

1. (2) Čemu služi operacijski sustav?

olakšavanje korištenja računala (skriva detalje)

omogućava učinkovito korištenje svih dijelova računala (ima upravljačke programe)

omogućava višeprogramski rad – najbitnija uloga, povećava učinkovitost sustava

2. (2) Opisati operacije koje procesor izvodi pri izvođenju instrukcije `RET` (povratak iz potprograma) ako je stanje sustava sljedeće: `PC=1000`, `SP=3000`, na stogu je vrijednost `2000`.

`RET` - vrati se iz potprograma: vrijednost s vrha stoga stavi u `PC`

za zadani primjer, nakon `RET`:

`PC = 2000` (i `SP = 3004`)

3. (2) Što je to kontekst dretve i zašto ga je potrebno pohraniti pri zamjeni jedne dretve drugom?

kontekst dretve čine svi registri procesora (`R0-RX`, `PC`, `SR`, `SP`)

pri zamjeni jedne dretve drugom, nakon zaustavljanja rada dretve (npr. prekidom) sprema se kontekst dretve koja je zaustavljena (i opisnik te dretve) te se iz opisnika druge dretve obnavlja kontekst u registre procesora

4. (2) Opisati postupak prihvata prekida UI naprave **od strane procesora**, od trenutka pojave zahtjeva za prekid na prekidnom ulazu procesora do poziva prekidna potprograma u sustavu bez sklopa za prihvata prekida ako se trenutno izvodi korisnička dretva (prekidanje je dozvoljeno)

a. pojava zahtjeva za prekid

b. dovrši izvođenje trenutne instrukcije

1. zabrani prekidanje

2. prebaci se u prekidni način rada i aktiviraj prekidnu kazaljku stoga

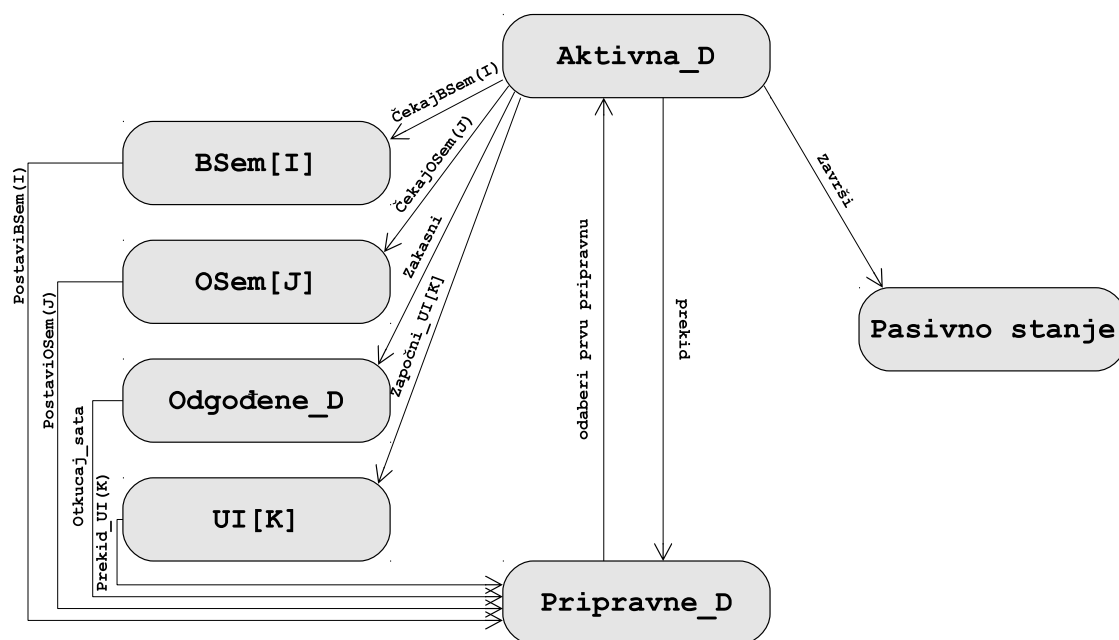
3. spremi `PC` i `SR` na stog

4. u `PC` stavi adresu prekidnog potprograma

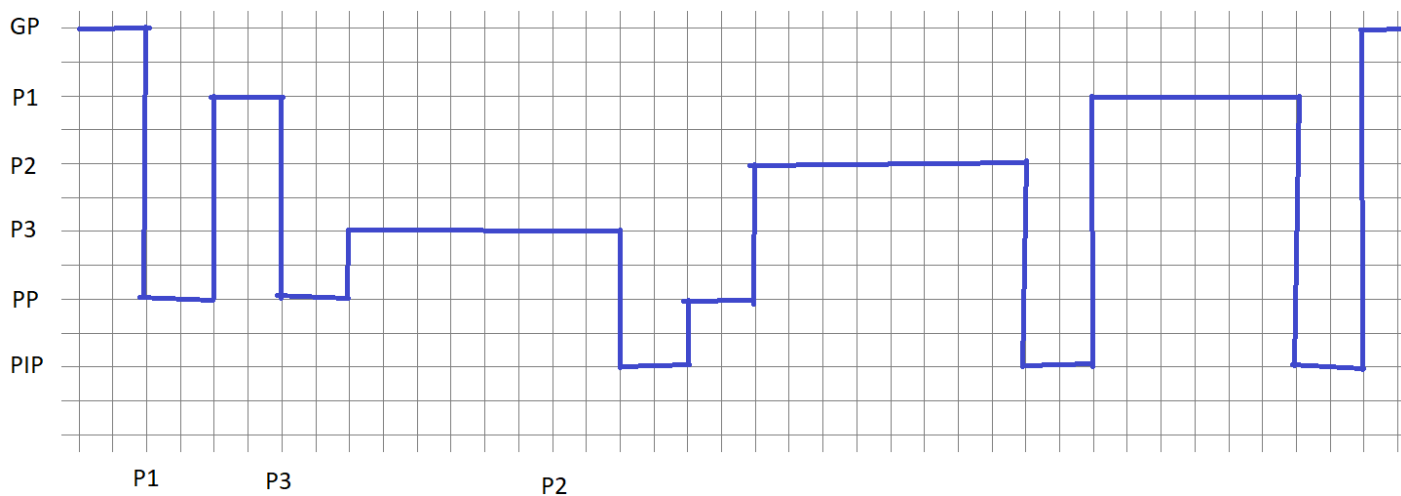
5. (2) Koji je osnovni nedostatak Dekkerova i Lamportova algoritma (zašto umjesto njih koristimo binarne semafore)?

radno čekanje

6. (4) Skicirati stanja dretvi te jezgrine funkcije koje mogu uzrokovati promjene stanja.



7. (4) U sustavu koji ima sklop za prihvat prekida pojavljuju se zahtjevi: P1 u 1. ms, P2 u 7. ms te P3 u 3. ms. Indeks zadatka označava prioritet prekida (P3 ima najveći). Ukoliko prihvat prekida (PP) traje 1 ms, povratak iz prekida (PIP) traje 1 ms, a obrade prekida traju po 4 ms, pokazati rad procesora u glavnom programu (GP), obradama prekida (Pi) i procedurama za prihvat prekida (PP) i povratak iz prekida (PIP).

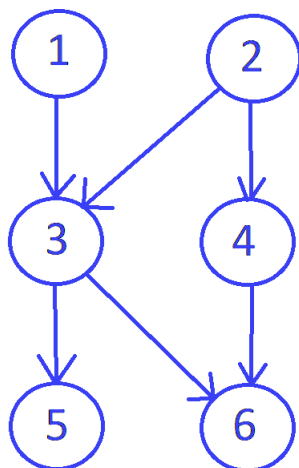


8. (4) U nekom sustavu redovi pripravnih dretvi i redovi semafora su uređeni prema prioritetu. Prioritet dretve određen je njenim indeksom, gdje veći broj označava veći prioritet. Početno stanje sustava prikazano je u tablici u stupcu "početno". Pokazati promjene za pozive jezgre redom zadane u ostalim stupcima u tablici.

stanje nakon:	početno	Odgodi(5)	PostaviBSEM(1)	PrekidUI(1)	OtkucajSata()	ČekajOSEM(1)
AktivnaD:	6	4	4	5	5	4
PripravneD:	4 2	2	2	4 2	4 3 2	3 2
OdgođeneD:	$3^1 1^2$	$3^1 1^2 6^2$	$3^1 1^2 6^2$	$3^1 1^2 6^2$	$1^2 6^2$	$1^2 6^2$
UI[1]:	5	5	5	-	-	-
OSEM[1]:	7 (.v=0)	7 (.v=0)	7 (.v=0)	7 (.v=0)	7 (.v=0)	7 5 (.v=0)
BSEM[1]:	- (.v=0)	- (.v=0)	- (.v=1)	- (.v=1)	- (.v=1)	- (.v=1)

9. (2) Sustav zadataka $Z_1 - Z_6$ će dati ispravno rješenje ako se izvodi redom $Z_1 \rightarrow Z_2 \rightarrow Z_3 \rightarrow Z_4 \rightarrow Z_5 \rightarrow Z_6$. Domene (D) i kodomene (K) zadataka prikazane su tablicom ispod. Na osnovu tablice izgraditi maksimalno paralelni sustav zadataka i prikazati ga usmjerenim grafom.

	Z_1	Z_2	Z_3	Z_4	Z_5	Z_6
M_1	D		K		K	
M_2		D		K		D
M_3	K				D	
M_4		K	D	D		K



10. (4) Obrada objekta upravljana je računalom a sastoji se od radnji: pomak trake, rad na objektu i ispitivanje ispravnosti. Ispred strojeva za obradu i ispitivanje se nalaze dva objekta: svaki stroj prvo obrađuje jedan tek pa onda drugi (ali različiti strojevi mogu paralelno raditi na različitim objektima ako je jedan završio s prvim objektom). Pomakom trake upravlja dretva pomak (koja pomiče traku s `pomakni()`, radom na objektu dvije dretve `obrada_1` i `obrada_2` (prva poziva `obradi_11()` za prvi objekt ispred stroja te `obradi_12()` za drugi objekt; druga ddretva slično `obradi_21()` i `obradi_22()`), te dvije dretve `ispitaj_1` i `ispitaj_2` na ispitivanju (`ispitaj_11()/ispitaj_12()` i `ispitaj_21()/ispitaj_22()`). Ispitivanje pojedinog objekta se može obaviti tek nakon što je objekt obrađen: `ispitaj_11()` nakon `obradi_11()`, `ispitaj_12()` nakon `obradi_12()` i slično). Pomak trake treba obavljati nakon što je i ispitivanje oba objekta ispred strojeva gotovo. Pomakom se ispred strojeva postavljaju nova dva objekta. Pretpostaviti da su objekti već na traci, ali ne ispred strojeva – prvo treba napraviti pomak trake a tek onda krenuti s radom, kao na slici. Sinkronizirati dretve semaforima.

```

[strojevi za obradu]
=====
[objekt-2][objekt-1][ prazno ][ prazno ]   traka
=====
[strojevi za ispitivanje]

dretva pomak() {
    ponavljaaj {
        pomakni_traku()
        PostaviBSEM(R1)
        PostaviBSEM(R2)
        ČekajOSEM(G)
    }
}

dretva obrada_1() {
    ponavljaaj {
        ČekajBSEM(R1)
        obradi_1_1()
        PostaviBSEM(I1_1)
        obradi_1_2()
        PostaviBSEM(I1_2)
    }
}

dretva ispitaj_1() {
    ponavljaaj {
        ČekajBSEM(I1_1)
        ispitaj_11()
        ČekajBSEM(I1_2)
        ispitaj_12()
        PostaviOSEM(G)
    }
}

dretva obrada_2() {
    ponavljaaj {
        ČekajBSEM(R2)
        obradi_2_1()
        PostaviBSEM(I2_1)
        obradi_2_2()
        PostaviBSEM(I2_2)
    }
}

dretva ispitaj_2() {
    ponavljaaj {
        ČekajBSEM(I2_1)
        ispitaj_21()
        ČekajBSEM(I2_2)
        ispitaj_22()
        PostaviOSEM(G)
    }
}

```

Početne vrijednosti korištenih semafora:

```

BSEM[R1].v = BSEM[R2].v = 0
BSEM[I1_1].v = BSEM[I1_2].v = 0
BSEM[I2_1].v = BSEM[I2_2].v = 0
OSEM[G].v = 0

```