

Problem pušača cigareta

„Problem pušača cigareta“ predstavio je Patil kako bi ukazao na problem sinkronizacije korištenjem semafora. U ovom radu koristi se isti problem, ali različito rješenje radi ilustracije nekih problema sa semaforima.

U ovom slučaju sustav se sastoji od tri dretve koje predstavljaju pušače te jedne dretve trgovca. Svaki od „pušača“ ima kod sebe u neograničenoj količini jedan od tri sastojka potrebnih da se napravi i zapali cigareta: papir, duhan ili šibice. Svaki pušač ima različiti sastojak, tj. jedan ima samo papir, drugi duhan i treći samo šibice. Trgovac ima sve tri komponente u neograničenim količinama. Trgovac nasumice odabire dvije različite komponente, stavlja ih na stol te signalizira pušačima da su sastojci na stolu. Jedino pušač kojemu nedostaju sastojci stavljeni na stol, smije uzeti oba sastojka, signalizirati trgovcu te potom saviti cigaretu i popušiti ju. Trgovac tada stavlja nova dva sastojka na stol i postupak se ponavlja.

Jedno od mogućih rješenja sinkronizacije prikazano je u nastavku. Svaki pušač čeka na svom binarnom semaforu, a trgovac pri stavljanju novih sastojaka postavlja sve semafore. Pušač će tada samo jednom pogledati što ima na stolu, dok će u slijedećem pokušaju biti blokiran dok god trgovac ne stavi nove sastojke.

```
dretva Trgovac() {
    ponavlja {
        (s1, s2) = nasumice_odaberi_dva_razlicita_sastojka
        ČekajBSEM(K0)
        stavi_sastojke_na_stol(s1, s2)
        PostaviBSem(K0)
        PostaviBSem(p1)
        PostaviBSem(p2)
        PostaviBSem(p3)
        ČekajBSem(stol_prazan)
    } do ZAUVIJEK
}

dretva Pušač(p) {
    (r1, r2) = sastojci_koje_pušač_nema(p)
    ponavlja {
        ČekajBSem(p)
        ČekajBSEM(K0)
        ako (na_stolu_sastojci(r1, r2) = DA ) {
            uzmi_sastojke(r1, r2)
            PostaviBSem(K0)
            PostaviBSem(stol_prazan)
            smotaj_zapali_puši()
        }
        inače {
```

```
        PostaviBSem(K0)
    }
} do ZAUVIJEK
}
```

Zadatak

Osvariti sinkronizaciju 3 pušača i jednog trgovca korištenjem procesa za simulaciju svakog sudionika zasebno, te semafora korištenjem funkcija: `sem_init`, `sem_post` i `sem_wait`

Primjer ispisa pokretanja programa

```
# ./a.out
Pusac 1: ima papir
Pusac 2: ima duhan
Pusac 3: ima sibice

Trgovac stavlja: papir i duhan
Pusac 3: uzima sastojke i ...

Trgovac stavlja: sibice i duhan
Pusac 1: uzima sastojke i ...

^C
#
```