Primjer s jezgrinim funkcijama

```
D1 {
   ponavljaj {
      PostaviOSEM(1)
      Zakasni(3)
      ZapočniUI(1)
   }
}
```

```
D2 {
   ponavljaj {
      ČekajOSEM(1)
      ZapočniUI(2)
      PostaviBSEM(2)
   }
}
```

```
D3 {
   ponavljaj {
      ČekajBSEM(2)
      Zakasni(5)
   }
}
```

```
D4 {
   ponavljaj {
      radi nešto
   }
}
```

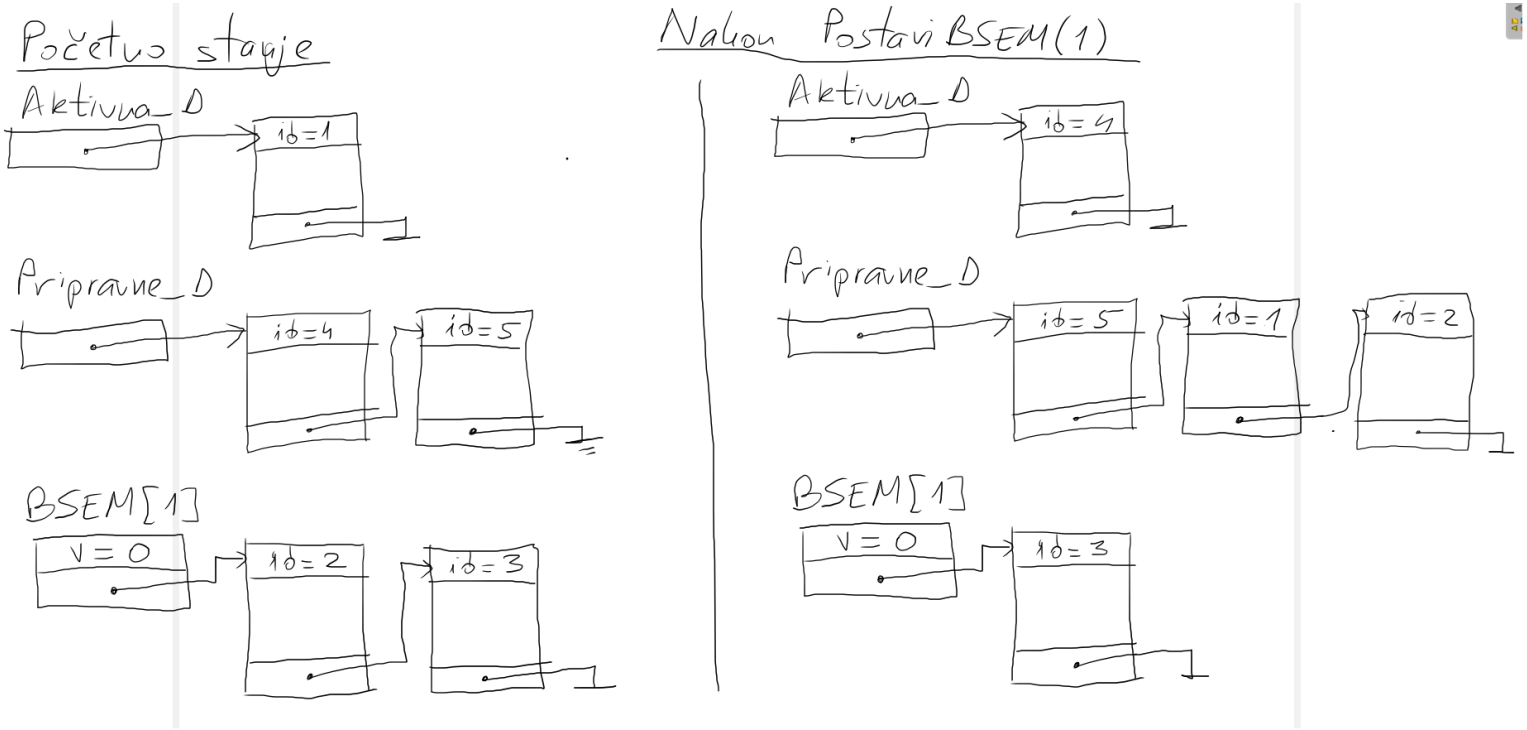
Početne vrijednosti svi semafora su 0, trajanje UI operacija su dva otkucaja sata.
Redovi pripravnih i semafora su prioritetni redovi. Indeks dretve predstavlja njen prioritet, manji indeks označava veći prioritet (u ovom primjeru).

Rješenje (tablično)

	početno stanje												
Aktivna_D	**1**	**1**	**2**	**2**	**3**	**4**	**4**	**4**	**2**	**2**	**3**	**4**	**1**
Pripravne_D	**2 3 4**	**2 3 4**	**3 4**	**3 4**	**4**	**-**	**-**	**-**	**4**	**3 4**	**4**	**-**	**4**
Odgođene_D	**-**	**-**	1^3	1^3	1^3	1^3	1^2	1^1	1^1	1^1	1^1	$1^1\ 3^4$	3^4
red UI[1]	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**
red UI[2]	**-**	**-**	**-**	**-**	**2**	**2**	**2**	**2**	**-**	**-**	**-**	**-**	**-**
OSEM[1]	**- (.v=0)**	**- (.v=1)**	**- (.v=1)**	**- (.v=0)**	**- (.v=0)**	**- (.v=0)**	**- (.v=0)**	**- (.v=0)**	**- (.v=0)**	**- (.v=0)**	**2 (.v=0)**	**2 (.v=0)**	**2 (.v=0)**
BSEM[2]	**- (.v=0)**	**- (.v=0)**	**- (.v=0)**	**- (.v=0)**	**- (.v=0)**	**3 (.v=0)**	**3 (.v=0)**	**3 (.v=0)**	**3 (.v=0)**	**- (.v=0)**	**(.v=0)**	**(.v=0)**	**(.v=0)**
iduća jezgrina funkcija	**D1: PostaviOSEM(1)**	**D1: Zakasni(3)**	**D2: ČekajOSEM(1)**	**D2 ZapočniUI(2)**	**D3 ČekajBSEM(2)**	**prekid_sata**	**prekid_sata**	**PrekidUI(2)**	**D2 PostaviBSEM(2)**	**D2 ČekajOSEM(1)**	**D3 Zakasni(5)**	**prekid_sata**	**D1 ZapočniUI(1)**

Zadatak 5.4: Strukture podataka jezgre

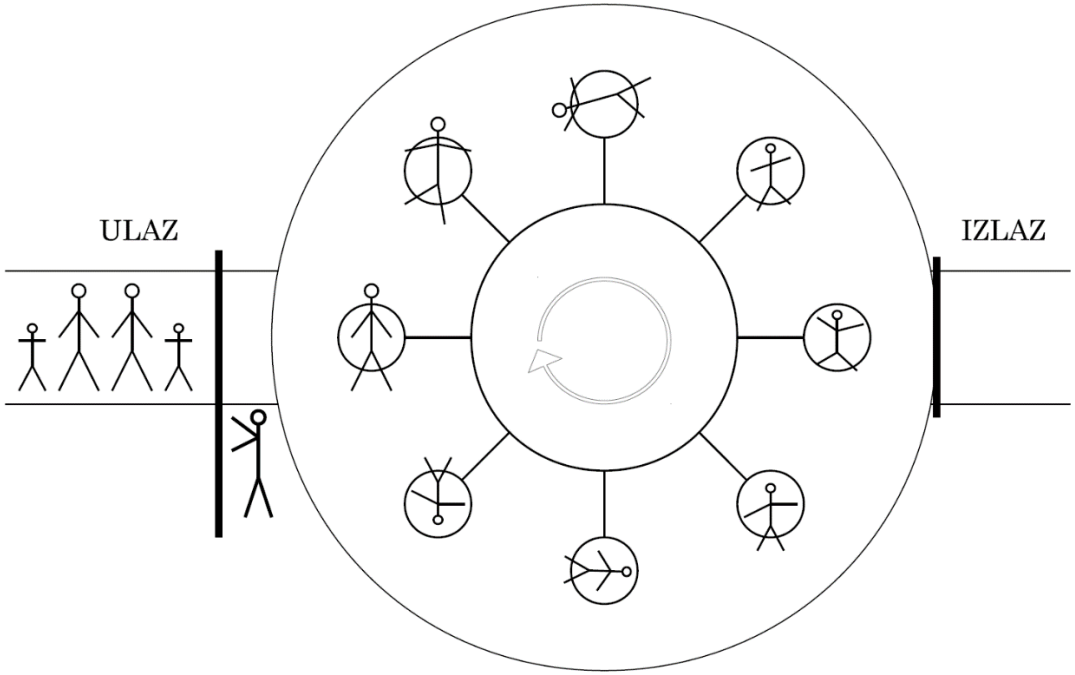
U promatranom trenutku stanje sustava je sljedeće: dretva 1 je aktivna; dretve 2 i 3 su u redu binarnog semafora 1 te dretve 4 i 5 su u redu pripravnih dretvi. Ako tada dretva 1 pozove jezgrinu funkciju PostaviBSem(1), kako će izgledati struktura podataka jezgre NAKON poziva? Svi redovi organizirani su po redu prispijeća (FIFO).



Prije poziva PostaviBSem(1):		**Međurezultati:**		**Poslije poziva PostaviBSem(1):**	
Aktivna_D:	1	-	-	Aktivna_D:	4
Red Pripravne_D:	4, 5	4,5,1	4,5,1,2	Red Pripravne_D:	5, 1, 2
Red BSEM[1]:	2, 3	2,3	3	Red BSEM[1]:	3

Zadatak 5.5: Vrtuljak

Modelirati vrtuljak s dva tipa dretvi: dretvama *posjetitelj* (koje predstavljaju posjetitelje koji žele na vožnju) te dretvom *vrtuljak*. Dretvama *posjetitelj* se ne smije dozvoliti ukrcati na vrtuljak kada više nema praznih mjesta (koristiti pomoćnu varijablu BR_MJESTA), a pokrenuti vrtuljak tek kada je pun. Za sinkronizaciju koristiti opće semafore i dodatne varijable.



```
Dretva posjetitelj() {
   ČekajOSem(vrtuljak);
   uđi i sjedi;
   PostaviOSem(sjeo);

   ČekajOSem(kraj);
   ustani i izađi;
   PostaviOSem(izašao);
}
```

```
Dretva vrtuljak() {
   dok je(1) {
      for (i = 1 do BR_MJESTA)
         PostaviOSem(vrtuljak);
      for (i = 1 do BR_MJESTA)
         ČekajOSem(sjeo);

      pokreni vrtuljak;
      zaustavi vrtuljak;

      for (i = 1 do BR_MJESTA)
         PostaviOSem(kraj);
      for (i = 1 do BR_MJESTA)
         ČekajOSem(izašao);
   }
}
```

Početne vrijednosti: svi su semafori neprolazni i imaju vrijednost 0