

# Programska podrška mjernih i procesnih sustava

Branko Jeren i Predrag Pale

Fakultet elektrotehnike i računarstva  
Zavod za elektroničke sustave i obradbu signala

---

---

---

---

---

---

---

---

## Deadlocks

proces i u zastoju

---

---

---

---

---

---

---

---

## Deadlock

- zastoj =  
situacija u kojoj  
dva (ili više) procesa  
čekaju jedan na drugoga
- procesi se sami ne mogu osloboditi zastoja
- dešava se kada  
proces ima isključivo pravo  
na neki objekt, sredstvo

---

---

---

---

---

---

---

---

## Resursi

LS&S

- u ovoj diskusiji
- resurs je sredstvo, objekt dodijeljen procesu
- resursi mogu biti:
  - zaposjedljivi (preemptable)
    - drugi proces ga može preuzeti bez štete
    - npr. memorija
  - nezaposjedljivi (nonpreemptable)
    - preuzimanje bi stvorilo štetu (zbrku)
    - npr. printer
- potencijalni zastoј na preemptable resursu
  - može se razriješiti realokacijom resursa
- pravi su zastoјi na nonpreemptable resursu

B.Jeren i P.Pale: Programska podrška mjernih i procesnih sustava

Deadlocks © LS&S 1998., 1999.

---

---

---

---

---

---

---

---

---

---

LS&S

*Skup procesa je u zastoju  
kad svaki proces u skupu  
čeka na događaj  
koji može izazvati  
samo neki proces iz istog skupa.*

B.Jeren i P.Pale: Programska podrška mjernih i procesnih sustava

Deadlocks © LS&S 1998., 1999.

---

---

---

---

---

---

---

---

---

---

## Kada se dešava zastoј ?

LS&S

(Coffman et al. 1971)

- uzajamna isključivost
  - svaki je resurs
    - ili slobodan
    - ili dodijeljen točno jednom procesu
- “hold & wait”
  - proces koji je dobio resurs može zatražiti još jedan
- nonpreemptable resurs
  - dodijeljeni resurs ne može se nasilno oteti
- cirkularno čekanje
  - dva ili više procesa čekaju na resurs kojeg drži netko drugi u lancu

B.Jeren i P.Pale: Programska podrška mjernih i procesnih sustava

Deadlocks © LS&S 1998., 1999.

---

---

---

---

---

---

---


---

---



---



## Grafički prikaz zastoja (Holt 1972)

- usmjereni grafovi

- resurs 

- proces 

- resurs dodijeljen  → 

- resurs tražen  → 

---

---

---

---

---

---

---

---

## Korisno za otkrivanje zastoja

Proces A drži R i želi S

Proces B želi T

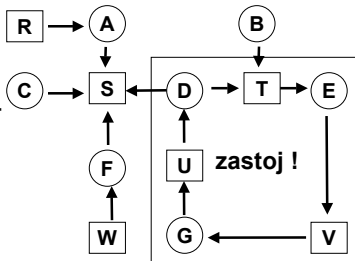
Proces C želi S

Proces D drži U i želi S i T

Proces E drži T i želi V

Proces F drži W i želi S

Proces G drži V i želi U




---

---

---

---

---

---

---

---

## Načini rješavanja zastoja

- ignoriranje (mogućnosti dolaženja do) zastoja
- otkrivanje i razbijanje (oporavak)
- dinamičko izbjegavanje
  - pažljivim dodjeljivanjem resursa
- sprečavanje
  - strukturnim negiranjem jednog od uvjeta nastajanja

---

---

---

---

---

---

---

---

## Strategija noja: ignoriranje

- ključna pitanja:
  - vjerojatnost zastoja
  - učestalost zastoja
- ako su manje od ostalih problematičnih događaja u sustavu
  - = sasvim prihvatljiva strategija

---

---

---

---

---

---

---

---

## Otkrivanje i oporavak

- sustav dopušta nastanak zastoja
- nužno je da ga može otkriti
- mora imati mehanizam prekidanja zastoja

---

---

---

---

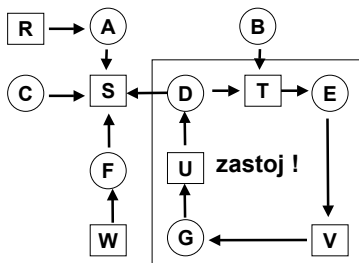
---

---

---

---

## Otkrivanje zastoja




---

---

---

---

---

---

---

---

## Oporavak zastoja

- zaposjedanjem
  - oduzeti resurs i privremeno ga dati drugom procesu
  - mora se raditi "ručno"
  - često nemoguće
- "film unatrag"
  - periodički vodimo "dnevnik stanja"
  - kad se zastoj desi, potražimo problematični resurs i proces
  - odvrtno proces unatrag, do stanja prije uzimanja resursa
- "ubijanjem" procesa
  - time će proces osloboditi i resurs koji je (možda) uzrokom
  - treba izabrati proces kojem ne smeta ponovljeno izvođenje

---

---

---

---

---

---

---

---

---

---

## Izbjegavanje zastoja

- sustav mora procijeniti je li sigurno dodijeliti resurs
- pažljivo dodjeljivanje resursa
  - zahtjeva informacije unaprijed
- koncept sigurnih i nesigurnih stanja

---

---

---

---

---

---

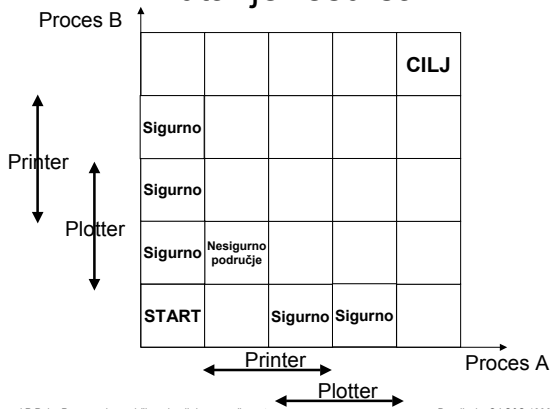
---

---

---

---

## Putanje resursa




---

---

---

---

---

---

---

---

---

---

## Bankrev algoritam

- prije dodjele resursa, provjera:
- da li preostalim resursima može u potpunosti zadovoljiti barem jedan proces

---

---

---

---

---

---

---

---

## Sprečavanje zastoja

- potrebno je “napasti” (ukloniti) barem jedan od 4 uvjeta za nastajanje zastoja
- time su oni strukturalno nemogući

---

---

---

---

---

---

---

---

## Napad na uzajamnu isključivost

- kad je ne bi bilo, ne bi bilo zastoja
- ali ona nam treba radi “race condition”
- neki uređaji (printer) se mogu dijeliti, ali preko posebne aplikacije
  - daemon, spooler
- nije uvijek moguće

---

---

---

---

---

---

---

---

## Napad na "hold & wait"

- ako proces zna što će mu sve trebati
  - neka zatraži prije izvođenja
- ne znaju to svi unaprijed
- podvarijanta je da kod zahtjeva za novi resurs
  - proces privremeno otpusti sve resurse koje drži
  - i zatraži sve resurse koje dalje treba
- ponekad neučinkovito

---

---

---

---

---

---

---

---

---

---

## Napad na nezaposjedljivost

- na primjer, oduzeti printer procesu koji je do prekida (scheduling) na njega pisao
- uglavnom nemoguće

---

---

---

---

---

---

---

---

---

---

## Napad na petlje

- nekoliko načina
  - samo jedan resurs u isto vrijeme
    - ponekad neprihvatljivo
  - globalni redosljed resursa
    - svakom se resursu dodijeli unikatni broj
    - resursi se mogu zatražiti samo slijedom njihovih brojeva (neće dobiti resurs s nižim brojem, ako je već dobio resurs s višim)
  - podvarijanta: računaju se samo resursi koje proces trenutno drži

---

---

---

---

---

---

---

---

---

---

## Sprečavanje zastoja - sažetak

- uzajamna isključivost
  - sve preko spoolera
- hold & wait
  - zatraži sve resurse na početku
- zaposljedljivost
  - oduzmi dodijeljeni resurs
- petlja zastoja
  - poredaj resurse slijedno

---

---

---

---

---

---

---

---

## Još neki problemi

- dvofazno zaključavanje (two-phase locking)
  - često praktično "rješenje" (naročito za baze podataka)
  - proces u prvoj fazi krene uzimati sve potrebne resurse
  - u drugoj krene obavljati posao (izmjene u bazi, npr)
  - ako u prvoj fazi naiđe na zaposjednuti resurs
    - otpusti sve što je već zauzeo
    - krene od početka prve faze
  - slično napadu na "hold & wait"
  - traži dobro pripremljen uvjete u kojim se može koristiti
- zastoji bez resursa
  - proces mogu čekati i na neki događaj
- izglednjivanje
  - potrebne su politike dodjeljivanja resursa, radi sprečavanja zastoja
  - može se desiti da neki proces nikad ne dobije resurs

---

---

---

---

---

---

---

---

## Programska podrška mjernih i procesnih sustava

www.zesoi.fer.hr  
<http://www.ZESOI.FER.hr/PPMPS>

ppmps@zesoi.fer.hr

---

---

---

---

---

---

---

---