
Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Operativni sistemi 2
Nastavnik: prof. dr Dragan Milićev
Odsek: Računarska tehnika i informatika, Softversko inženjerstvo
Kolokvijum: Prvi, januar 2020.
Datum: 8. 1. 2020.

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 klasa `Scheduler`, čiji je interfejs dat dole, realizuje raspoređivač spremnih procesa. Raspoređivanje je po prioritetu. Svaki proces ima svoj prioritet u opsegu $0..MaxPri$ (niža vrednost označava viši prioritet). U skupu procesa istog prioriteta, raspoređivanje je *round-robin*.

Kako bi se sprečilo to da *cpu-bound* procesi visokog prioriteta previše oduzimaju procesor, primenjuje se raspodela vremena (*time sharing*), kao i sledeća tehnika. Skup spremnih procesa podeljen je na dva disjunktna podskupa, „aktivni“ (*active*) i oni „sa isteklim vremenom“ (*expired*). Kada se proces stavlja u skup spremnih, stavlja se u odgovarajući podskup u zavisnosti od toga odakle dolazi: ako je taj proces reaktiviran iz suspenzije (`put(pcb, false)`), stavlja se u skup *active*; ako se proces izvršavao i vraća se u skup spremnih jer mu je isteklo dodeljeno procesorsko vreme (*time slice*) (`put(pcb, true)`), stavlja se u podskup *expired*. Spreman proces uzima se iz podskupa *active*; ako je ovaj podskup prazan, ova dva podskupa, *active* i *expired*, zamenjuju uloge.

U strukturi PCB dostupna su polja `pri` (tekući prioritet procesa) i `next` za ulančavanje u liste.

Implementirati klasu `Scheduler` tako da obe operacije, `put` i `get`, imaju složenost koja ne zavisi od broja spremnih procesa, pa ni od broja procesa samo određenog prioriteta ($O(1)$).

```
struct PCB { PCB *next; Priority pri; ... };
const int MaxPri = ...;
```

```
class Scheduler {
public:
    Scheduler ();
    PCB* get ();
    void put (PCB*, bool timeExpired=false);
};
```

Rešenje:

2. (10 poena) Međuprocesna komunikacija pomoću deljene promenljive

Posmatra se sistem od tri procesa koji predstavljaju igrače i jednog deljenog objekta koji predstavlja agenta. Svaki igrač ciklično traži od agenta dva žetona različite boje. Jedan igrač uvek traži jedan žeton crvene i jedan zelene boje; drugi igrač uvek traži žeton zelene i plave boje; treći igrač uvek traži žeton crvene i plave boje. Agent ima neograničene zalihe žetona sve tri boje. Agent postavlja na sto dva žetona različite boje izabrane slučajno. Igrač koji traži baš te žetone uzima ih sa stola, a potom agent postavlja nova dva žetona i tako ciklično. Realizovati deljeni objekat koji predstavlja agenta korišćenjem klasičnih monitora i uslovnih promenljivih.

Rešenje:

3. (10 poena) Međuprocesna komunikacija razmenom poruka

Implementirati server na programskom jeziku Java na koji se prijavljuju klijenti. Prilikom prijave klijent treba da čeka sve dok ne započne prijavu N klijenata posle njega. N je statički definisana konstanta. Komunikacija sa klijentima treba da se obavlja konkurentno.

Međuprocesnu komunikaciju realizovati preko priključnica (*socket*) i razmenom poruka (*message passing*). Dozvoljeno je korišćenje koda prikazanog na vežbama (kod sa vežbi ne treba prepisivati, nego npr. reći koja klasa ili koji metod se koriste i/ili menjaju, nasleđuju, ...). Napisati samo kod servera kao rešenje zadatka. Server prihvata zahteve klijenata na portu 5555.

Rešenje: