

# 1. kolokvij iz predmeta *Operacijski sustavi*, 20. 11. 2017.

PISATI ČITKO – nečitki odgovori su KRIVI!

Ime i prezime studenta

1. Čemu služi operacijski sustav? Koje su njegove osnovne zadaće?

olakšava korištenje računala  
skriva nepotrebne detalje  
omogućuje više zadaćnost ...

2. Što je to kontekst dretve? Od čega se sastoji? Tko s njime rukuje/upravlja?

sadržaji registara procesora za vrijeme izvođenja dretve  
pri prekidu kontekst spremnik/obnavlja jezgrina funkcija

3. Kada neka naprava dobavlja puno podataka koje treba pohraniti u radni spremnik, koji je najprikladniji način upravljanja s tom napravom? Obrazložiti.

korištenjem pristupna sklopa za izravni pristup spremniku (DMA)  
on najmanje koristi kućanske poslove u usporedbi s načinom rada koji  
koristi prekide

4. Navesti vrijednosti u registru PC **nakon** izvođenja instrukcija na adresama 1, 3, 7, 150.

1:	MOV R0, \$1	PC = 2
2:	MOV R1, \$1	
3:	CALL \$100	PC = 100
4:	ADD R1, R1, R0	
5:	SUB R0, R1, R0	
6:	CMP R1, 100	
7:	BLT \$3 ; ako je R1 < 100 skoči na 3	PC = 3
	...	
100:	(ispis sadržaja u R0)	
	...	
150:	RET	PC = 4

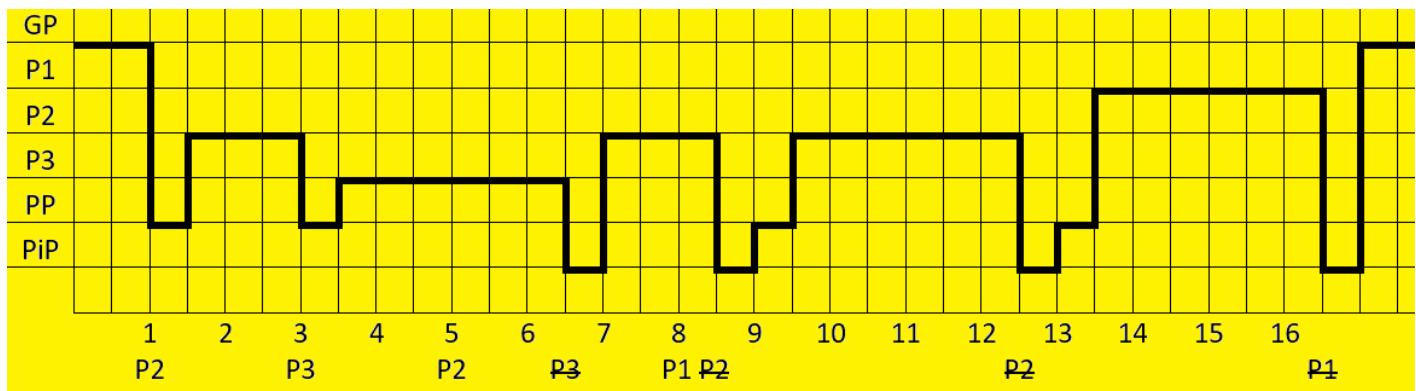
5. U nekom sustavu redovi pripravnih dretvi, kao i redovi semafora su uređeni prema prioritetu. Prioritet dretve određen je njenim indeksom, gdje veći broj označava veći prioritet.

Red	početno	nakon a)	nakon b)	nakon c)	nakon d)	nakon e)	nakon f)
Aktivna_D:	4	6	7	6	4	4	4
Pripravne_D:	3 2	4 3 2	6 4 3 2	4 3 2	3 2	3 2	3 2
Odgodjene_D:	1 <sup>2</sup> , 5 <sup>1</sup> , 7 <sup>2</sup>	1 <sup>2</sup> , 5 <sup>1</sup> , 7 <sup>2</sup>	1 <sup>1</sup> , 5 <sup>1</sup> , 7 <sup>2</sup>	1 <sup>1</sup> , 5 <sup>1</sup> , 7 <sup>2</sup>			
UI[1]:	6	-	-	-	-	-	-
OSEM[1]:	7 (.v=0)	7 (.v=0)	- (.v=0)	- (.v=0)	6 (.v=0)	6 (.v=0)	6 (.v=0)
BSEM[1]:	- (.v=1)	- (.v=1)	- (.v=1)	- (.v=1)	- (.v=1)	- (.v=1)	- (.v=0)

Pokazati stanje sustava ako se nakon početnog stanja dogode pozivi redom:

- a) Prekid\_UI(1), b) PostaviOSEM(1), c) Odgodi(5), d) ČekajOSEM(1),  
e) Otkucaj\_sata(), f) ČekajBSEM(1).

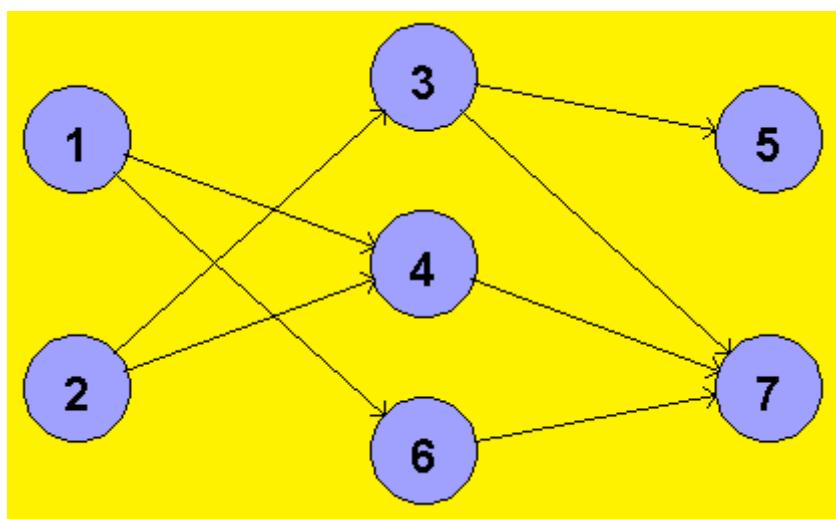
6. U sustavu koji IMA sklop za prihvat prekida pojavljuju se zahtjevi: P1 u 8. ms, P2 u 1. ms i 5. ms te P3 u 3. ms. Indeks zadatka označava prioritet (P3 ima najveći). Ukoliko prihvat prekida (PP) traje 0,5 ms, povratak iz prekida (PIP) traje 0,5 ms, a obrade prekida traju po 3 ms, pokazati rad procesora u glavnom programu (GP), obradama prekida (Pi) i procedurama za prihvatanje prekida (PP) i povratak iz prekida (PIP). Navesti stanje registara sklopa za prihvatanje prekida u t = 6. ms.



Stanje registara sklopa: KZ=010 TP=110

7. Sustav zadataka zadan je u obliku lanca:  $Z_1 \rightarrow Z_2 \rightarrow Z_3 \rightarrow Z_4 \rightarrow Z_5 \rightarrow Z_6 \rightarrow Z_7$ . Uzimajući u obzir domene i kodmene pojedinih zadataka odrediti maksimalno paralelan sustav zadataka.

	$Z_1$	$Z_2$	$Z_3$	$Z_4$	$Z_5$	$Z_6$	$Z_7$
$M_1$		K		K			
$M_2$	D		D				K
$M_3$			D		D		
$M_4$	K			D		D	K
$M_5$		D	K		K		
$M_6$	K					K	



8. Tri dretve proizvođača komuniciraju s dretvom potrošača preko ograničenog međuspremnika M veličine N=10 poruka. Sinkronizirati te dretve korištenjem semafora te dodatnih varijabli – napisati pseudokod dretvi proizvođač() i potrošač(). Navesti početne vrijednosti semafora i varijabli.

```
proizvođač
{
    ponavljaј {
        P = stvari poruku ();
        Čekaj_OSEM(2);
        Čekaj_BSEM(1);
        M[ULAZ] = P;
        ULAZ = (ULAZ + 1) MOD N;
        Postavi_BSEM(1);
        Postavi_OSEM(1);
    }
    do zauvijek
}
```

```
potrošač
{
    ponavljaј {
        Čekaj_OSEM(1);
        P = M[IZLAZ];
        IZLAZ = (IZLAZ + 1) MOD N;
        Postavi_OSEM(2);
        obradi poruku (P);
    }
    do zauvijek
}
```

Početne vrijednosti:	ULAZ=IZLAZ=0
	BSEM[1].v = 1
	OSEM[1].v = 0
	OSEM[1].v = 10

9. Radi promjene ljetnih u zimske gume ljudi dolaze do vulkanizera. Radnja se kod nekog vulkanizera sastoji od postolja, koje automatski podiže auto čim auto dođe iznad njega, te automatski spušta kad su gume zamijenjene. U radnji postoje četiri radnika koji mijenjaju svaki po jednu gumu. Simulirati sustav dretvama automobil() i radnik() tako da radnici odmah počnu raditi (operacijom zamijeni\_gumu) čim je auto došlo na postolje (operacijom popni\_se\_na\_postolje), te da auto odlazi (operacijom odvezi\_se\_s\_postolja) čim su sve gume postavljene. Čim jedan auto ode, drugi može na postolje. Koristiti binarne/opće semafore, prema potrebi.

```
automobil
{
    Čekaj_BSEM(postolje)
    popni_se_na_postolje
    za i = 1 do 4 radi
        PostaviOSEM(kreni)
    za i = 1 do 4 radi
        ČekajOSEM(gotovo)
    odvezi_se_s_postolja
    Postavi_BSEM(postolje)
}
```

```
radnik
{
    ponavljaј {
        Čekaj_OSEM(kreni)
        zamijeni_gumu
        Postavi_OSEM(gotovo)
    }
}
```

Početno:	BSEM[postolje].v = 1
	OSEM[kreni].v = 0
	OSEM[gotovo].v = 0

10. U stanici za tehnički pregled rade tri ispitivača koji provjeravaju ispravnost automobila: 1. kočnica, 2. ovjesa, 3. rasvjete i 4. ispušne plinove. Svaki ispitivač zaprima auto u svojoj traci na kojoj se nalaze svi potrebni mjerni instrumenti, osim uređaja za provjeru ispušnih plinova koji dijele svo troje. Provjera za točke 1, 2 i 3 ide tim redom, dok se provjera za ispušne plinove može obaviti bilo kada, prije/poslije bilo koje druge provjere ako je uređaj trenutno slobodan. Simulirati sustav cikličkim dretvama ispitivač(i) koje ciklički provjeravaju automobile (u svakoj iteraciji idući). U pseudokodu uzimanje idućeg auta označiti s idući, a provjere označiti sa: kočnice, ovjes, rasvjeta, ispuh. Prepostaviti da će svaki ispitivač prije nego ide na iduću točku provjeriti je li uređaj za provjeru ispušnih plinova slobodan i ako jest odabratи taj test kao idući (i uzeti uređaj). Koristiti binarne semafore i dodatno potrebne varijable. Preporuka je da se napravi funkcija probaj\_ispitati\_ispuh(x) koja će ispitati ispušne plinove ako je uređaj slobodan i varijabla x ima vrijednost 1. U protivnom neće ispitati ispušne plinove. Kao povratnu vrijednost vraća 0 ako su ispušni plinovi ispitani ili je x=0 te 1 inače. Takva bi se funkcija mogla pozvati iz pojedine dretve ispitivača sa: t[i]=probaj\_ispitati\_ispušne\_plinove(t[i]) i to više puta uz početnu vrijednost t[i]=1 (za svaki novi auto).

```

ispitivač(i)
{
    ponavljam {
        idući
        t[i] = probaj_ispitati_ispuh(1)
        kočnice
        t[i] = probaj_ispitati_ispuh(t[i])
        ovjes
        t[i] = probaj_ispitati_ispuh(t[i])
        rasvjeta
        dok je t[i] == 1 radi {
            t[i] = probaj_ispitati_ispuh(t[i])
            odgodi(1)
        }
    }
}
}

```

Rješenje:

Početno: BSEM[KO].v = 1, instrument = SLOBODAN

```

probaj_ispitati_ispuh(x)
{
    ako je x == 1 tada {
        ČekajBSEM(KO)
        ako je instrument == SLOBODAN tada {
            instrument = ZAUZET
            x = 0
        }
        PostaviBSEM(KO)
        ako je x == 0 tada {
            ispuh
            ČekajBSEM(KO)
            instrument = SLOBODAN
            PostaviBSEM(KO)
        }
    }
    vrati x
}

```