

OPERATIVNI SISTEMI

Slajdovi su kreirani na osnovu knjige "Operativni sistemi, principi unutrašnje organizacije i dizajna, 7. izdanje", William Stallings, CET, Beograd, 2013.

Procesi

Sadržaj

- **Pojam procesa**
- Stanja procesa
- Upravljačke strukture procesa
- Upravljanje procesom
- Međuprocesna komunikacija
- Izvršavanje OS

OS upravlja izvršavanjem aplikacija

- Resursi treba da budu dostupni za više aplikacija
- Procesor deli više aplikacija
- Procesor i U/I uređaji treba da budu korišćeni efikasno
- Pristup u savremenim OS koji ovo obezbeđuje zasniva se na modelu u kojem izvršavanje aplikacije odgovara postojanju jednog ili više procesa

Definicije procesa

- Program u izvršavanju
- Primerak programa koji se izvršava u računaru
- Entitet koji se može dodeliti i izvršavati u procesoru
- Jedinica aktivnosti koju karakterišu naredbe za izvršavanje, tekuće stanje i dodeljeni skup sistemskih resursa

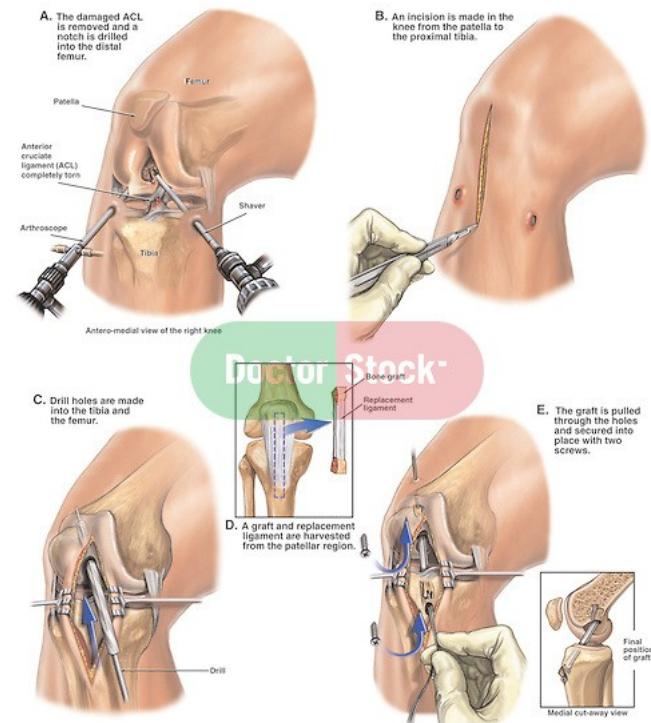
Zahtevi OS pri upravljanju procesima

- Preplitanje izvršavanja više procesa
- Dodela resursa procesima i zaštita dodeljenih resursa od nekontrolisanog pristupa drugih procesa
- Deljenje i razmena informacija između procesa
- Synchronizacija između procesa

Elementi procesa

□ Programske elemente

- šta proces treba da uradi
- može biti deljen između više instanci istog procesa



Elementi procesa

- Skup podataka nad kojima se posao izvršava



Elementi procesa

- Atributi koji opisuju stanje procesa
 - Dodatne informacije koje OS skladišti da bi mogao da upravlja procesom





Izvor: www.youtube.com

Atributi procesa

- Identifikator
- Stanje
- Prioritet
- Programski brojač
 - adresa sledeće instrukcije u programskom kodu
- Pokazivači na memorejske blokove
 - pokazivači na kod i podatke vezane za proces
- Kontekstni podaci
 - podaci iz registara procesora vezani za trenutno stanje procesa
- U/I statusne informacije
 - neobrađeni U/I zahtevi, dodeljeni U/I uređaji, lista fajlova koje proces koristi, ...
- Računovodstvene informacije
 - korišćeno procesorsko vreme, vremenska ograničenja, ...

Upravljački blok procesa (UBP)

- Sadrži atribute procesa
- OS je zadužen za kreiranje i upravljanje upravljačkim blokom procesa
- Omogućuje multiprogramiranje
 - sadrži informacije koje omogućuju da se proces prekine i kasnije nastavi izvršavanje kao da nije došlo do prekida
 - kada se proces prekine, snimaju se vrednosti procesorskih registara u UBP
 - kada proces nastavlja izvršavanje snimljene vrednosti se učitavaju iz UBP u procesorske registre

Identifier
State
Priority
Program counter
Memory pointers
Context data
I/O status information
Accounting information
:

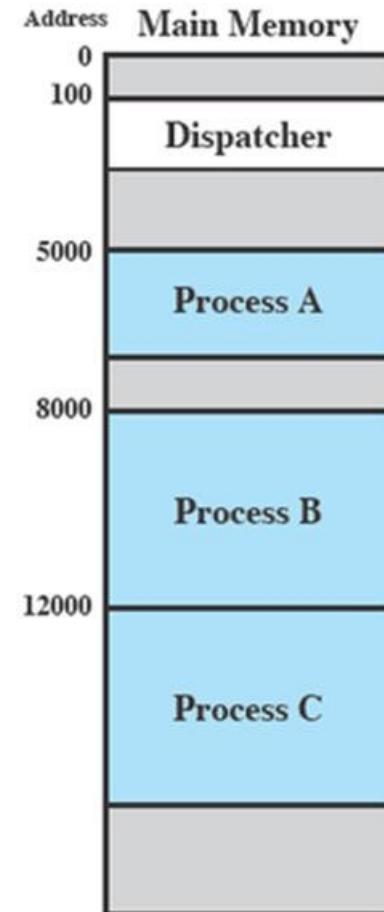
Praćenje izvršavanja procesa

- Trag procesa - niz instrukcija koje proces izvršava, čime je predstavljeno ponašanje procesa
- Ponašanje procesora može se predstaviti isprepletenim prikazom tragova različitih procesa
- Raspoređivač (eng. *dispatcher*) – program koji prebacuje procesor sa jednog procesa na drugi

Primer

Tri procesa – raspored memorije

- Sva tri procesa su kompletno u glavnoj memoriji
- Programski kod raspoređivača je takođe u memoriji



Primer Tri procesa - trag

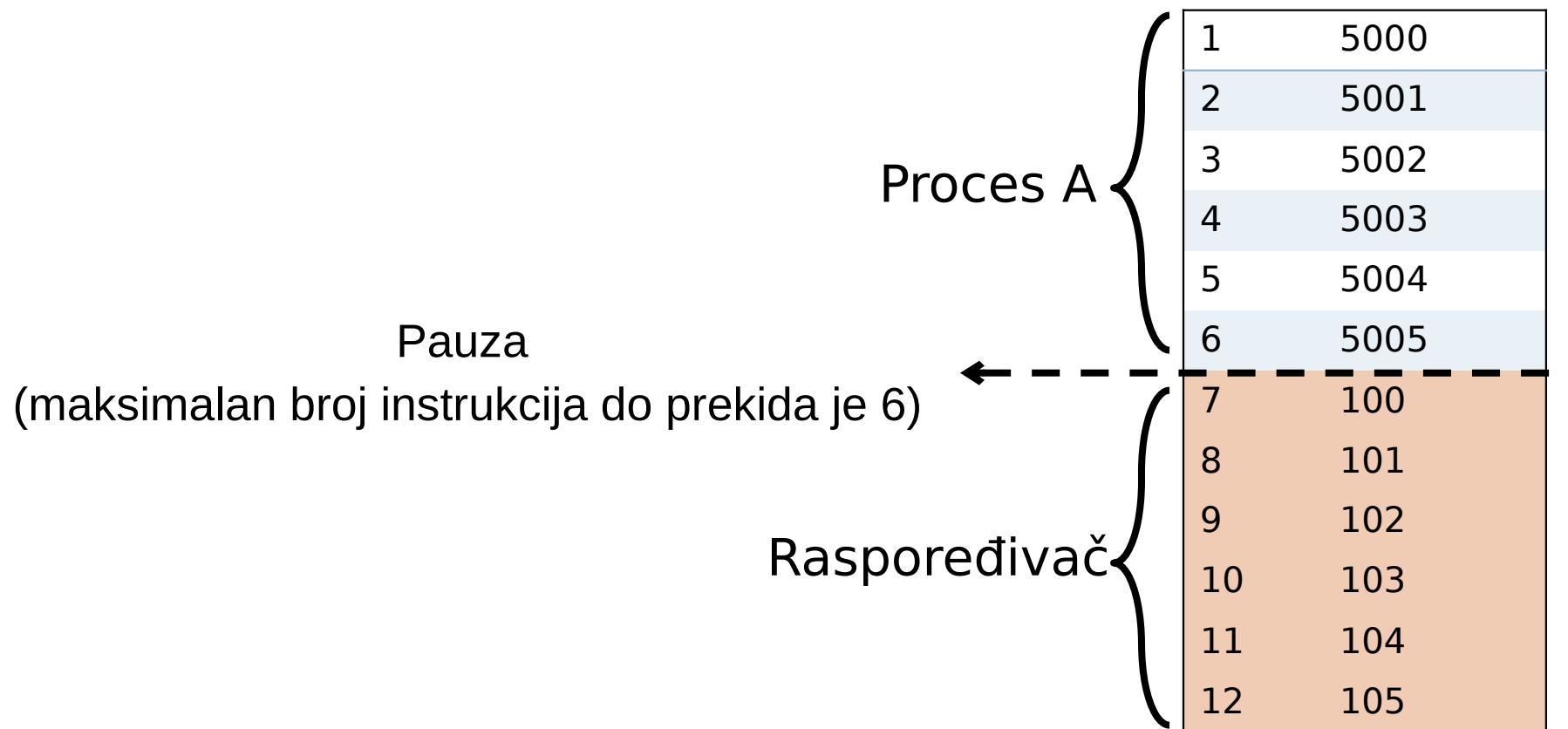
5000	8000	12000
5001	8001	12001
5002	8002	12002
5003	8003	12003
5004		12004
5005		12005
5006		12006
5007		12007
5008		12008
5009		12009
5010		12010
5011		12011

Proces A

Proces B

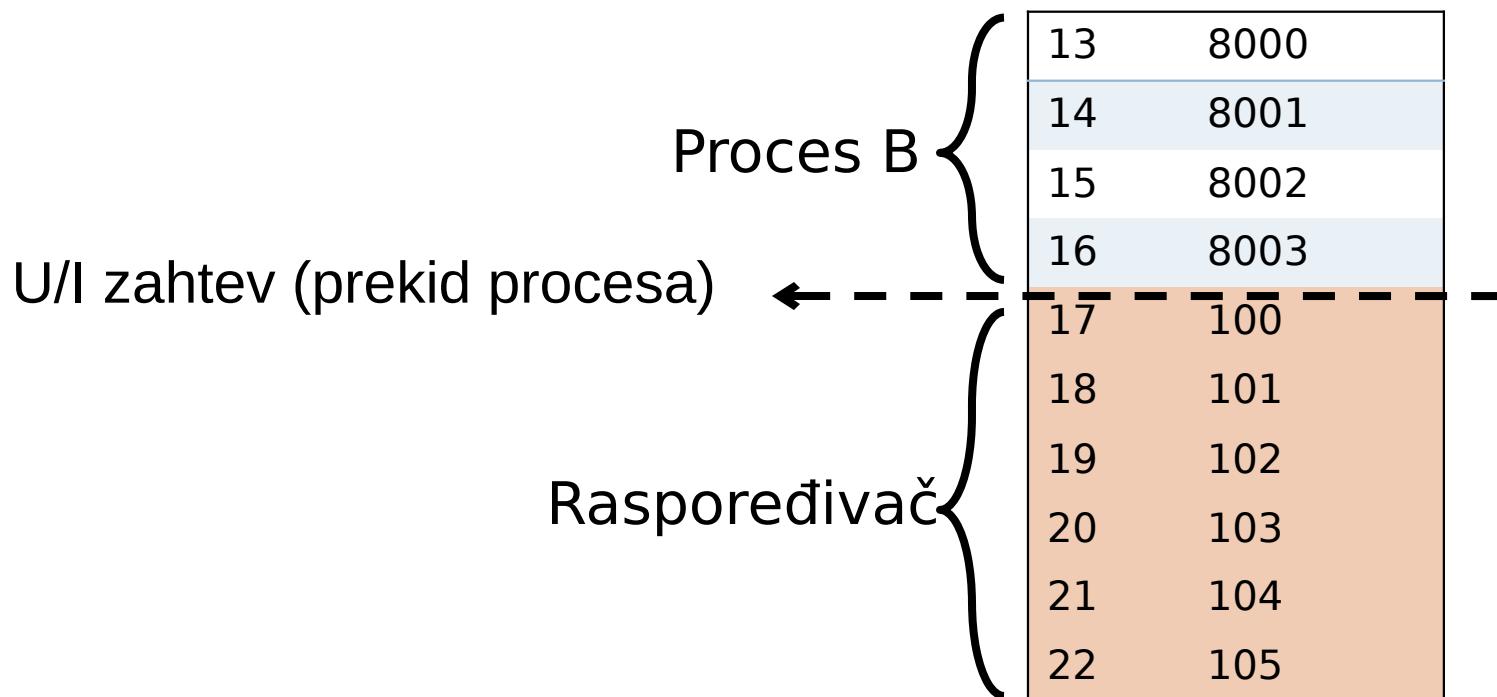
Proces C

Primer Tri procesa – rad procesora



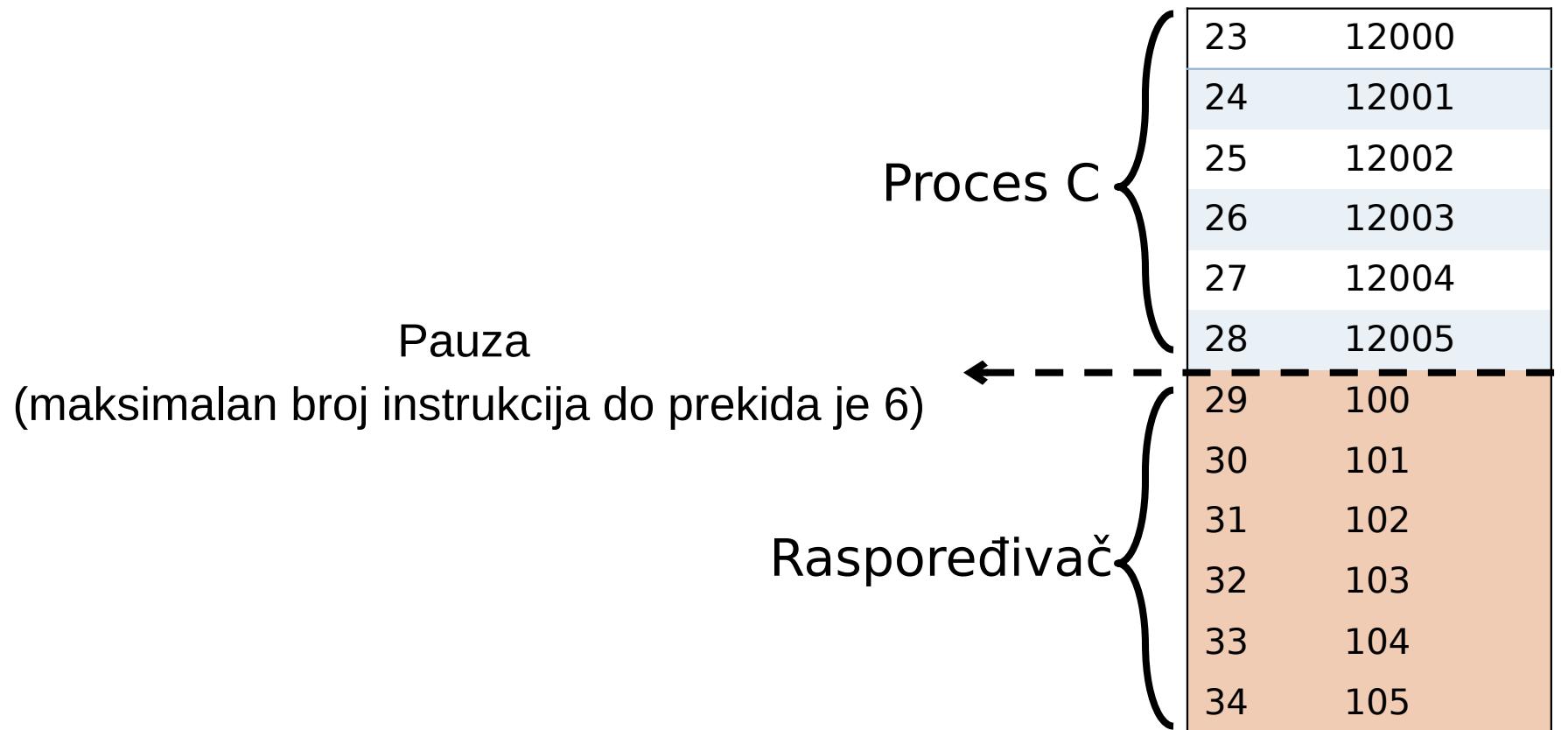
Primer

Tri procesa – rad procesora



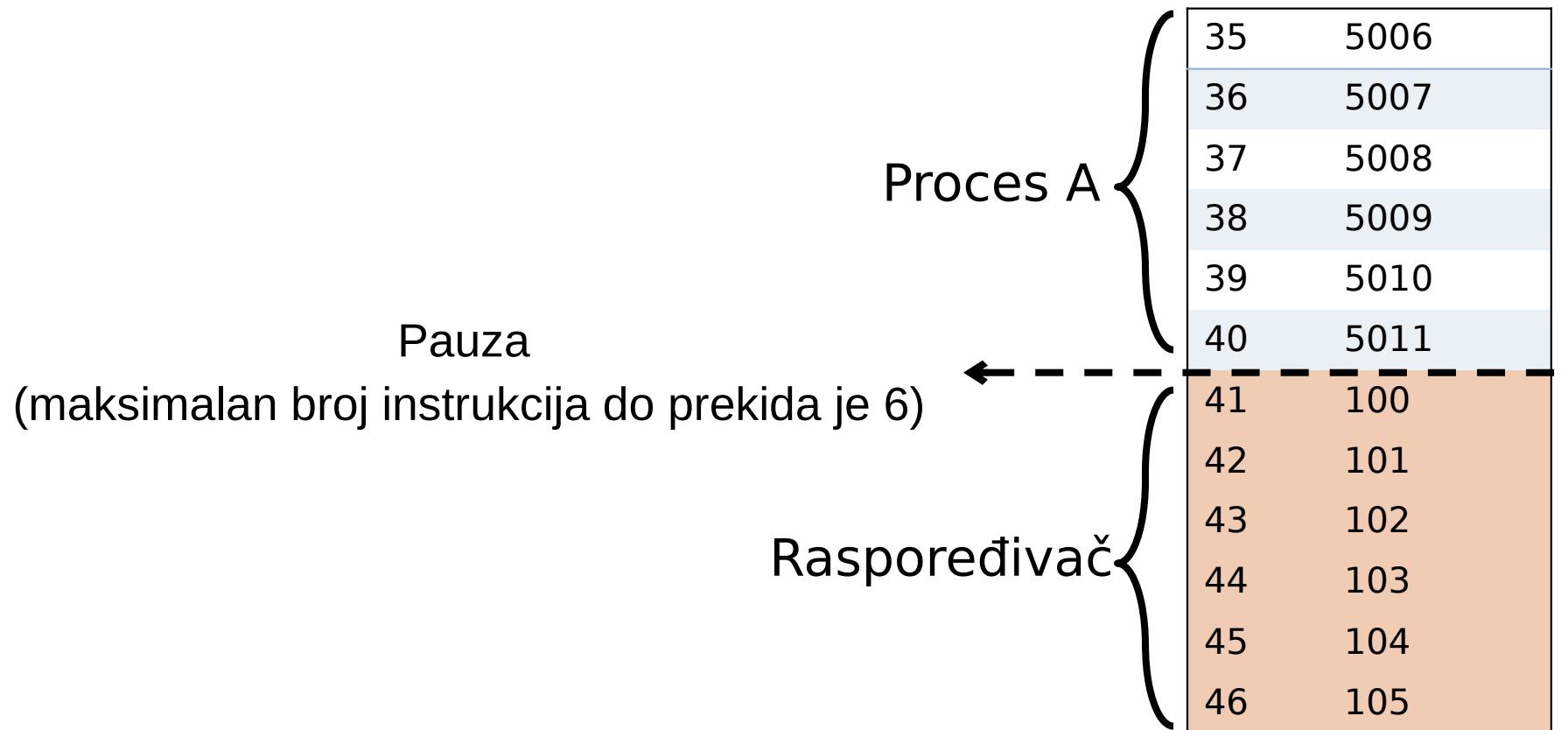
Primer

Tri procesa – rad procesora



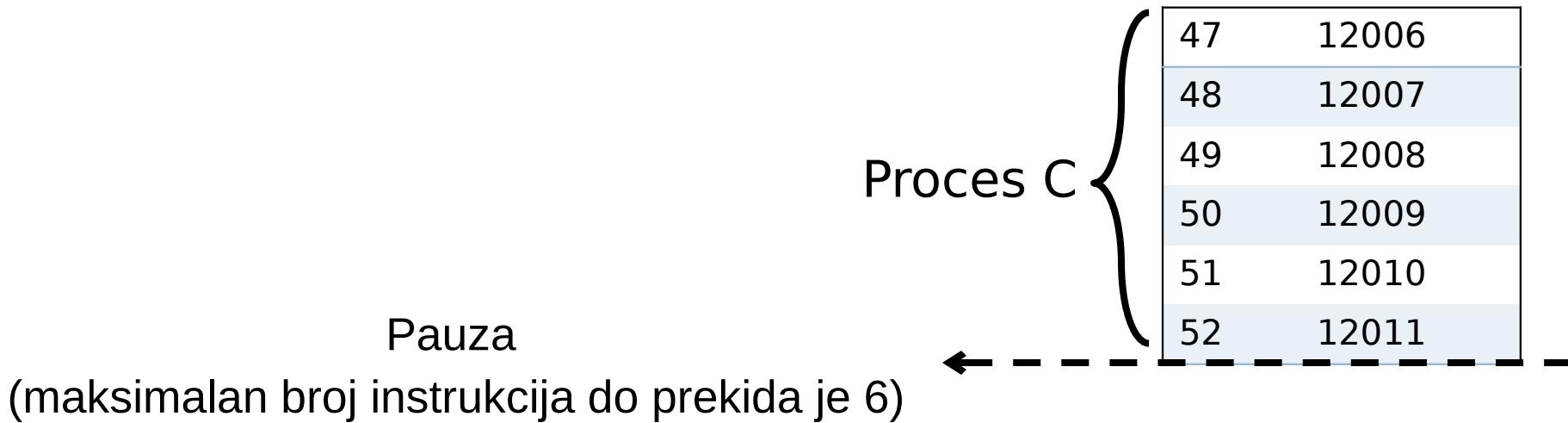
Primer

Tri procesa – rad procesora



Primer

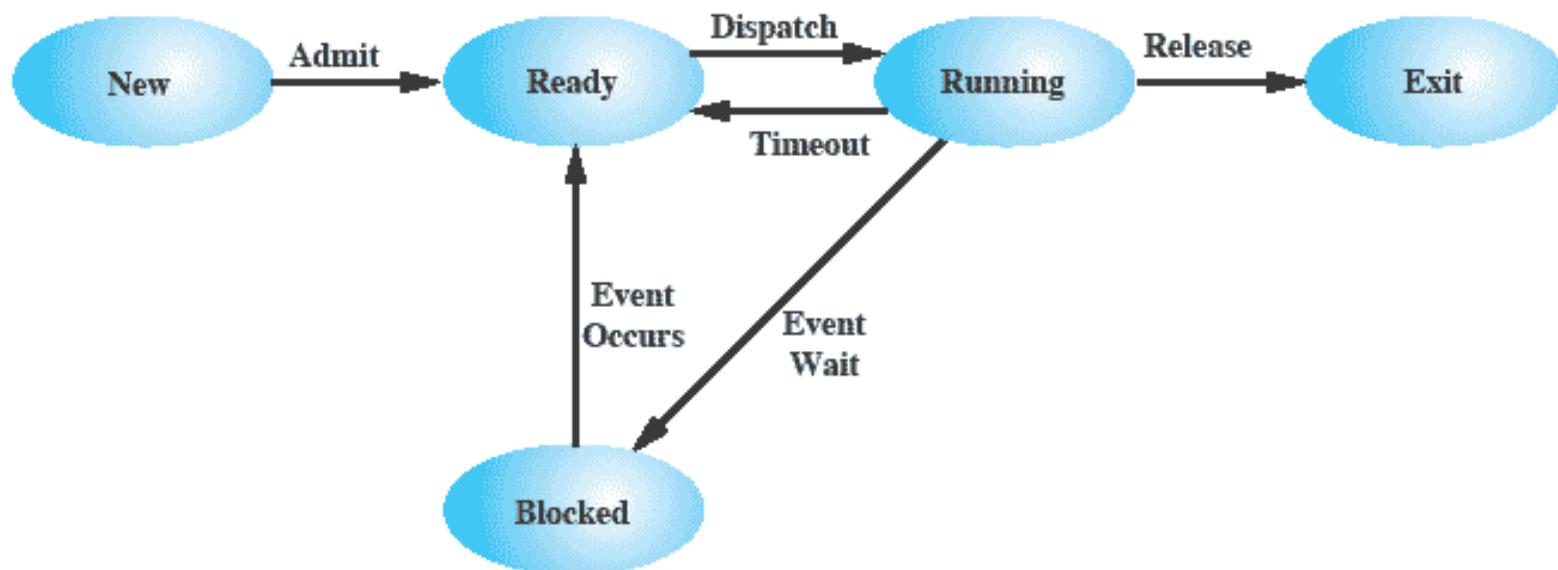
Tri procesa – rad procesora



Sadržaj

- Pojam procesa
- **Stanja procesa**
- Upravljačke strukture procesa
- Upravljanje procesom
- Međuprocesna komunikacija
- Izvršavanje OS

Model procesa sa pet stanja



Stanja procesa

- Izvršavanje
 - proces čije instrukcije procesor trenutno izvršava
- Spreman
 - proces koji je spreman za izvršavanje, ali trenutno procesor ne izvršava njegove instrukcije
- Blokiran (U čekanju)
 - proces koji se ne može izvršavati dok se ne pojavi neki događaj
 - npr. završetak U/I operacije
- Novi
 - proces koji je upravo stvoren, ali ga OS još nije prihvatio u red spremnih procesa
- Izlaz
 - proces koji je OS uklonio iz reda spremnih procesa zato što je završio rad

Stvaranje procesa

- Pri dodavanju novog procesa OS kreira strukture podataka za upravljanje procesom i učitava u radnu memoriju proces ili deo procesa
- Novi proces se kreira od strane OS
 - pri startovanju programa
 - pri izvršavanju procesa, proces (roditelj) može od OS da zatraži kreiranje drugog procesa (potomak)

Završavanje procesa

- Mora postojati mehanizam da OS završi proces na osnovu indikatora dobijenog od procesa
- Indikator za završavanje može da bude
 - korisnička akcija (izlazak iz aplikacije)
 - greška u programu
 - instrukcija roditeljskog procesa za završetak procesa potomka
 - završavanje roditeljskog procesa
- OS je zadužen da osloboди resurse koje je proces koristio

Prelazi stanja

- Null → Novi
 - kada se kreira novi proces
- Novi → Spreman
 - prebacuje se kada je OS spremna da prihvati novi proces
 - obično zbog performansi postoji ograničenje broja aktivnih procesa
- Spreman → Izvršavanje
 - raspoređivač prebacuje jedan od spremnih procesa u stanje izvršavanja

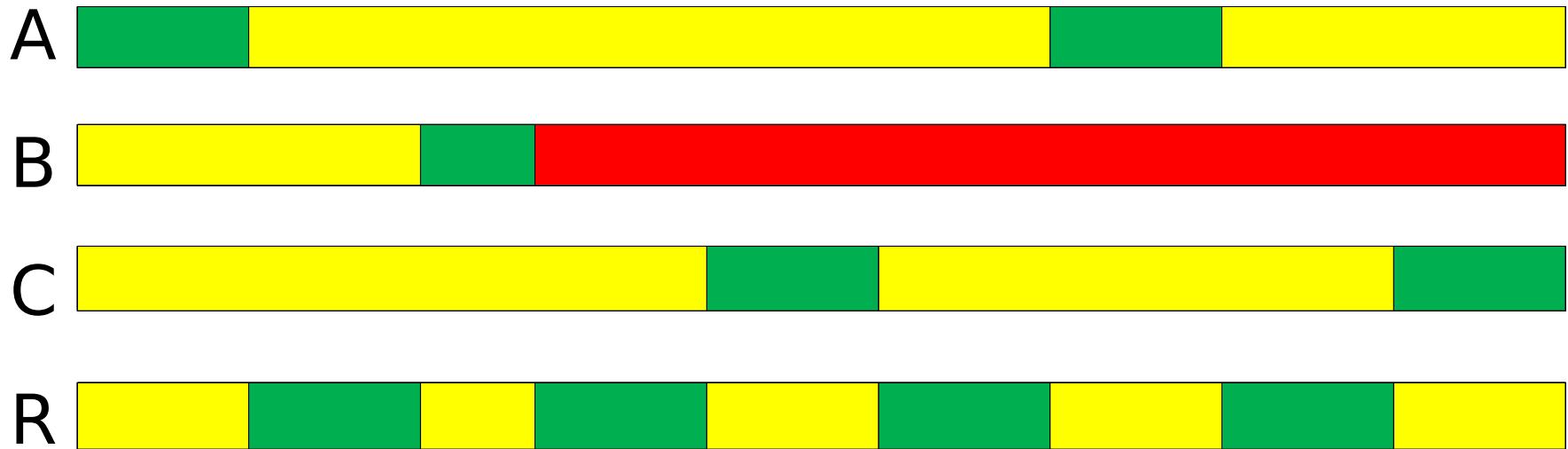
Prelazi stanja

- Izvršavanje → Izlaz
 - kada proces da znak da je završio izvršavanje ili ukoliko prekine sa radom
- Izvršavanje → Spreman
 - kada procesu istekne maksimalno dozvoljeno vreme neprekinutog izvršavanja (*quantum*)
 - kada se pojavi spremjan proces višeg prioriteta
 - kada proces dobровoljno prepusti procesor

Prelazi stanja

- Izvršavanje → Blokiran
 - kada proces zahteva operaciju čiji završetak mora da sačeka
 - npr. ako proces zahteva U/I operaciju ili čeka podatke od drugog procesa
- Blokiran → Spreman
 - kada se pojavi događaj zbog kojeg je proces bio blokiran
- Spreman → Izlaz i Blokiran → Izlaz
 - Proces potomak može biti prekinut od roditelja dok je u stanju Spreman ili Blokiran

Primer – stanja tri procesa



Izvršavanj

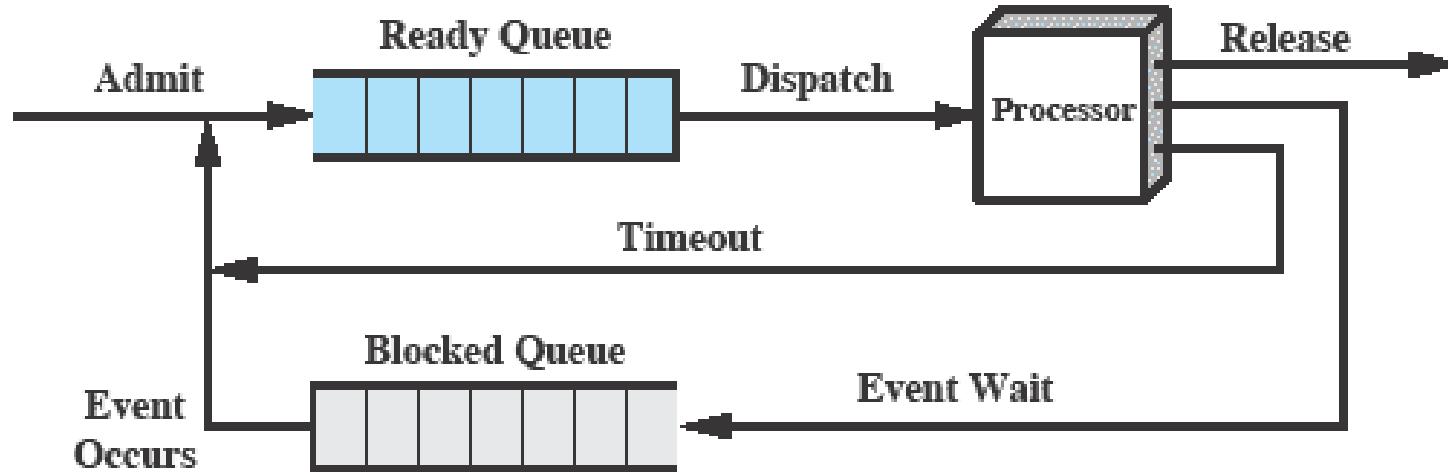


e
Spreman



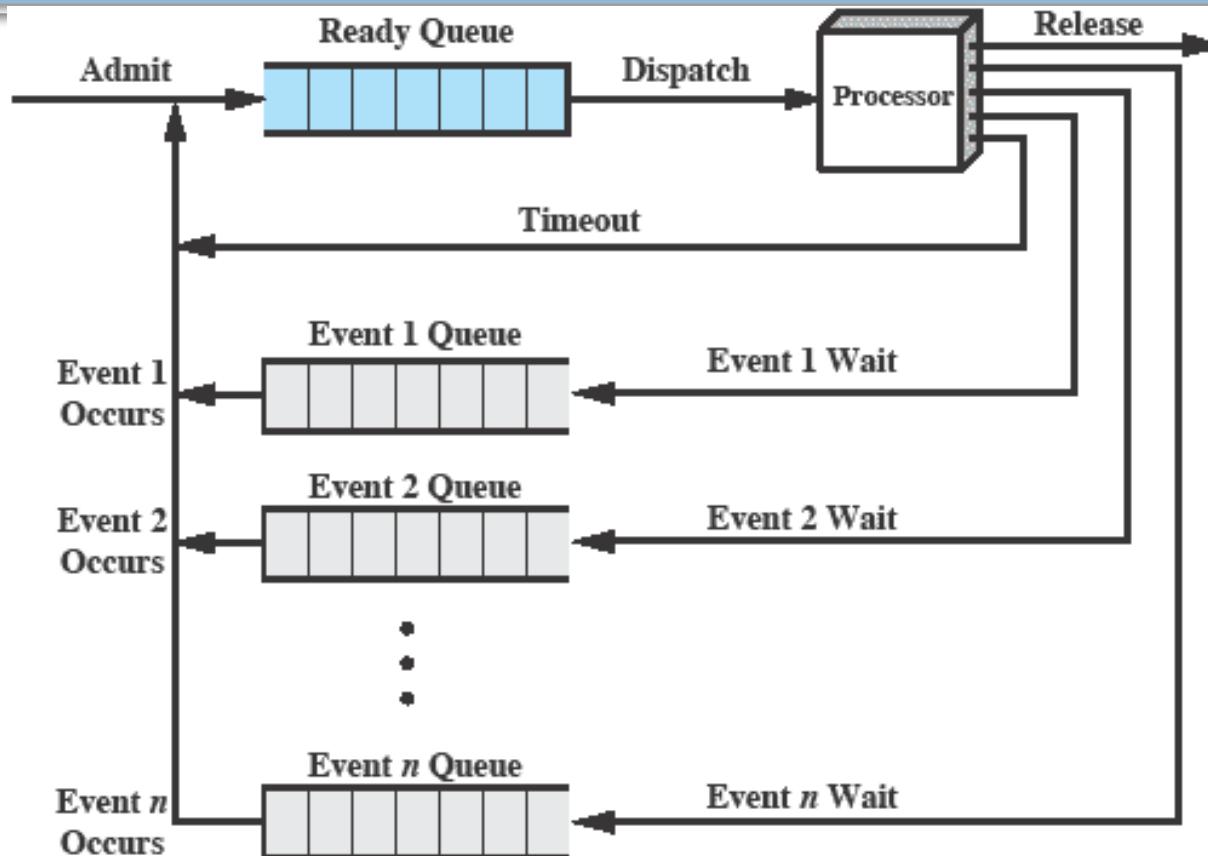
Blokiran

Raspoređivanje sa dva reda



- Red spremnih procesa
 - iz ovog reda se bira jedan za izvršavanje
- Red blokiranih procesa
 - kada se pojavi događaj proces prelazi u red spremnih
 - kada se pojavi događaj OS mora da proveri ceo red

Više redova za blokirane procese



- Za svaki događaj odvojen red blokiranih procesa
- Kada se pojavi događaj, svi procesi iz tog reda se prebacuju u red spremnih

Sadržaj

- Pojam procesa
- Stanja procesa
- **Upravljačke strukture procesa**
- Upravljanje procesom
- Međuprocesna komunikacija
- Izvršavanje OS

Upravljačke strukture procesa

- Da bi upravljaо procesima OS mora da zna
 - lokaciju procesa
 - attribute procesa
 - identifikator
 - stanje
 - ...

Fizička reprezentacija procesa (Slika procesa)

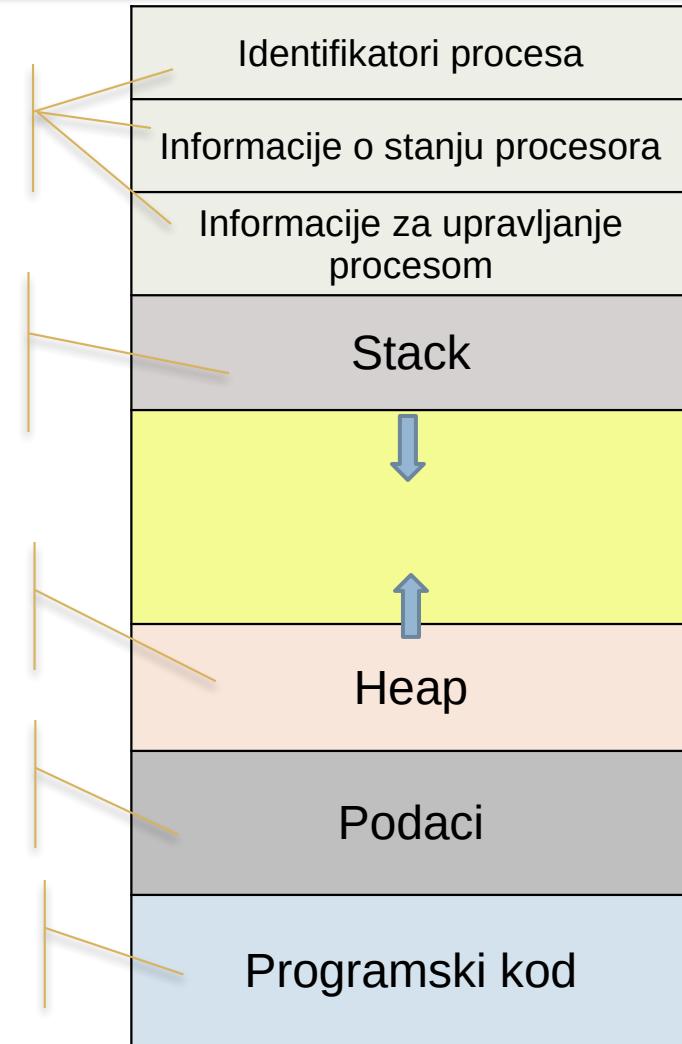
Upravljački blok procesa
(kolekcija atributa procesa)

Privremeni podaci u toku izvršavanja procesa
(parametri i povratne vrednosti funkcija,
lokalne promenljive)

Dinamički alocirana memorija u
toku izvršavanja procesa

Globalne promenljive

Instrukcije koje proces izvršava



Lokacija slike procesa

- Proces je podeljen u stranice
- Sve stranice se nalaze u sekundarnoj memoriji
- U svakom trenutku jedan deo stranica je u glavnoj memoriji
- Stranice ne moraju biti smeštene u fizički susednim blokovima memorije

Identifikatori procesa

- Jedinstveni identifikator procesa
 - obično pozitivan ceo broj
 - upravljačke strukture OS mogu da referenciraju proces preko ovog identifikatora
- Identifikator roditeljskog procesa
 - identifikator procesa koji je stvorio proces
- Korisnički identifikator
 - identifikator korisnika koji je odgovoran za proces

Informacije o stanju procesora

- Predstavljaju sadržaj registara procesora
 - kada se proces prekine, potrebno je sačuvati trenutno stanje procesorskih registara
- Registri vidljivi korisniku
 - mogu se referencirati iz mašinskog koda programa
- Upravljački i statusni registri
 - programski brojač
 - uslovni kodovi – rezultat poslednje operacije
 - informacije o statusu
- Pokazivači na stek
 - pokazuje na vrh steka procesa

Informacije za upravljanje procesom

□ Informacije za raspoređivanje

- stanje procesa
- prioritet
- dodatne informacije za raspoređivanje
 - zavisi od algoritma raspoređivanja
 - npr. vreme čekanja na procesor
- događaj na koji proces čeka

□ Strukturiranje podataka

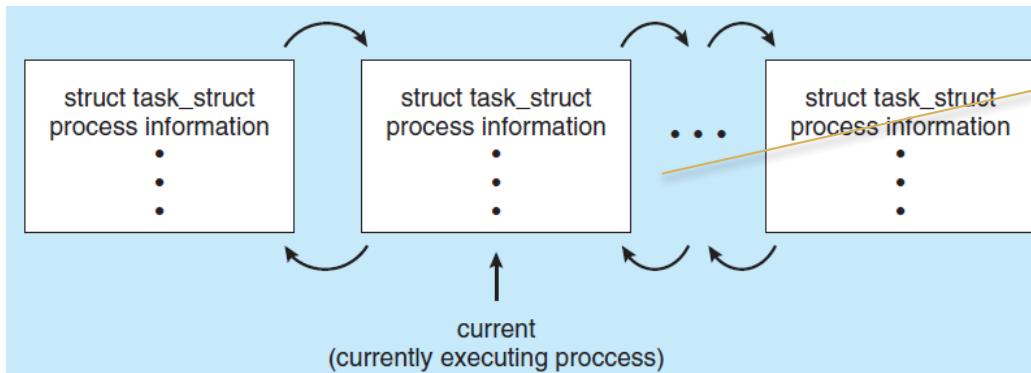
- struktura koja međusobno povezuje procese
- npr. red čekanja procesa istog prioriteta

Informacije za upravljanje procesom

- Međuprocesna komunikacija
 - markeri, signali i poruke vezani sa komunikacijom između procesa
- Privilegije procesa
 - kojim delovima memorije proces može da pristupa i koje instrukcije ima pravo da izvršava
- Upravljanje memorijom
 - pokazivači na tabelu stranica za virtuelnu memoriju
- Vlasništvo nad resursima
 - koje resurse proces kontroliše

Reprezentacija procesa u Linuxu

```
struct task_struct {  
    long state; /* state of the process */  
    struct sched_entity se; /* scheduling information */  
    struct task_struct *parent; /* this process's parent */  
    struct list_head children; /* this process's children */  
    struct files_struct *files; /* list of open files */  
    struct mm_struct *mm; /* address space of this process */  
};
```



Lista aktivnih
procesa u Linuxu

Sadržaj

- Pojam procesa
- Stanja procesa
- Upravljačke strukture procesa
- **Upravljanje procesom**
- Međuprocesna komunikacija
- Izvršavanje OS

Režimi izvršenja procesora

□ Korisnički režim

- režim sa manje privilegija
- za korisničke programe



Izvor: www.youtube.com

Režimi izvršenja procesora

□ Režim kernela

- moguće je izvršavati i privilegovane instrukcije
- potpuna kontrola nad procesorom i svim njegovim instrukcijama, registrima i memorijom
- OS se izvršava u ovom modu



Režim izvršenja procesora

- Promena režima izvršavanja
 - obično postoji bit u statusnom registru procesora koji ukazuje na režim
 - bit se menja kada se desi određeni događaj
 - Sistemski poziv prevodi sistem iz korisničkog režima u režim kernela

Stvaranje procesa

1. Novom procesu se dodeljuje jedinstveni identifikator
 - u tabelu procesa dodaje se nova stavka
2. Dodeljuje se prostor za proces
 - za sve elemente slike procesa
 - Ovaj prostor može da bude i virtuelan, a ne fizički
3. Inicijalizuje se upravljački blok procesa
 - programski brojač se postavlja na prvu instrukciju
 - ostale vrednosti na *default*
4. Ažuriraju se strukture sa vezama procesa
 - npr. proces se uvezuje u listu procesa u odgovarajućem stanju
5. Kreiraju se ili ažuriraju druge strukture podataka
 - npr. kreira se fajl sa podacima o aktivnosti procesa

Komutiranje procesa

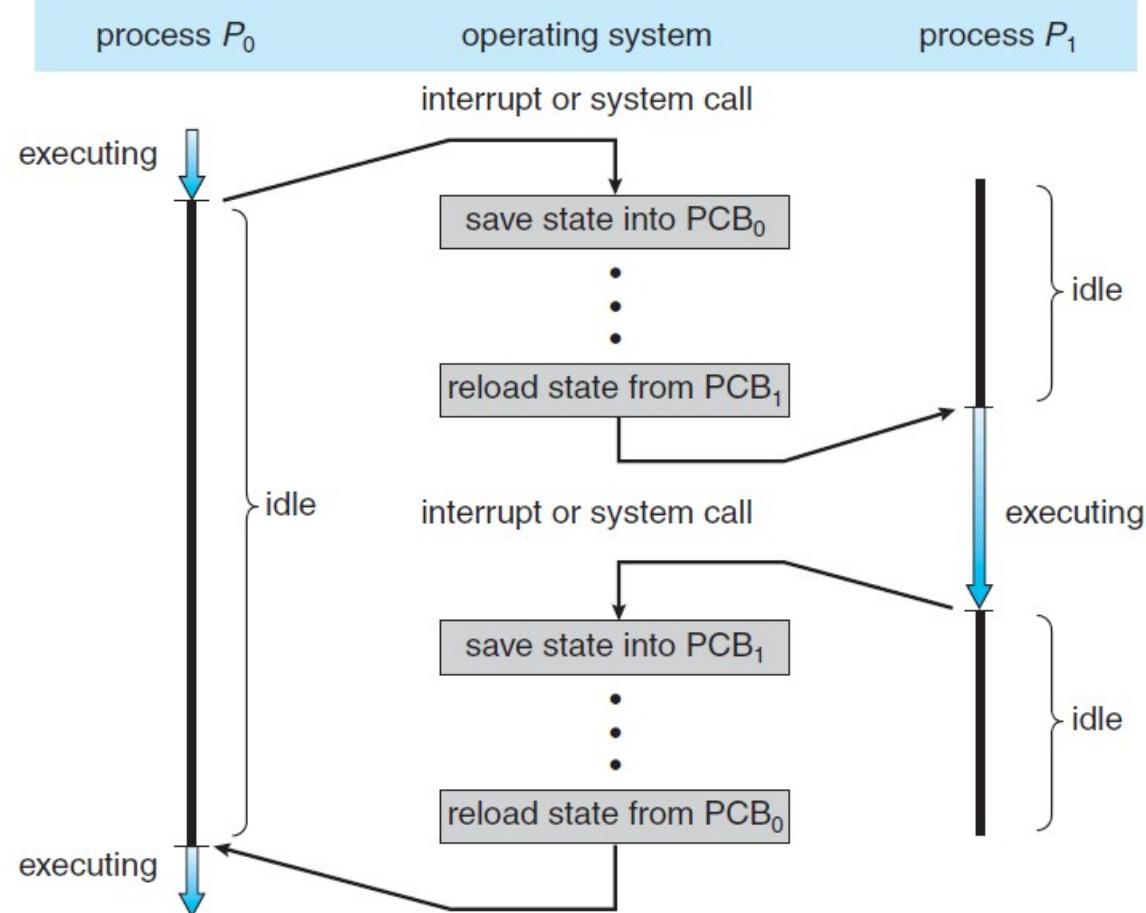
- Promena trenutno aktivnog procesa koji se izvršava na procesoru



Izvor: www.youtube.com

Komutiranje procesa

- Prekida se proces koji se izvršava
- OS drugom procesu dodeljuje stanje Izvršavanje
- Predaje se kontrola tom drugom procesu



Komutiranje procesa

- Koji događaj pokreće komutiranje procesa?
- Koje strukture podataka OS ažurira pri komutaciji procesa?

Kada se komutira proces

- Kada se desi prekid
 - Prekid generatora takta
 - ako je isteklo maksimalno dozvoljeno korišćenje procesora (*time slice*), proces se prekida i prebacuje u stanje Spreman
 - U/I prekid
 - OS prebacuje procese blokirane zbog tog U/I događaja u stanje Spreman
 - Greška memorije
 - proces referencira stranicu koja nije u glavnoj memoriji
 - OS preuzima kontrolu da bi prebacio traženu stranicu iz spoljne memorije u glavnu memoriju

Kada se komutira proces

- Kada se desi greška u izvršavanju programa
 - OS preuzima kontrolu da bi reagovao na grešku
- Poziv operacije OS iz procesa (sistemska poziv)
 - npr. proces zahteva pristup fajlu i OS preuzima kontrolu da bi izvršio U/I akciju
 - Ne podrazumeva nužno prelazak na izvršavanje drugog procesa
 - Ako operacija ne zahteva blokiranje procesa, proces će nastaviti da izvršava kod OS u režimu kernela
 - Moraju se samo sačuvati stanja procesorskih registara

Promena stanja procesa

1. Čuvanje sadržaja procesora
 - programski brojač
 - drugi registri
2. Ažuriranje upravljačkog bloka procesa
 - promena stanja na Spreman ili Blokiran
 - ažuriranje drugih polja
3. Prebacivanje upravljačkog bloka procesa u odgovarajući red
 - red spremnih ili blokiranih
4. Izbor narednog procesa za izvršavanje

Promena stanja procesa

5. Ažuriranje upravljačkog bloka izabranog procesa
 - promena stanja na Izvršavanje
6. Ažuriranje strukture podataka za upravljanje memorijom
 - moguće izmene stranica u glavnoj memoriji
7. Postavljanje sadržaja registara procesora u skladu sa sadržajem upravljačkog bloka procesa izabranog za izvršavanje

Sadržaj

- Pojam procesa
- Stanja procesa
- Upravljačke strukture procesa
- Upravljanje procesom
- **Međuprocesna komunikacija**
- Izvršavanje OS

Kooperacija između procesa

- Procesi koji se konkurentno izvršavaju su
 - Nezavisni
 - Ako na njihovo izvršavanje ne utiču drugi procesi
 - Proces koji ne deli podatke sa drugim procesima je nezavisan
 - Međusobno zavisni
 - Ako utiču jedni drugima na izvršavanje
 - Svaki proces koji deli podatke sa drugim procesima je zavisan od njih
 - Međusobno zavisni procesi moraju da sarađuju putem međuprocesne komunikacije

Razlozi za međuprocesnu komunikaciju

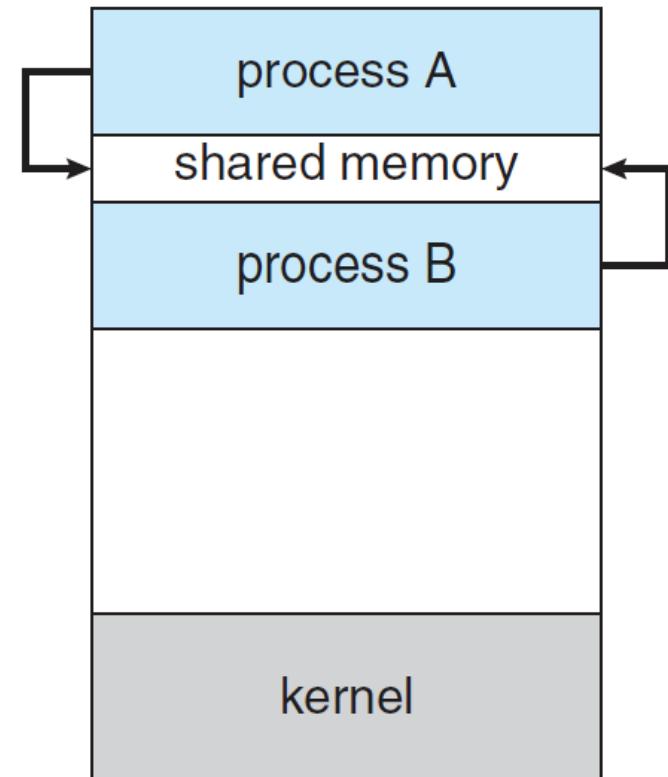
- Deljenje informacija
 - Različitim procesima trebaju isti podaci
 - Npr. fajl kojem pristupaju različiti procesi
- Ubrzanje izračunavanja
 - Nad istim podacima izračunavanja mogu da vrše različiti procesi kako bi na multiprocesorskoj arhitekturi dobili ubrzanje
- Modularnost
 - Pri implementaciji se mogu napraviti logički tokovi izvršavanja koji pristupaju istim podacima

Načini međuprocesne komunikacije

- Dva osnovna načina postoje
 - Deljena memorija i
 - Razmena poruka

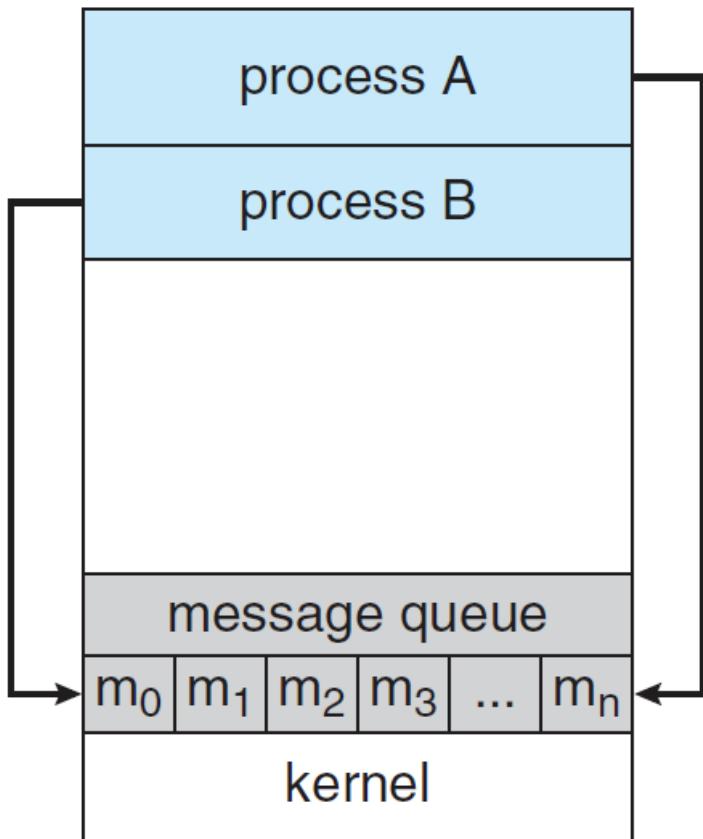
Deljena memorija

- Procesi uspostavljaju region deljene memorije
 - Najčešće u adresnom prostoru procesa koji kreira region
 - Drugi proces zakači region u svoj adresni prostor
- Procesi komuniciraju kroz upis i čitanje podataka u ovom regionu
- Format podataka, isključivost i sinhronizacija su odgovornost procesa



Razmena poruka

- Procesi ne moraju eksplicitno da kreiraju region deljene memorije
- Komuniciraju pozivima operacija
 - `send(message)`
 - `receive(message)`



Načini razmene poruka

- Blokirana ili neblokirana komunikacija
- Direktno ili indirektno adresiranje
- Baferovani ili nebaferovani zapis

Blokirana komunikacija

- Blokirano slanje
 - Nakon slanja poruke pošiljalac ostaje blokiran dok primalac ne primi poruku
- Blokirani prijem
 - Nakon poziva prijema poruke, primalac ostaje blokiran dok poruka ne bude raspoloživa

Neblokirana komunikacija

- Neblokirano slanje
 - Nakon slanja poruke pošiljalac nastavlja sa izvršavanjem
- Neblokirani prijem
 - Nakon poziva prijema poruke, primalac dobija poruku ako je raspoloživa ili NULL

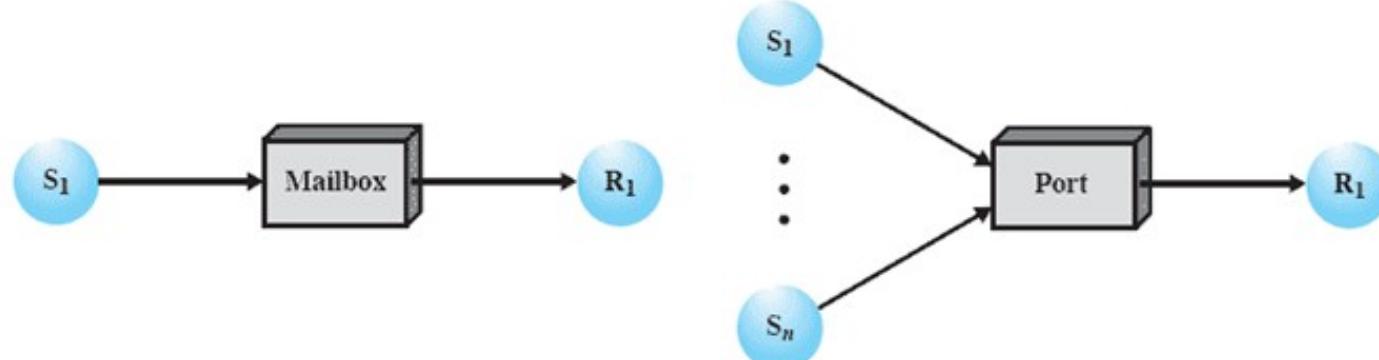
Direktno adresiranje

- Proces mora eksplisitno da navede identifikator drugog procesa kojim komunicira
 - `send(P, message)`
 - `receive(Q, message)`
- Komunikacioni link povezuje tačno dva procesa
- Operacija receive može biti realizovana i tako da se ne navodi od kojeg pošiljaoca se prima poruka
 - `receive(id, message)`
 - u parametar id se upiše identifikator procesa čija je poruka pročitana

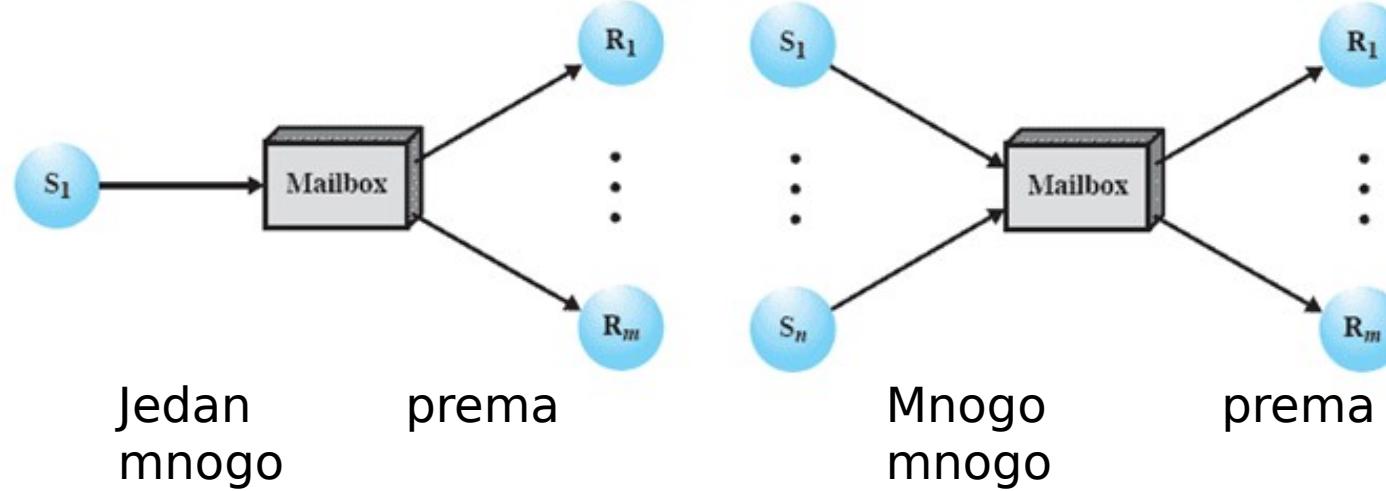
Indrektno adresiranje

- Poruke se ne šalju direktno od pošiljaoca do primaoca
 - Šalju se deljenim strukturama podataka
 - Poštanski sandučići (*mailboxes*)
- Proces/procesi upisuju podatke u sandučić
- Drugi proces/procesi čitaju podatke iz sandučića

Oblici indirektnog adresiranja



Jedan jedan prema Mnogo jedan prema



Jedan mnogo prema Mnogo mnogo prema

Vlasništvo nad sandučićem

- Ako je vlasnik proces
 - Sandučić nestaje kada se proces završi
 - Procesi koji komuniciraju sa ovim sandučićem moraju dobiti obaveštenje da sandučić više ne postoji
- Ako je vlasnik OS
 - Sandučić postoji nezavisno od procesa
 - OS pruža mehanizam da se
 - Kreira novi sandučić
 - Pošalje/primi poruka u/iz sandučića
 - Obriše sandučić

Baferovanje poruka

- Poruke u sandučiću stoje u privremenom redu
- Red može biti takav da ima
 - Nulti kapacitet
 - Samo jedna poruka može da stane
 - Ograničen kapacitet
 - Najviše N poruka može istovremeno da stane u bafer
 - Neograničen kapacitet
 - Bilo koji broj poruka se može ubaciti u bafer

Poređenje tipova međuprocesne komunikacije

- Oba tipa komunikacije se koriste u savremenim OS
- Deljena memorija
 - Brža, jer ne zahteva sistemske pozive za pristup podacima
- Razmena poruka
 - Pogodniji za distribuirane sisteme u kojima uglavnom nema deljene memorije

Pipes

- Pipes je poseban tip međuprocesne komunikacije koji OS standardno pružaju
- Omogućuju komunikaciju dva procesa po principu proizvođač-potrošač
- Proizvođač upisuje u jedan kraj
- Potrošač čita podatke sa drugog kraja



Pipes



Izvor: www.youtube.com

Tipovi pipes

- Da li je komunikacija jednosmerna ili dvosmerna
- Ako je dvosmerna komunikacija
 - Da li istovremeno podaci mogu da putuju u oba smera (*full duplex*) ili
 - U jednom trenutku samo u jednom smeru (*half duplex*)
- Da li treba neka relacija (npr. roditelj-potomak) da postoji između procesa
- Da li je komunikacija samo između procesa na istom računaru ili može i preko mreže

Primer pipes u Linuxu

ls | grep Temp

ls → Documents
Downloads
Templates
Temp



grep Temp → **Templates**
Temp

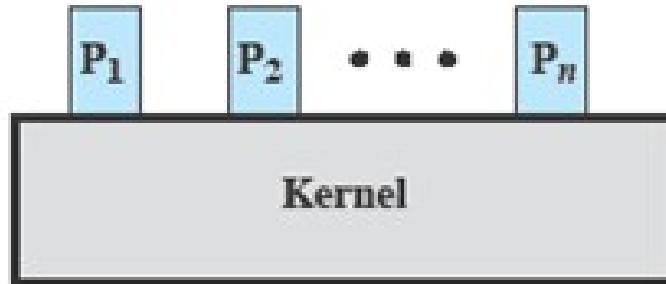
Sadržaj

- Pojam procesa
- Stanja procesa
- Upravljačke strukture procesa
- Upravljanje procesom
- Međuprocesna komunikacija
- **Izvršavanje OS**

Izvršavanje OS

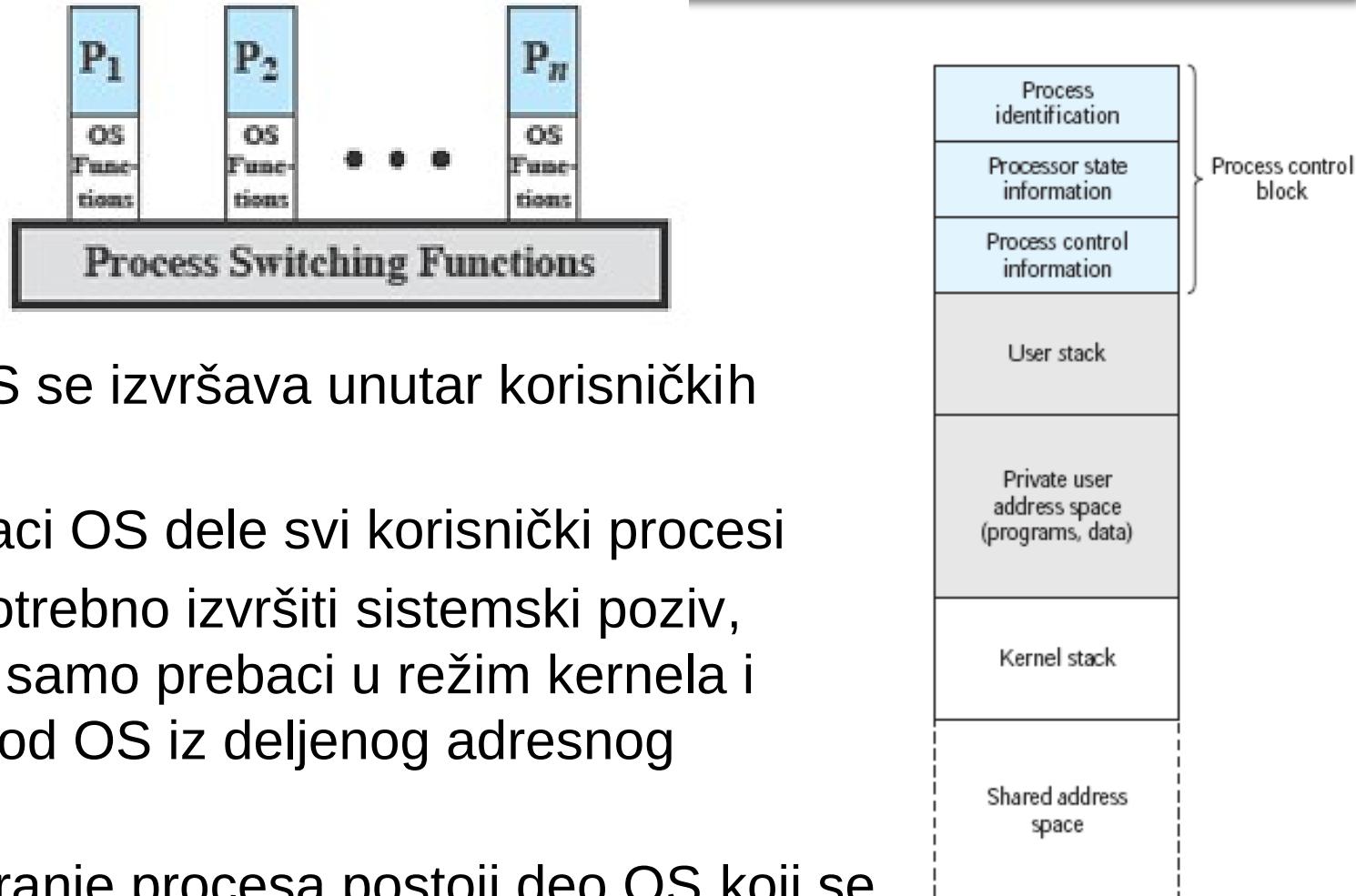
- OS je skup programa koje izvršava procesor
- Radi kao i ostali softver
- Da li je OS proces?
- Ko i kako njime upravlja?
- Postoje različiti pristupi u dizajnu

Jezgro OS koje nije proces



- Kod starijih OS
- OS se izvršava izvan bilo kog procesa
- OS je zaseban entitet
- Kod OS izvršava se u privilegovanim režimu
- Kao procesi se tretiraju samo korisnički procesi

Izvršavanje unutar korisničkih procesa



- Softver OS se izvršava unutar korisničkih procesa
- Kod i podaci OS dele svi korisnički procesi
- Kada je potrebno izvršiti sistemski poziv, proces se samo prebaci u režim kernela i izvršava kod OS iz deljenog adresnog prostora
- Za komutiranje procesa postoji deo OS koji se izvršava izvan korisničkih procesa

OS zasnovan na procesima



- OS je implementiran kao skup procesa
- Funkcije jezgra organizovane su u procese
- Ovi procesi se izvršavaju u režimu kernela