

Prvi kolokvij iz predmeta *Operacijski sustavi*, 24. 4. 2014.

1. (2) Navesti zadaće operacijskog sustava (čemu on služi).
2. (2) Opisati operaciju skoka (npr. JMP pet1ja) i operaciju poziva potprograma (npr. CALL potprogram). Što procesor radi u tim instrukcijama (što se mijenja i kako)?
3. (2) Pristupni sklop (ne sklop za prihvrat prekida) koji se s jedne strane spaja na sabirnicu a s druge na ulazno-izlaznu napravu, ima nekoliko registara. Koji su to registri i čemu služe. (Uzeti pristupne sklopove koji se koriste kod upravljanja UI napravama mehanizmom prekida.)
4. (2) Algoritmi međusobnog isključivanja moraju zadovoljavati neke uvjete. Koji su to uvjeti (a koje Dekkerov, Lamportov i Petersonov algoritam zadovoljavaju, kao i onaj s TAS instrukcijom)?
5. (2) Opišite uređenje reda odgođenih dretvi (u jednostavnom modelu jezgre).
6. Prekidi se u nekom sustavu javljaju u 1. ms ( $P_1$ ), 5. ms ( $P_2$ ), 7. ms ( $P_3$ ) te 20. ms ( $P_4$ ). Prihvrat prekida (PP) traje 1 ms, obrada 5 ms ( $P_i$ ) te povratak iz prekida (PIP) 0,5 ms. Za vrijeme obrade prekida zabranjeno je prekidanje (novi zahtjevi čekaju da se prethodni obradi). Sustav nema sklopa za prihvrat prekida. U ispitnom lancu pristupni sklopovi se ispituju redom  $P_4, P_3, P_2$  i  $P_1$  (" $P_4$  ima najveći prioritet").

(a) (3) Prikazati stanje procesora pri obradi gornjih zahtjeva za prekid.

(b) (2) Koji je zahtjev za prekid najviše čekao na početak obrade i koliko (koja je najveća odgoda početka obrade zahtjeva za prekid uključujući i operaciju prihvata prekida)?

7. (3) Zadan je sljedeći algoritam međusobna isključivanja:

```

uđi_u_KO() {                               izadi_iz_KO() {
    dok je ( semafor != ZELENO )           semafor = ŽUTO;
        ; /* radno čekanje */             semafor = ZELENO;
    semafor = CRVENO;                       }
}
    
```

Ako je početna vrijednost zajedničke varijable semafor = ZELENO je li algoritam dobar za međusobnu isključivanje N dretvi?

8. (5) U nekom trenutku stanje sustava prikazano je tablicom:

aktivna dretva:	5
pripravne dretve:	1, 4, 2, 3
red semafora OSEM[1]:	6, 7
red naprave UI[1]:	8
red odgođenih:	(prazan red)

Idući pozivi jezgrinih funkcija koji se zbivaju iz (različitih) dretvi i prekida su:

1. Odgodi(5);
2. Postavi\_OSEM(1);
3. Prekid\_UI(1);
4. Otkucaj\_sata();

Navesti stanje jezgre nakon svakog od navedenih poziva.

9. (5) Skup od N dretvi: dretva\_x { A(); ... B(); } izvodi svoje zadatke koji se sastoje od A() i B() dijela. Međutim, nakon što sve te dretve obave svoj A() dio treba pokrenuti posao C() koji radi dretva: dretva\_y { ... C(); ... }. Nakon što je ta dretva obavila svoj posao, prijašnjih N dretvi treba nastaviti s poslom B(). Sinkronizirati dretve semaforom (zamijeniti ... potrebnim kodom za sinkronizaciju). Navesti početne vrijednosti korištenih semafora i varijabli (ako se koriste).
10. (5) Korištenjem semafora (i potrebnih dodatnih varijabli) sinkronizirati dretve tipa A, B i C koje u svom izvođenju jednom pozovu ispiši (X); gdje je X 'A' ili 'B' ili 'C', kako za koju dretvu (dretva A ispisuje 'A'). Jedini zahtjev nad izvođenjem te operacije jest da se u ispisu isto slovo ne ponovi dva puta za redom. Dretve tipa A, B i C se nasumično pojavljuju u sustavu te nakon obavljanja svog posla nestaju. Navesti početne vrijednosti korištenih semafora i varijabli (ako se koriste).