj podrazumevani prioritet, kao i dinamički prioritet koji je inicijalno jednak podrazumevanom prioritetu.
- Procesi se raspoređuju na osnovu trenutne vrednosti dinamičkog prioriteta.
- Dinamički prioritet procesa se menja samo u opsegu ± 3 u odnosu na njegov podrazumevani prioritet (ne iskače izvan granica vrednosti $p+3..p-3$).
- Procesu koji je u red spremnih došao iz suspenzije prioritet se povećava (smanjuje se celobrojna vrednost) za jedan u odnosu na vrednost njegovog dinamičkog prioriteta koju je imao kada je otišao u suspenziju.
- Procesu kome je istekao vremenski odsečak smanjuje se prioritet (povećava se celobrojna vrednost) za jedan.

Posmatra se proces sa podrazumevanom vrednošću prioriteta 5, vremenskim odsečkom dužine 2 jedinice vremena i nizom susednih naleta izvršavanja (*CPU burst*), između kojih proces odlazi u suspenziju, a koji imaju redom sledeće dužine u jedinicama vremena:

5, 3, 1, 3, 1, 1

Napisati redom vrednosti dinamičkog prioriteta koje ovaj proces ima svaki put kada dobije procesor (posle svakog isteklog vremenskog odsečka ili suspenzije).

Odgovor: _____

2. (10 poena) Međuprocesna komunikacija pomoću deljene promenljive

Korišćenjem brojačkih semafora implementirati klasu `ReaderWriters` koja obezbeđuje potrebnu sinhronizaciju pristupa deljenom podatku tipa `T` od strane jednog čitaoca i više pisaca na sledeći način. Pisci upisuju podatak tipa `T` operacijom `write` (prostim kopiranjem argumenta) u isti prostor, jedan preko drugog, s tim da je svaki taj upis izolovan (atomičan, međusobno isključiv) jer `T` može biti složen tip. Čitalac dobija pokazivač na instancu tipa `T` koju može da čita pozivom operacije `read`; tu instancu na koju pokazuje vraćeni pokazivač može da čita u svojoj, potencijalno dužoj operaciji van ove operacije `read`, pristupajući joj preko tog pokazivača. Čitalac treba da čita onu instancu podatka tipa `T` koja je poslednja upisana pre poziva operacije `read` (operacija `read` treba da vrati pokazivač na tu instancu); ako nijedna dotad nije upisana, operacija `read` vraća *null*. Za ovakvu sinhronizaciju dovoljno je da u objektu klase `ReaderWriters` postoje dve instance tipa `T`, jedna u koju se upisuje, a druga koju čitalac trenutno može da čita, koje po potrebi zamenjuju uloge, kao i odgovarajuća sinhronizacija. Tip `T` ima dostupan operator dodele kopiranjem.

```
class ReaderWriters {
public:
    ReadersWriters ();
    const T* read ();
    void write (const T&);

private:
    T data[2];
    ...
};
```

Rešenje:

3. (10 poena) Međuprocesna komunikacija razmenom poruka

Data je klasa `CalculateRPC`. Deklaracija metoda klase su:

```
public class CalculateRPC {  
    public void setData(int[] array);  
    public int[] calculate();  
};
```

Implementirati potrebne klase na klijentskoj strani koje će krajnjem korisniku obezbediti identičan interfejs kao i klasa `CalculateRPC` i koje će mu omogućiti da sa udaljenog računara kroz obezbeđeni interfejs poziva operacije klase `CalculateRPC`.

Napraviti da pozivi metoda budu sinhroni. Ako metoda ne uspe da se izvrši u roku od 10 sekundi prekinuti izvršavanje metode i baciti izuzetak proizvoljnog tipa.

Server osluškuje na portu 5555.

Rešenje: