

Operacijski sustavi - drugi međuispit 2010

1

- a) Faktor iskorištenja poslužitelja može poprimiti vrijednost _____ samo u potpuno determinističkom okruženju.

1

- b) Uz zadano prosječno zadržavanje posla u sustavu i prosječni broj dolazaka poslova u jedinici vremena, možemo izračunati _____

prosječni broj poslova u sustavu

- c) Prosječni broj dolazaka u jedinici vremena za mješavinu više skupina poslova jednak je _____ prosječnog broja dolazaka pojedinih skupina poslova.

zbroju

- d) Koji uvjet nastanka potpunog zastoja ima smisla pokušati ukloniti u stvarnim uvjetima: _____

dretva drži dodijeljeno sredstvo dok čeka na dodjelu drugog sredstva

- e) U jednostavnom modelu jezgre, dretva može iz reda _____ prijeći izravno u red čekanja na binarnom semaforu.

aktivnih dretvi

2A

U nekom sustavu zahtjevi za obradu podliježu Poissonovoj razdiobi, a vrijeme obrade ima eksponencijalnu razdiobu. Mjerenjem je ustanovljeno prosječno zadržavanje posla u sustavu $\bar{T} = 0.1s$, a vjerojatnost blokiranja proizvođača koji proizvode zahtjeve za obradu uz međuspremnik s 10 pretinaca iznosi 1% (blokiranje se ne događa ako je bar jedno mjesto u međuspremniku prazno).

- a) Koliko iznosi faktor iskorištenja sustava?
- b) Koliko iznosi prosječno trajanje posluživanja jednoga posla?
- c) Koliko bi pretinaca trebao imati spremnik kako bi vjerojatnost blokiranja bila manja od 0.1%?

$$a) \quad p(i > N) = \rho^{N+1} \Rightarrow \log \rho = \frac{\log p(i > N)}{N+1} \Rightarrow \rho = 0.6579$$

$$b) \quad \bar{n} = \frac{\rho}{1-\rho} = 1.923; \quad \alpha = \frac{\bar{n}}{T} = 19.23s^{-1}; \quad \frac{1}{\beta} = \frac{\rho}{\alpha} = 0.0342s$$

$$c) \quad \rho^{M+1} < 0.001 \Rightarrow M > \frac{\log 0.001}{\log \rho} - 1 = 15.5 \Rightarrow M = 16$$

2B

U nekom sustavu zahtjevi za obradu podliježu Poissonovoj razdiobi, a vrijeme obrade ima eksponencijalnu razdiobu. Mjerenjem je ustanovljeno prosječno zadržavanje posla u sustavu $\bar{T} = 0.5s$, a vjerojatnost blokiranja proizvođača koji proizvode zahtjeve za obradu uz međuspremnik s 20 pretinaca iznosi 1% (blokiranje se ne događa ako je bar jedno mjesto u međuspremniku prazno).

- a) Koliko iznosi faktor iskorištenja sustava?
- b) Koliko iznosi prosječno trajanje posluživanja jednoga posla?
- c) Koliko bi pretinaca trebao imati spremnik kako bi vjerojatnost blokiranja bila manja od 0.1%?

$$a) \quad p(i > N) = \rho^{N+1} \Rightarrow \log \rho = \frac{\log p(i > N)}{N+1} \Rightarrow \rho = 0.8031$$

$$b) \quad \bar{n} = \frac{\rho}{1-\rho} = 4.078; \quad \alpha = \frac{\bar{n}}{T} = 8.15s^{-1}; \quad \frac{1}{\beta} = \frac{\rho}{\alpha} = 0.0985s$$

$$c) \quad \rho^{M+1} < 0.001 \Rightarrow M > \frac{\log 0.001}{\log \rho} - 1 = 30.5 \Rightarrow M = 31$$

3A

Sustav obrađuje poslove koji dolaze periodno s periodom od 16 sekundi. U prvoj periodi poslovi dolaze u trenucima 1s, 3s, 7s i 9s, a trajanja poslova su 5s, 2s, 3s i 4s, istim redoslijedom.

- a) Koliko iznosi prosječni broj poslova u sustavu?
- b) Koliko je prosječno zadržavanje poslova u redu?
- c) Koliko iznosi faktor iskorištenja sustava?

t_d	T_p	t_n	T	T_r
1	5	6	5	0
3	2	8	5	3
7	3	11	4	1
9	4	15	6	2

- a) $\bar{n} = \alpha \cdot \bar{T} = 0.25 \cdot 5 = 1.25$
- b) $6/4 = 1.5$ s
- c) $\rho = \frac{15-1}{16} = 0.875$

3B

Sustav obrađuje poslove koji dolaze periodno s periodom od 20 sekundi. U prvoj periodi poslovi dolaze u trenucima 2s, 3s, 7s i 11s, a trajanja poslova su 5s, 2s, 4s i 5s, istim redoslijedom.

- d) Koliko iznosi prosječni broj poslova u sustavu?
- e) Koliko je prosječno zadržavanje poslova u redu?
- f) Koliko iznosi faktor iskorištenja sustava?

t_d	T_p	t_n	T	T_r
2	5	7	5	0
3	2	9	6	4
7	4	13	6	2
11	5	18	7	2

- d) $\bar{n} = \alpha \cdot \bar{T} = 0.2 \cdot 6 = 1.2$
- e) $8/4 = 2$ s
- f) $\rho = \frac{18-2}{20} = 0.8$

4

U jednogprocesorskom računalu pokrenut je sustav dretvi D_1 , D_2 , D_3 i D_4 . Indeks dretve predstavlja i prioritet dretve. Najviši prioritet je 4. Svi zadaci koje obavljaju dretve su istog oblika D_x . Red pripravnih dretvi i red semafora su prioritetni. Aktivna je dretva koja je prva u redu pripravnih (nema posebnog reda aktivnih dretvi). Prije pokretanja sustava dretvi semafor S je bio zatvoren. Nakon nekog vremena sve dretve se nađu u redu semafora S . Ako se tada pozove procedura $\text{PostaviBSEM}(S)$, dretve započinju ispisivati. Kako će izgledati 4 zadnja ispisa?

rješenje:

```
Dretva Dx
{
    za(i = 1 do x)  // x - indeks dretve!
    {
        ČekajBSEM(S);
        -- mjesto A --
        piši(Px);
        PostaviBSEM(S);
        -- mjesto B --
        piši(Zx);
    }
}
```

BSEM.v	red BSEM	Pripravne	Ispis
0	4321	-	
	321	4a	P4
	21	4b 3a	Z4
	421	3a	P3
	21	4a 3b	P4
	1	4b 3b 2a	Z4
	41	3b 2a	Z3
	431	2a	P2
	31	4a 2b	P4
	1	4b 3a 2b	Z4
	41	3a 2b	P3
	1	4a 3b 2b	P4
	-	4b 3b 2b 1a	Z4
	-	3b 2b 1a	Z3
	3	2b 1a	Z2
	32	1a	P1
	2	3a 1b	P3
	-	3b 2a 1b	Z3
	-	2a 1b	P2
1	-	2b 1b	Z2
1	-	1b	Z1

5

Prilikom ulaska i izlaska iz svemirske postaje, astronauti prolaze kroz zračnu komoru koja ima dvojna vrata (s unutarnje i vanjske strane postaje) od kojih su barem jedna uvijek zatvorena. Kada u komoru uđu 3 astronauta, komora se sama automatski zatvara. Tek kada se komora zatvori, astronauti smiju započeti (de)kompresiju. Nakon što astronauti prođu kroz komoru u jednom smjeru (npr. u postaju), sljedeći prolazak kroz komoru mora biti u drugom smjeru (iz postaje). U komoru se ne smije ući prije nego svi astronauti iz suprotnog smjera prethodno ne izađu. Sinkronizirati astronaute-dretve monitorom (napisati zadane monitorske funkcije) tako da navedeni uvjeti budu zadovoljeni. Parametar `smjer` može poprimiti vrijednost 1 ili 0. Za ostvarenje monitora koristiti funkcije `lock(M)` i `unlock(M)` za zaključavanje i otključavanje monitora, `wait(M, <uvjet>)` za uvrštavanje u red uvjeta i `signal(M, <uvjet>)` ili `broadcast(M, <uvjet>)` za oslobađanje jedne odnosno svih dretvi iz reda uvjeta (jednako djelovanje kao odgovarajuće POSIX funkcije). Definirati sve potrebne podatke i njihove početne vrijednosti.

<i>(zadano)</i>	<i>moгуće rješenje</i>
<pre>Astronaut(smjer) { m_fja Udji(smjer); obavi (de)kompresiju; m_fja Izadji(smjer); }</pre>	<pre>(pocetno: stanje = 0, broj = 0; 3 reda uvjeta: 0, 1, 2) 0, 1 - cekanje u smjeru 2 - cekanje da se napuni Udji(smjer) { lock(M); dok (stanje != smjer broj > 2) wait(M, smjer); broj++; ako (broj == 3) broadcast(M, 2); inace wait(M, 2); unlock(M); } Izadji(smjer) { lock(M); broj--; ako (broj == 0) { stanje = 1-smjer; broadcast(M, 1-smjer); } unlock(M); }</pre>