

Dekkerov postupak međusobnog isključivanja

Uz pomoć Dekkerovog algoritma moguće je sinkronizirati dvije dretve ili dva procesa koji primjerice izgledaju ovako:

```
proces proc(i){           /* i  [0,1] */
    ponavljaj{
        uđi u kritični odsječak
        K.O.;
        izađi iz kritičnog odsjeka
        N.K.O.;
    }dok je zadovoljen(uvjet);
}
```

Dekkerov algoritam:

zajedničke varijable: PRAVO, ZASTAVICA[0..1]

```
funkcija uđi_u_kritični_odsječak(i,j)
{
    ZASTAVICA[i] = 1
    dok_je ZASTAVICA[j]<>0 čini {
        ako_je PRAVO==j onda {
            ZASTAVICA[i] = 0
            dok_je PRAVO==j čini {
                ništa
            }
            ZASTAVICA[i] = 1
        }
    }
}
```

```
funkcija izađi_iz_kritičnog_odsjeka(i,j)
{
    PRAVO = j
    ZASTAVICA[i] = 0
}
```

Dodatne upute:

Ako se program rješava s procesima tada treba zajedničke varijable tako organizirati da se prostor za njih zauzme odjednom i podijeli među njima. Ovo je nužno zbog ograničenog broja segmenata i velikog broja korisnika.

Ovisno o opterećenju računala i broju procesa koji se pokreću, a da bi se vidjele razlike prilikom izvođenja programa može se usporiti izvršavanje programa sa:

```
sleep(1);
```

na kraju kritičnog odsječka (K.O.).

[Problemi zbog izvođenja instrukcija "preko reda"](#)