

# **POSLOVNA INFORMATIKA**

## **VEŽBE-ZADACI-3-KOLOKVIJUM**

---

**prof. dr Zlatko Langović**  
**ass. Ognjen Jevremović**

# LITERATURA

---

- R. Kelly Rainer Jr, Efraim Turban, Uvod u informacione sisteme, Datastatus, 2009.

# PLAN RADA - VEŽBE

---

- Modeli i modelovanje
- Analiza zahteva i specifikacija is
- SSA – Strukturna sistemska analiza, dijagram konteksta, dijagram toka podataka, rečnik podataka, minispecifikacija primitivnih procesa, vežbanje
- Model podataka, MOV (Model objekti-veze (eng. Entity Relationship model - ER)), vežbanje
- Normalizacija (projektovanje relacija), vežbanje
- SQL = osnove, vežbanje

## **1. Definisiranje ciljeva razvoja IS**

- definisiranje ciljeva razvoja IS u skladu sa strategijom i ciljevima preduzeća,
- utvrđivanje mogućnosti primene savremenih informacionih tehnologija
- plan razvoja IS

### **1. Analiza zahteva korisnika**

- snimanje stanja: pregled dokumenata, intervju
- logička specifikacija procesa - strukturna systemska analiza,

### **2. Projektovanje (logicko i fizicko)**

- model sistema, objekti, veze i odnosi izmedju njih (MOV, ER)
- generisanje baza podataka

### **3. Aplikativno modeliranje**

- Izrada aplikacija, testiranje

### **4. Implementacija**

- fizičko povezivanje opreme, instalacija softvera

### **5. Održavanje**

- korekcije, usavršavanje

# MODELI I MODELOVANJE

---

## MODELI I MODELOVANJE

- Model je "subjektivni odraz objektivne stvarnosti". Zbog toga može da postoji više različitih modela istog sistema, sa istog ili različitih aspekata.
- Modelovanjem se bolje razume realni sistem
- Primer jezika-alata za modelovanje sistema - **UML**

---

- **MODELI ZA OPIS FUNKCIJA SISTEMA:**

Dijagrami slučajeva korišćenja: "Use Case Diagrams"

(Strukturalna sistemska analiza)

- **MODELI ZA OPIS STRUKTURE SISTEMA:**

Dijagrami klasa (Model objekti veze)

- **MODELI ZA OPIS DINAMIKE:**

Dijagrami sekvenci ( Sequence Diagrams) Dijagrami

- ☛ kolaboracije (Collaboration Diagrams)

Dijagrami promene stanja (State Transition Diagrams)

- ☛ Dijagrami aktivnosti (Activity Diagrams)

- **DIJAGRAMI ZA OPIS IMPLEMENTACIJE**

Deployment Diagrams

- ☛

# ANALIZA ZAHTEVA I SPECIFIKACIJA IS

---

## FUNKCIONALNI MODEL SISTEMA

- ✓ predstavlja sistem kao "crnu kutiju"
- ✓ predstavlja se funkcionalnost sistema na način kako je vide spoljni objekti
- ✓ predstavljaju se ulazi i izlazi iz sistema i funkcije koje transformišu ulaze (pobudu, stimulaciju) u izlaze
- ✓ Zasto ? – jer su IS po pravilu složeni, te prvi modeli u razvoju sistema treba da budu funkcionalni
- ✓ kao alat za modelovanje funkcija sistema- strukturna sistemska analiza

# ANALIZA ZAHTEVA I SPECIFIKACIJA IS

---

## FUNKCIONALNI MODEL SISTEMA

Zadatak funkcionalnog modelovanja je:

- dekomponovanje složenog sistema na skup podsistema – SSA (strukturna systemska analiza)
- opis pojedinačnih podmodela – SK (use case - slučajevi koriscenja)



# ANALIZA ZAHTEVA I SPECIFIKACIJA IS

---

## STRUKTURNA SISTEMSKA ANALIZA

- Strukturna sistemska analiza(SSA) je jedan od alata za specifikaciju informacionog sistema.
- SSA posmatra informacioni sistem kao **funkciju (proces obrade)** koja, na bazi ulaznih, generiše autput - izlazne podatke.

# ANALIZA ZAHTEVA I SPECIFIKACIJA IS

## STRUKTURNA SISTEMSKA ANALIZA

### Osnovni koncepti za specifikaciju IS u SSA su:

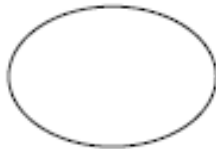
- funkcije, odnosno procesi obrade podataka,
  - interfejsi, objekti van sistema sa kojima sistem komunicira preko tokova podataka,
  - tokovi podataka, preko kojih se podaci prenose između
  - interfejsa, funkcije i skladišta,
    - skladišta podataka, u kojima se permanentno čuvaju
- stanja sistema.

Njihov međusobni odnos se prikazuje preko **dijagrama toka podataka** koji prikazuje vezu interfejsa, odnosno skladišta kao izvora odnosno ponora podataka, sa

# ANALIZA ZAHTEVA I SPECIFIKACIJA IS

## SSA- osnovni koncepti

### Grafički simboli:



Funkcija ili proces obrade podataka



Interfejs



Tok podataka



Skladište podataka

# ANALIZA ZAHTEVA I SPECIFIKACIJA IS

---

## SSA – osnovni koncepti

- dijagram toka podataka na najvišem nivou hijerarhije
- **dijagram konteksta**
- pravilo SSA - jedan proces (funkcija) u DK
- hijerarhijska dekompozicija procesa - razlaganje procesa na potprocese do primitivnih funkcija (koje se dalje ne razlazu)

# ~~ANALIZA ZAHTEVA I SPECIFIKACIJA IS~~

## SSA

Potpunu specifikaciju IS čine:

1. Hijerarhijski organizovan skup dijagrama toka podataka;
2. Rečnik podataka koji opisuje sadržaj i strukturu svih tokova skladišta podataka;
3. Specifikacija logike primitivnih procesa.

- **Dijagram dekompozicije** koji prikazuje celokupnu dekompoziciju sistema, od Dijagrama konteksta do primitivnih funkcija - **Jackson-ovi dijagrami za opis strukture podataka i programa**

# ANALIZA ZAHTEVA I SPECIFIKACIJA IS

---

## SSA- SINTAKSNA I METODOLOŠKA PRAVILA

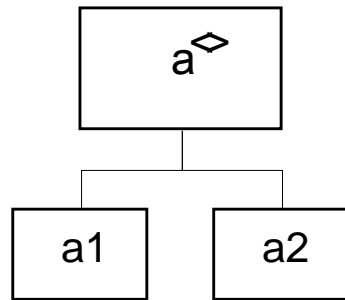
- **pravilo balansa tokova:** Ulazni i izlazni tokovi na celokunom DTP-u koji je dobijen dekompozicijom nekog procesa P moraju biti ekvivalentni sa ulaznim i izlaznim tokovima toga procesa P na dijagramu višeg nivoa.
- Kao najvažnije metodološko pravilo koristi se pravilo da **funkcije na DTP-u između sebe treba da komuniciraju isključivo preko skladišta.**

# ANALIZA ZAHTEVA I SPECIFIKACIJA IS

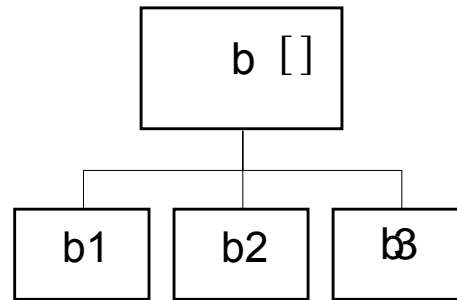
---

## JACKSON-ovi DIJAGRAMI ZA OPIS STRUKTURE PODATAKA I PROGRAMA

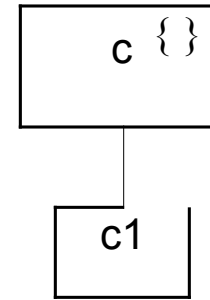
Oznake:



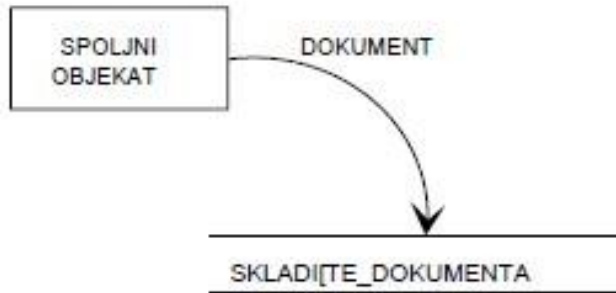
Agregacija  
podataka  
Sekvencija



Selekcija  
podataka  
Selekcija



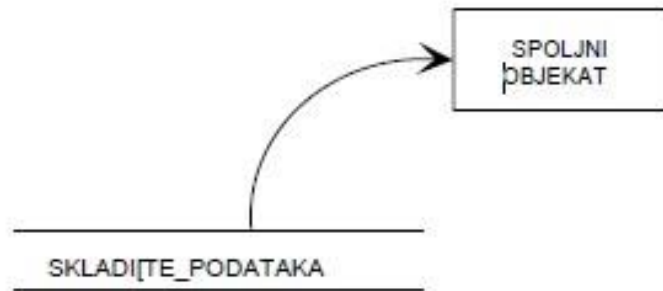
Skup  
Iteracija



(a)



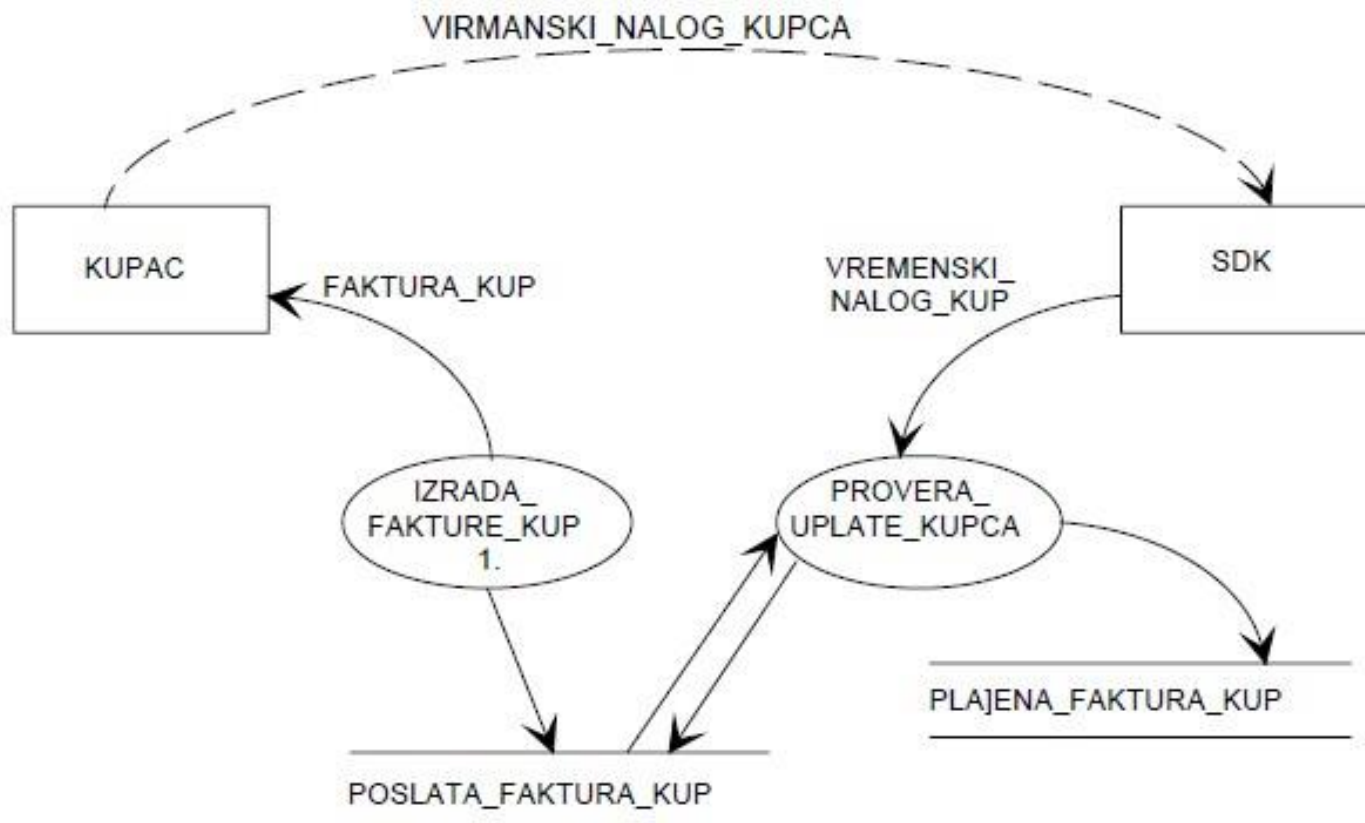
(b)



nepravilan DTP

pravilan DTP





nepravilan DTP

# PRIMER 1. VERBALNI OPIS SISTEMA

---

Kupac se obrati prodajnoj službi preduzeća i pregledava katalog sa modelima koje proizvodi preduzeće. Kupac se odlučuje za neki od modela (moguće ukoliko ima neki specifičan zahtev, sa razvojnim odeljenjem ugovara detalje specijalne porudžbine)...

Proizvodnja na osnovu primljenih naloga za izradu proizvoda obavlja odg.aktivnosti...

Sluzba nabavke materijala radi na osnovu pregleda stanja zaliha...

Razvojna sluzba dizajnira nove modele na osnovu praćenja trenda na tržištu, kao i na zahtev kupaca...

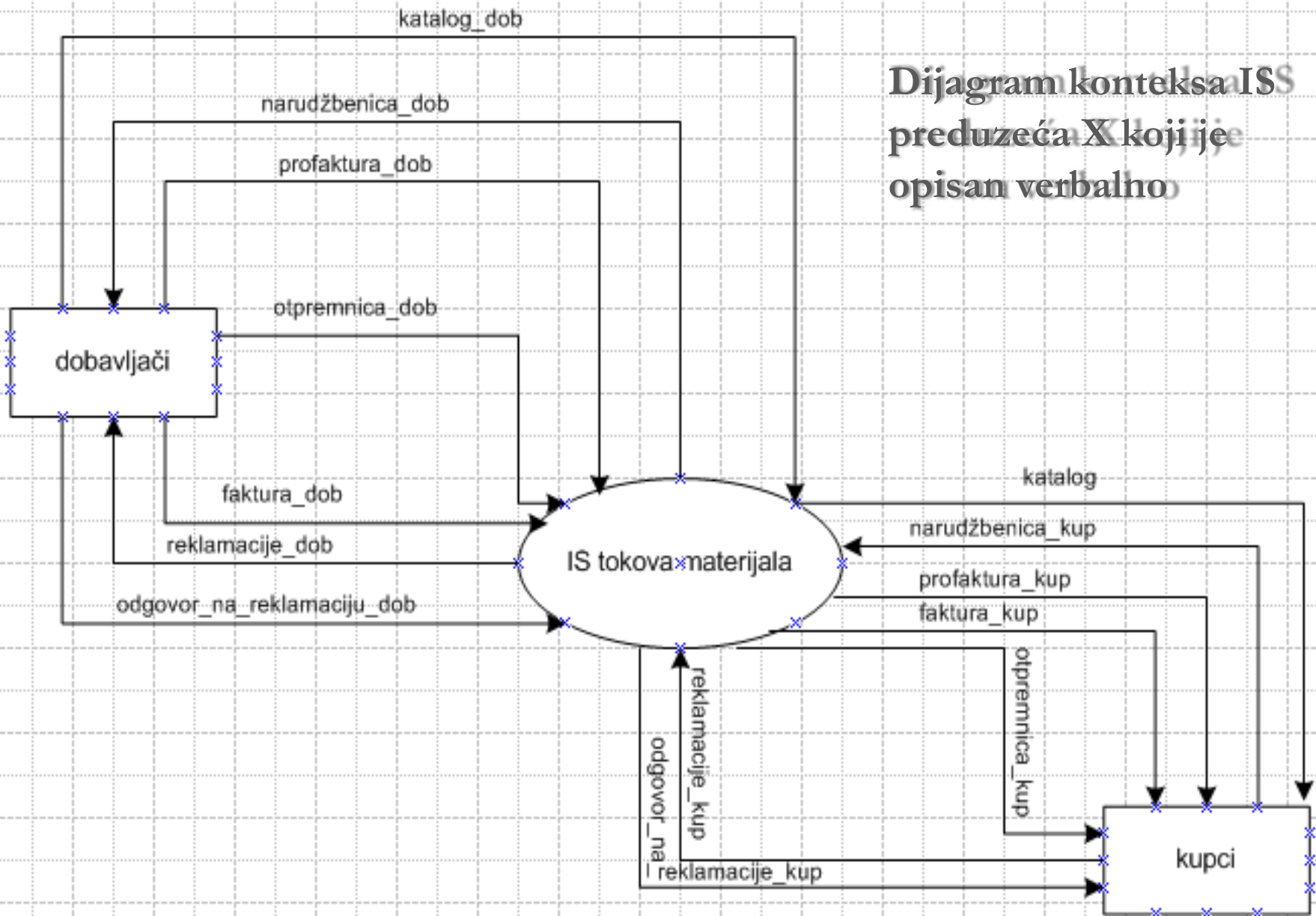
# PRIMER 1. VERBALNI OPIS SISTEMA

## Osnovne poslovne funkcije

1. Prodajna služba (opis...)
2. Proizvodnja (opis...)
3. Skladištenje (opis...)
4. Služba nabavke (opis...)
5. Razvojna služba (opis...)

*nastavak →*

Dijagram konteksta IS  
preduzeća X koji je  
opisan verbalno



# ZADATAK

---

- Potrebno je napraviti IS restorana koji posluje sa dobavljačima prehrambenih proizvoda i gostima. U okviru restorana postoje tri procesa: kuhinja koja posluje sa dobavljačem, usluga koja posluje sa gostima i finansije koje vrše isplate na osnovu računa koje primaju i izdaju drugi procesi.

- Kreiraj dijagram konteksta sa sledećim objektima:

Spoljni objekti : DOBAVLJAC, GOST;

Tokovi podataka : Cenovnik dobavljacka, Narudzbenica dobavljacu,  
Racun dobavljacka, Reklamacija dobavljacu,  
Uplata dobavljacu, Rezervacija gosta,  
Meni gostu, Porudzbina gosta, Racun gostu,  
Primedba gosta

# ZADATAK

---

- Kreirati dijagram dekompozicije prvog nivoa sa sledećim objektima:

Procesi: KUHINJA, USLUGA, FINANSIJE;

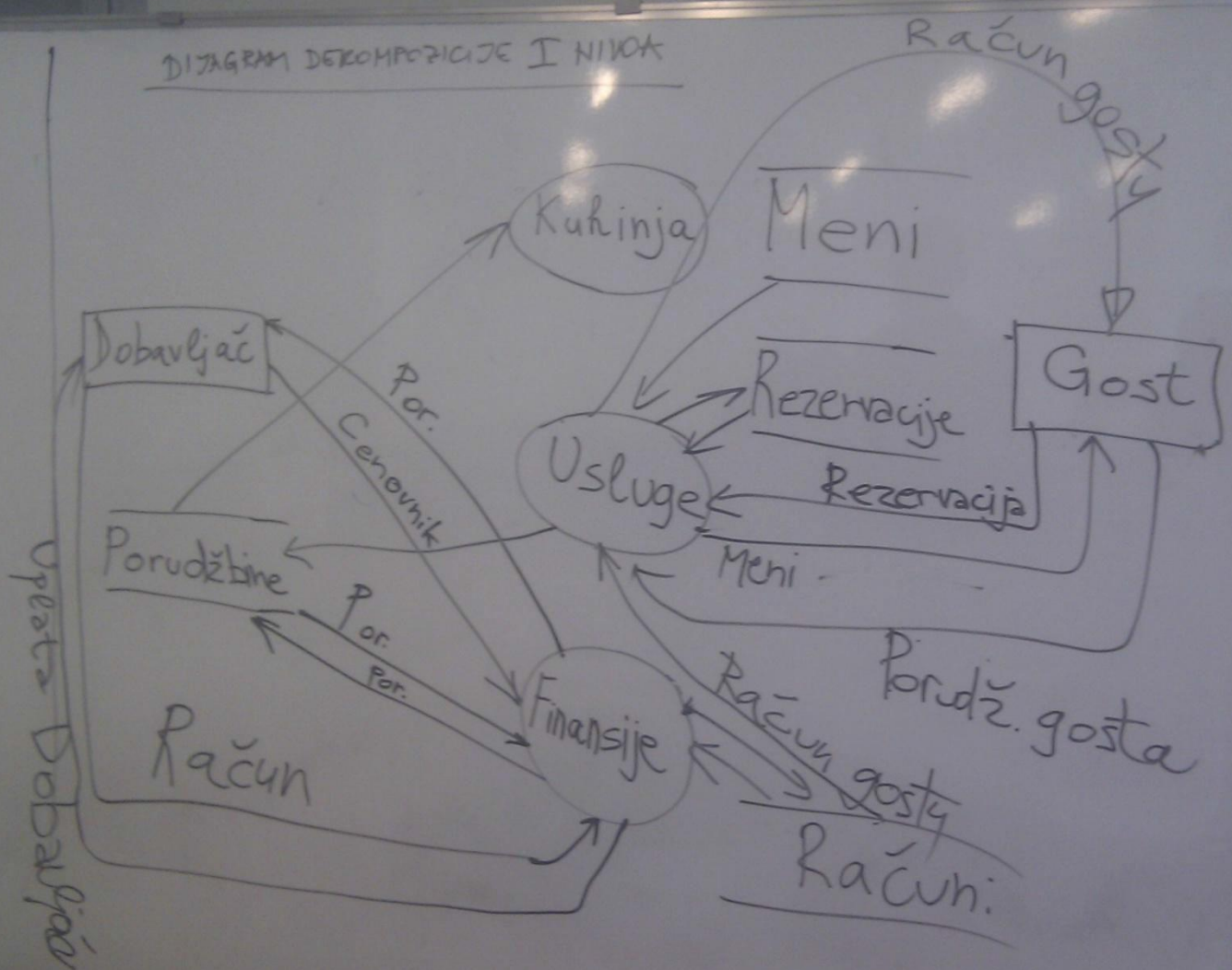
Spoljni objekti: DOBAVLJAC, GOST;

Skladišta: Porudzbine, Racuni, Rezervacije, Meni;

Korigovati model... npr nesto ovako...

DIJAGRAM DEKOMPOZICIJE I NIVOA

435T



Operater Dobavljač

Porudž. gosta  
Račun gosty  
Računi

# ANALIZA ZAHTEVA I SPECIFIKACIJA IS

---

## SSA – rečnik podataka

- opisuje strukturu i sadržaj svih tokova i skladišta podataka



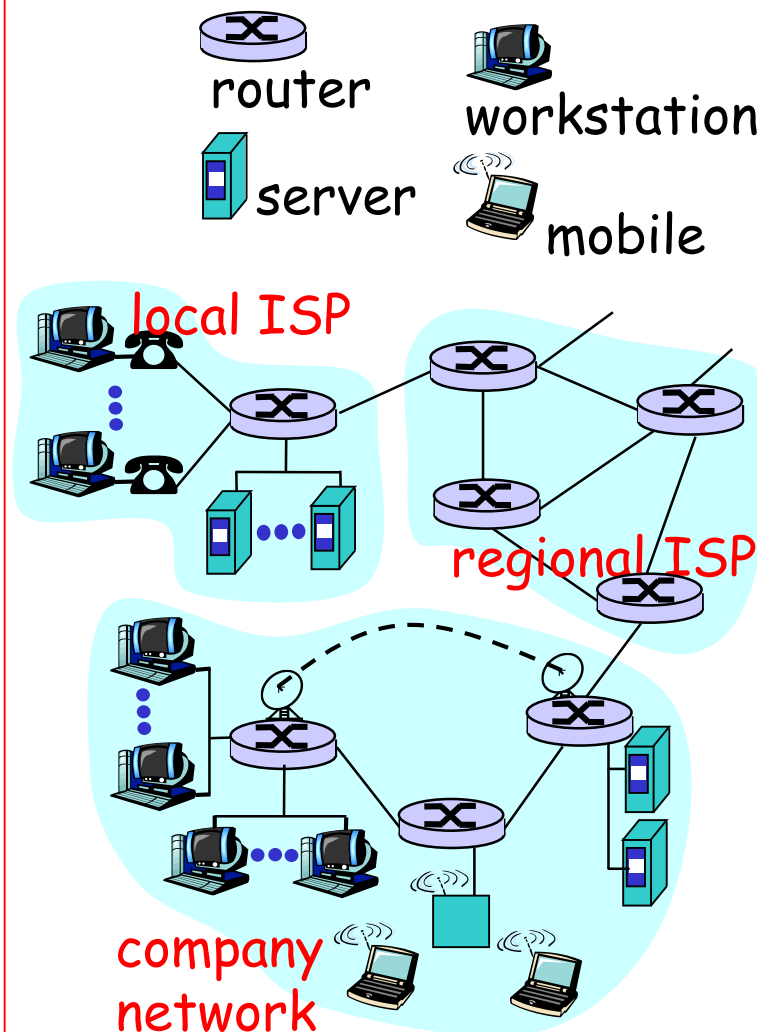
# POSLOVNA INFORMATIKA KOLOKVIJUM-2-ZADACI

- 1.1 Uvod
- 1.2. Mrežni sloj
- 1.3 Virtuelna kola i datagram mreže
- 1.4 Šta je u unutrašnjosti rutera
- 1.5 IP: Internet protokol
  - format datagrama
  - IPv4 adresiranje
  - IPv6
- 1.6 zadaci

- 2.1 Softversko inženjerstvo
- 2.2 metode i metodologije (UML)
- 2.3 Zadaci

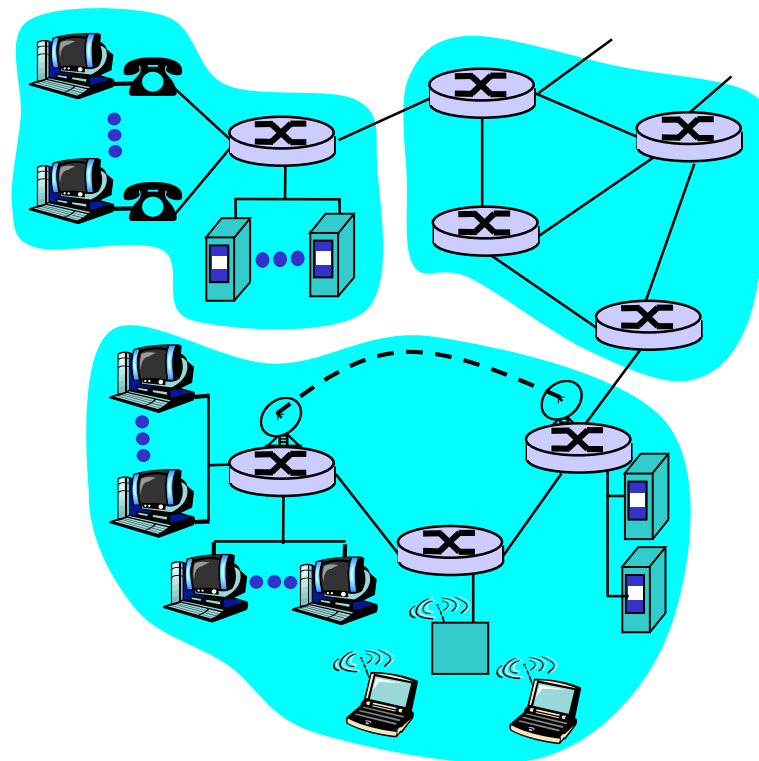
# 1.1. Uvod-Šta je Internet:

- ❑ milioni povezanih računarskih uređaja: *hosts = krajevi sistema*
- ❑ izvođenje *mrežne aplikacije*
- ❑ *komunikacioni linkovi*
  - optika, bakar, radio, satelit
  - brzina prenosa = *širina opsega*
- ❑ *ruteri*: prosleđivanje paketa (gomila podataka)
- ❑ *protokoli*: kontrolišu slanje, prijem poruka
  - TCP, IP, HTTP, FTP, PPP
- ❑ *Internet: "mreža svih mreža"*
  - slaba hijerarhija
  - javni Internet protiv privatnog intranet-a
- ❑ Internet standardi
  - RFC: Request for comments
  - IETF: Internet Engineering Task Force



# Šta je Internet: pogled pružanja servisa

- **komunikacije**  
*infrastruktura* omogućuje distribuirane aplikacije:
  - Web, e-mail, video igre, e-poslovanje, deljenje fajlova
- **komunikacioni servisi**  
*obezbeđeni aplikacijama*:
  - Bez uspostavljanja konekcije nepouzdana servisi
  - Konekciono-orijentisani pouzdani servisi



# Šta je protokol?

## Ljudski protokoli:

- "Koliko je sati?"
- "da vas pitam"
- upoznavanja

... specifične poruke poslate

... specifične aktivnosti preduzete kada se poruke prime, ili drugi događaji

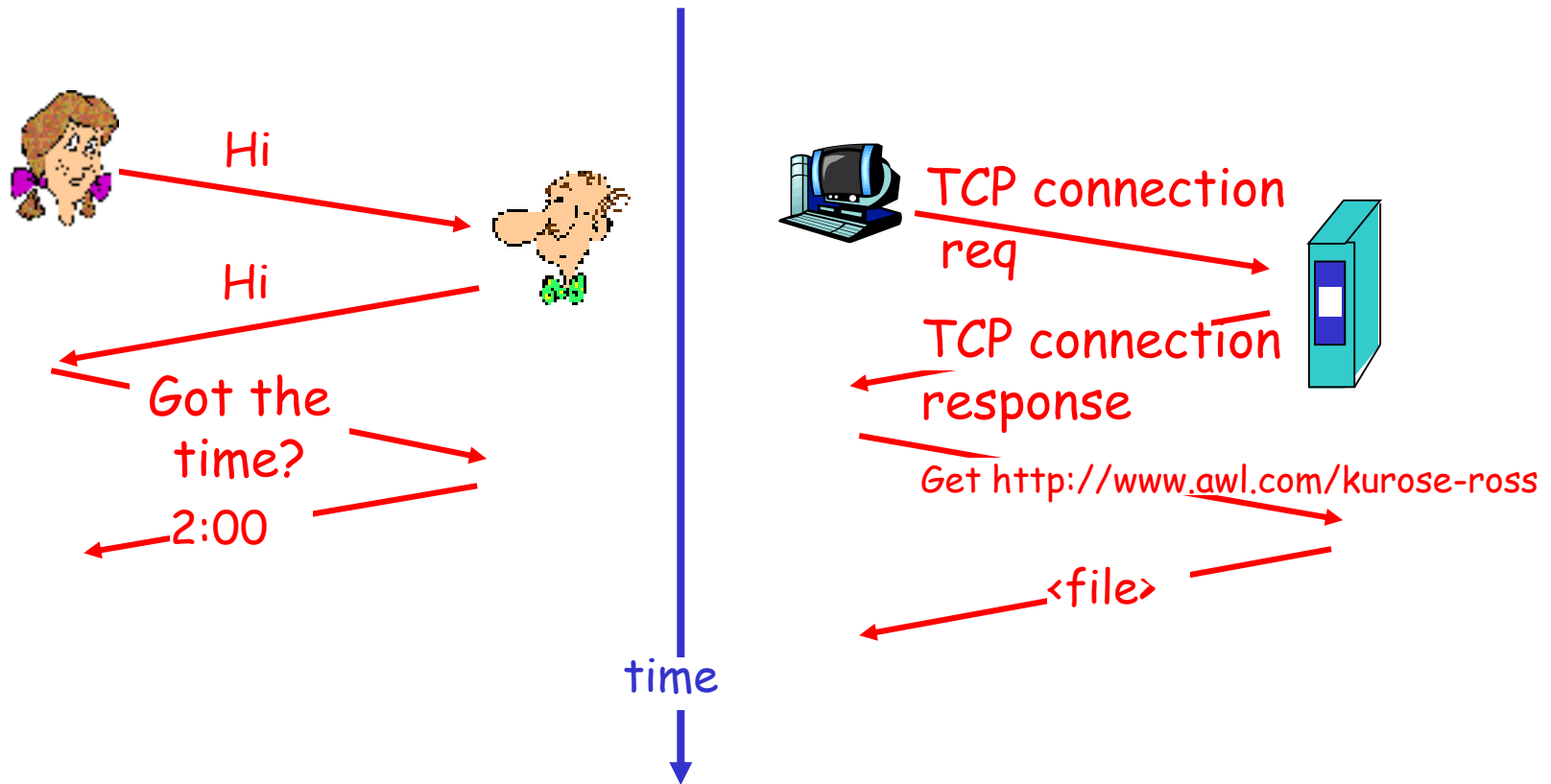
## Mrežni protokoli:

- mašine pre ljudi
- protokoli upravljaju svim aktivnostima prilikom komunikacija na Internet-u

*protokoli definišu format i redosled poruka poslatih i primljenih između mrežnih entiteta kao i akcije preduzete pri transmisiji i prijemu poruka*

# Šta je protokol?

Ljudski protokol i protokol računarskih mreža:



Q: Drugi ljudski protokoli?

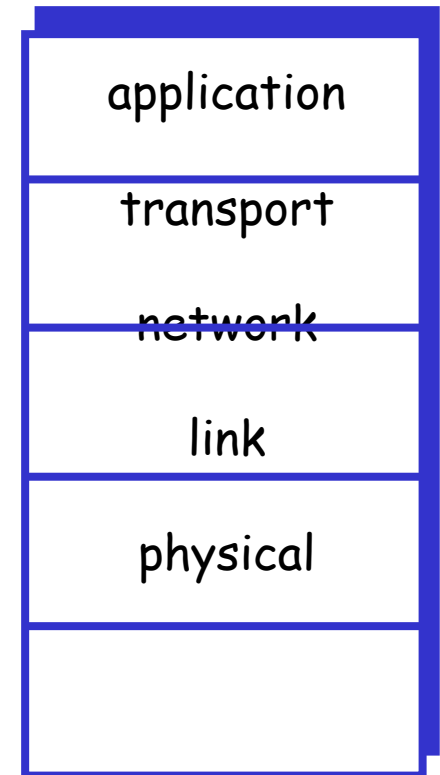
# Zašto slojevi?

Bavljenje kompleksnim sistemima:

- eksplicitne strukture dozvoljavaju identifikaciju, vezu delova kompleksnih sistema
  - uslojeni **model**
- modularizacija olakšava održavanje, ažuriranje sistema
  - promena primene servisa slojeva treba da bude transparentna ostatku sistema
  - npr. promena u gate-proceduri ne utiče na ostatak sistema

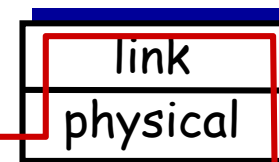
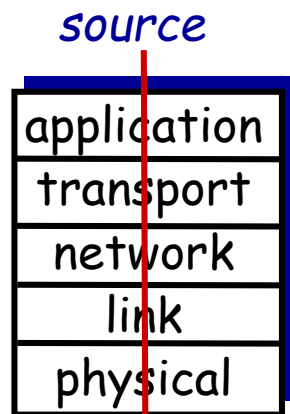
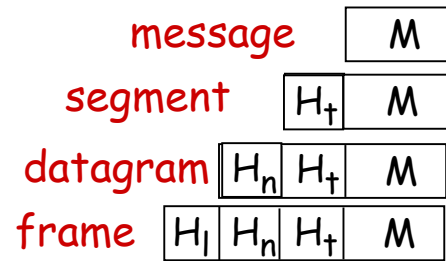
# Internet protokol stack

- **aplikacioni:** podržava mrežne aplikacije
  - FTP, SMTP, HTTP
- **transportni:** transfer podataka host-host
  - TCP, UDP
- **mrežni:** rutiranje datagrama od izvora do odredišta
  - IP, rutiranje protokoli
- **link:** transfer podataka između susjednih mrežnih elemenata
  - PPP, Ethernet
- **fizički:** bit-ovi "na žici"



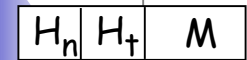
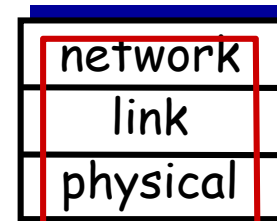
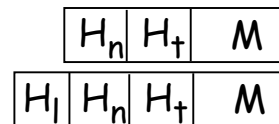
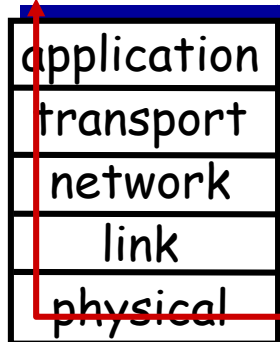


# Enkapsulacija



switch

*destination*

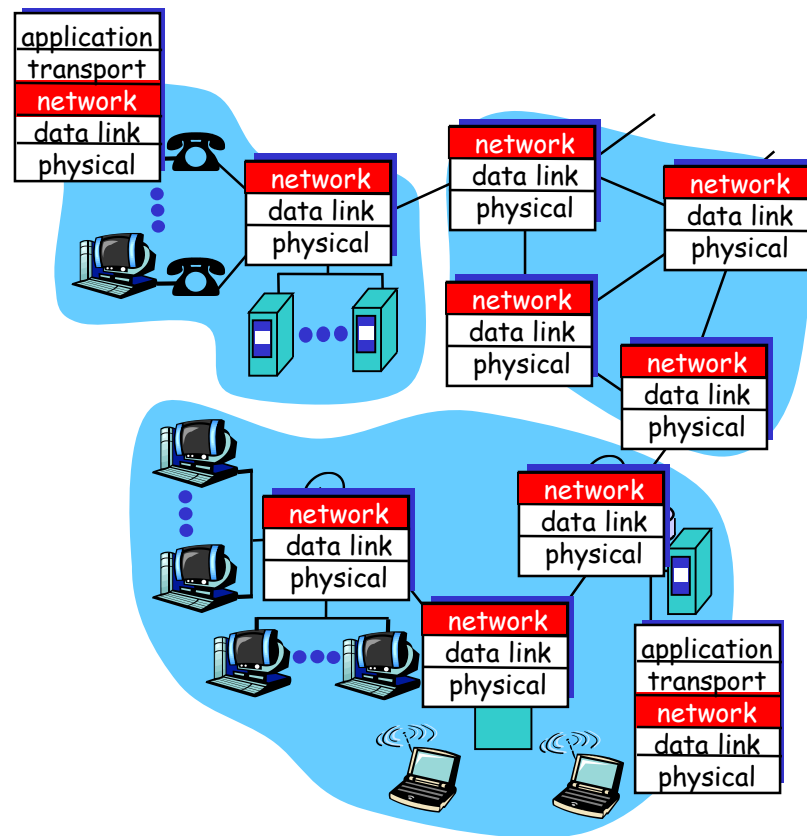


router

- 1.1 Uvod
- 1.2. Mrežni sloj
- 1.3 Virtuelna kola i datagram mreže
- 1.4 Šta je u unutrašnjosti rutera
- 1.5 IP: Internet protokol
  - format datagrama
  - IPv4 adresiranje
  - IPv6
- 1.6 zadaci
- 2.1 Softversko inženjerstvo
- 2.2 metode i metodologije (UML)
- 2.3 Zadaci

# 1.2. Mrežni sloj

- ❑ Transport segmenta od hosta koji ga šalje od hosta koji ga prima
- ❑ na strani pošiljaoca enkapsuliraju se segmenti u datagrame
- ❑ na prijemnoj strani, isporučuju se segmenti transportnom sloju
- ❑ protokoli mrežnog sloja na svakom hostu, ruteru
- ❑ Ruter ispituje polja u hederu svakog IP datagrama koje on propušta



# Ključne funkcije mrežnog sloja

□ *otpremanje-isporučivanje*: pomera pakete od ulaza u ruter do odgovarajućeg ruterovog izlaza (proces koji se odvija u samom ruteru)

□ *rutiranje*: određuje putanju (dobijenu od paketa) od izvora do odredišta.

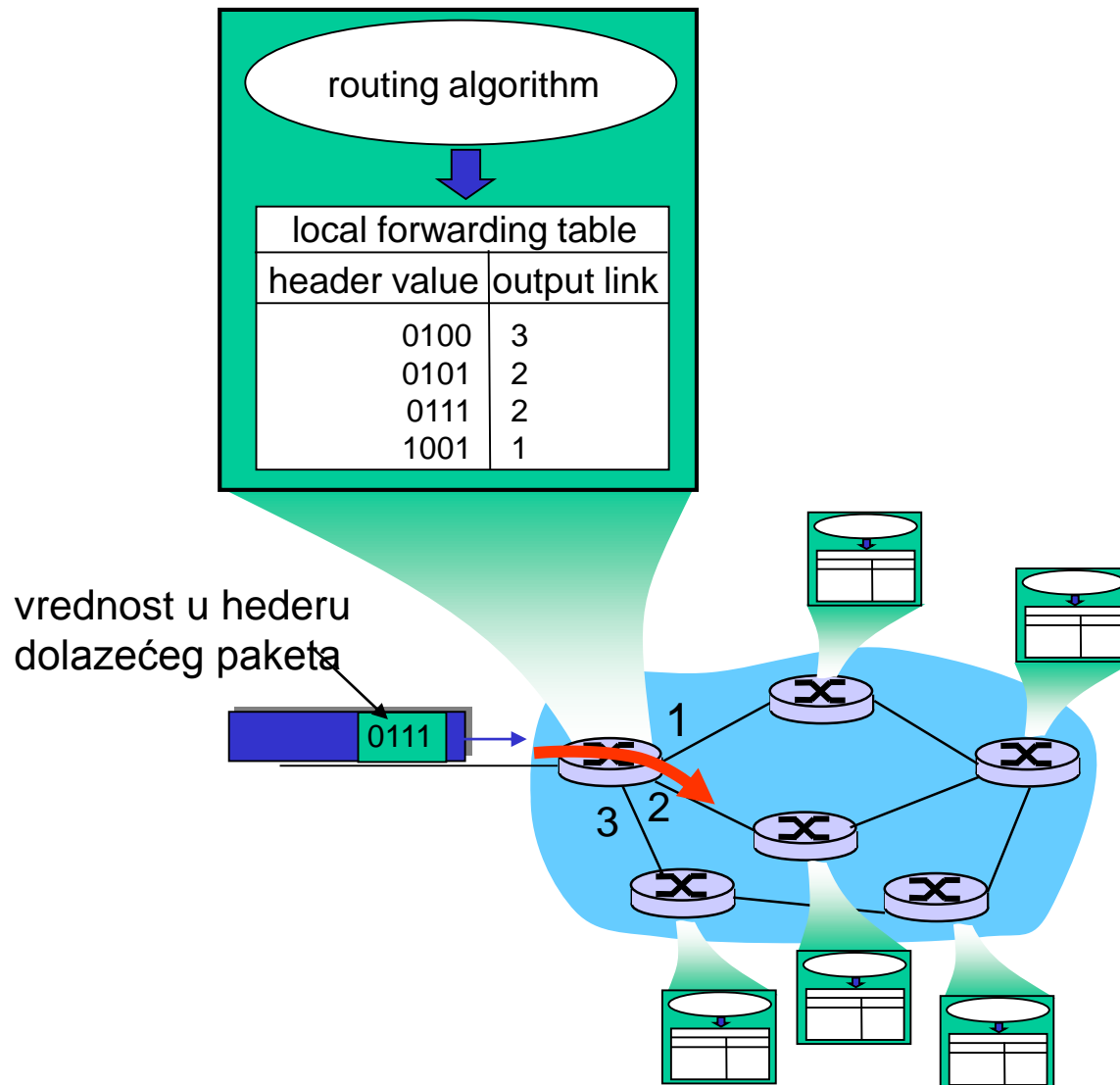
○ *Algoritmi rutiranja*

*analogija*:

□ *rutiranje*: proces planiranja putovanja od polaznog mesta do odredišta (lokalni karakter)

□ *isporučivanje*: proces prolaska-napredovanja kroz jedno mesto (globalni karakter)

# Veza između rutiranja i otpremanja



# Uspostavljanje konekcije

- ❑ treća važna funkcija u *nekim mrežnim arhitekturama (connection setup)*:
  - ATM, frame relay, X.25
- ❑ Pre početka slanja datagrama, dva hosta i uključeni ruteri **uspostavljaju virtuelnu konekciju**
  - Ruteri su uključeni
- ❑ Servis mrežnog i transportnog sloja:
  - **Mreža**: između dva hosta
  - **Transport**: između dva procesa

# Model mrežnog servisa

Q: Šta je *model servisa-usluge* za "kanal" koji transportuje datagrame od sendera do risivera?

## Primeri servisa za pojedinačne datagrame:

- garantovana isporuka
- garantovana isporuka sa ograničenim čekanjem (npr. manje od 100 msec kašnjenja)

## Primeri servisa za tok datagrama:

- isporuka datagrama po redu
- garantovana minimalna širina opsega za protok
- garantovani max. jitter  
-vreme između transmisije dva susedna paketa na senderu je približno jednako vremenu na prijemu

- 1.1 Uvod
- 1.2. Mrežni sloj
- 1.3 Virtuelna kola i datagram mreže
- 1.4 Šta je u unutrašnjosti rutera
- 1.5 IP: Internet protokol
  - format datagrama
  - IPv4 adresiranje
  - IPv6
- 1.6 zadaci

- 2.1 Softversko inženjerstvo
- 2.2 metode i metodologije (UML)
- 2.3 Zadaci



# Konekcija mrežnog sloja i servisi bez uspostavljanja konekcije

- **Datagram mreže** obezbeđuju servise bez uspostavljanja konekcije na mrežnom sloju
- **VC mreže** obezbeđuju servise uspostavljanja konekcije na mrežnom sloju

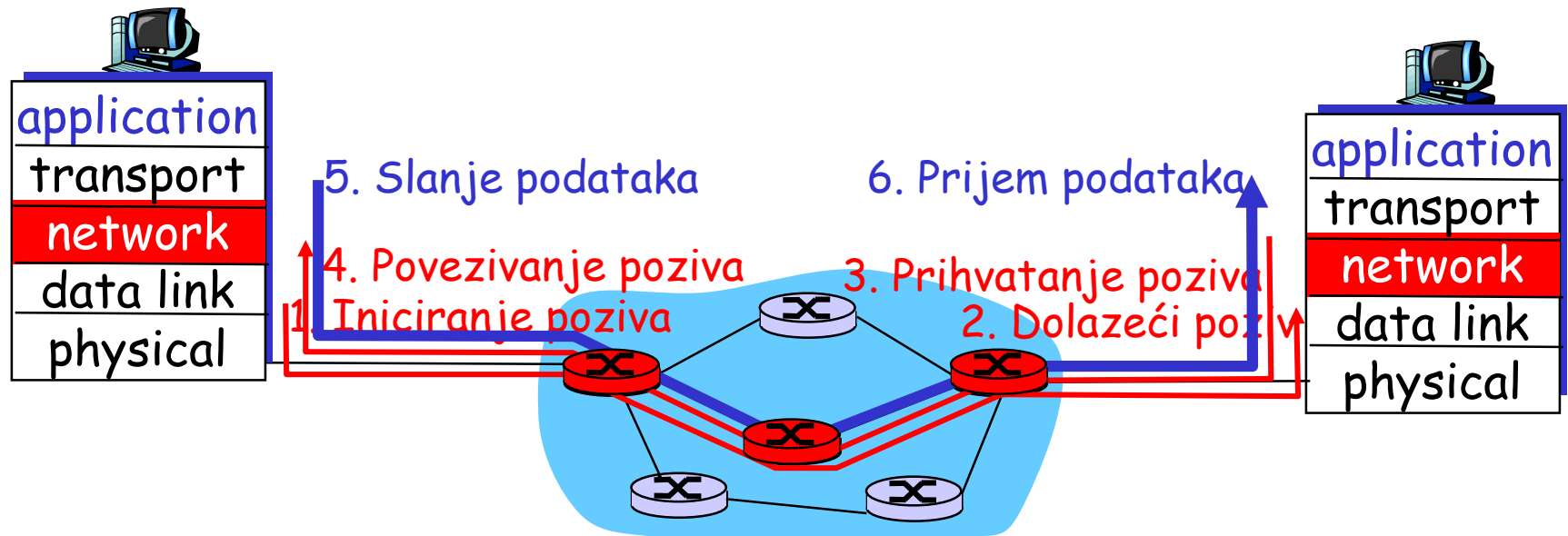
# Virtuelna kola

“putanja izvorište-odredište se ponaša kao telefonsko kolo”

- performanse
- akcije mreže na putanji izvorište-odredište

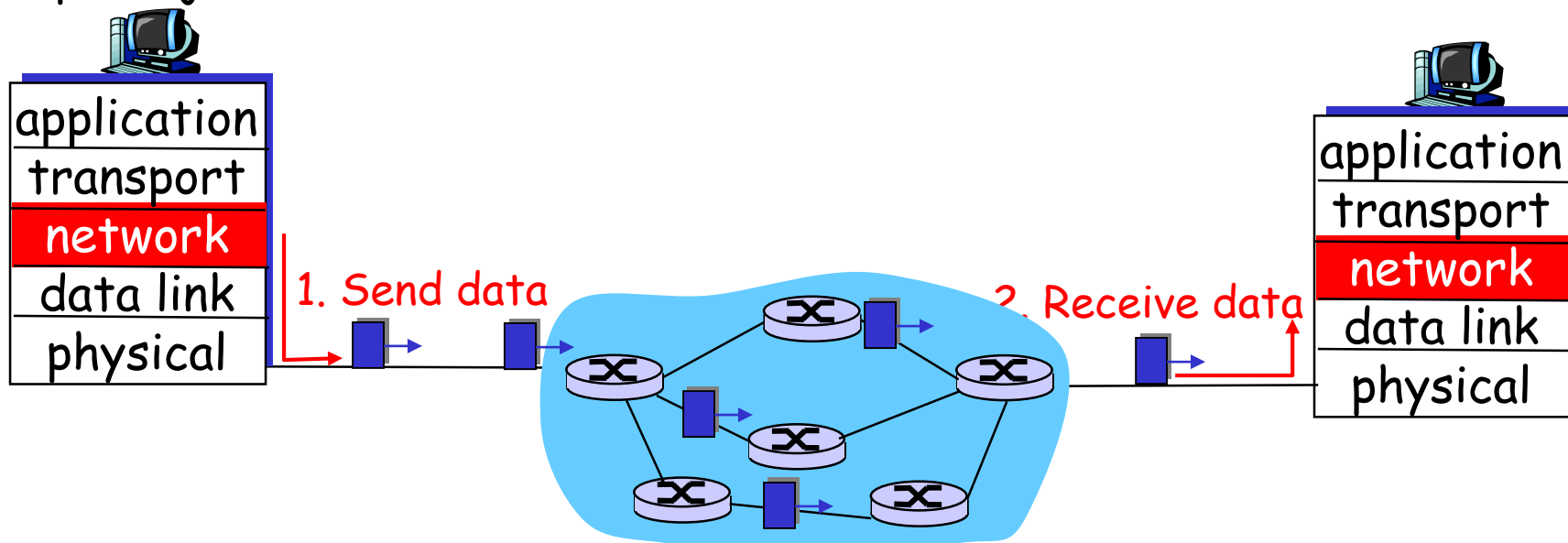
# Virtuelna kola: protokoli signalizacije

- ❑ koriste se za setap, održavanje i teardown VC (završetak)
- ❑ koriste se u ATM, frame-relay, X.25
- ❑ ne koriste se na sadašnjem Internet-u



# Datagram mreže

- nema setap poziva na mrežnom sloju
- ruteri: ne održavaju informacije o stanju konekcija od kraja do kraja
  - nema koncepta "konekcije" mrežnog sloja
- paketi se prosleđuju koristeći adresu odredišnog hosta
  - paketi između istog para izvor-destinacija mogu da imaju različite putanje



# Tabela prosleđivanja

4 milijarde  
mogućih upisa

<u>Opseg adresa destinacija</u>	<u>Link Interface</u>
11001000 00010111 00010000 00000000 through 11001000 00010111 00010111 11111111	0
11001000 00010111 00011000 00000000 through 11001000 00010111 00011000 11111111	1
11001000 00010111 00011001 00000000 through 11001000 00010111 00011111 11111111	2
otherwise	3

# Podešavanje sa najdužim prefix-om

<u>Prefix Match</u>	<u>Link Interface</u>
11001000 00010111 00010	0
11001000 00010111 00011000	1
11001000 00010111 00011	2
otherwise	3

## Primeri

DA: 11001000 00010111 00010110 10100001

Koji interfejs?

DA: 11001000 00010111 00011000 10101010

Koji interfejs?

# Datagram mreža ili VC mreža: zašto?

## Internet

- ❑ razmena podataka između računara
  - "elastic" servisi, ne zahteva se striktni tajming-vremenska ograničenja
- ❑ "pametni" krajevi sistema (računari)
  - može da adaptira, izvršava kontrolu, popravlja greške
  - prost unutar mreže, kompleksan na "ivici"
- ❑ više tipova linkova
  - različite karakteristike

## ATM

- ❑ razvijen iz telefonije
- ❑ govorna komunikacija ljudi:
  - striktni tajming, pouzdanost
  - potreba garantovanog servisa
- ❑ krajevi sistema
  - telefoni
  - kompleksnost unutar mreže

- 1.1 Uvod
- 1.2. Mrežni sloj
- 1.3 Virtuelna kola i datagram mreže
- 1.4 Šta je u unutrašnjosti rutera
- 1.5 IP: Internet protokol
  - format datagrama
  - IPv4 adresiranje
  - IPv6
- 1.6 zadaci

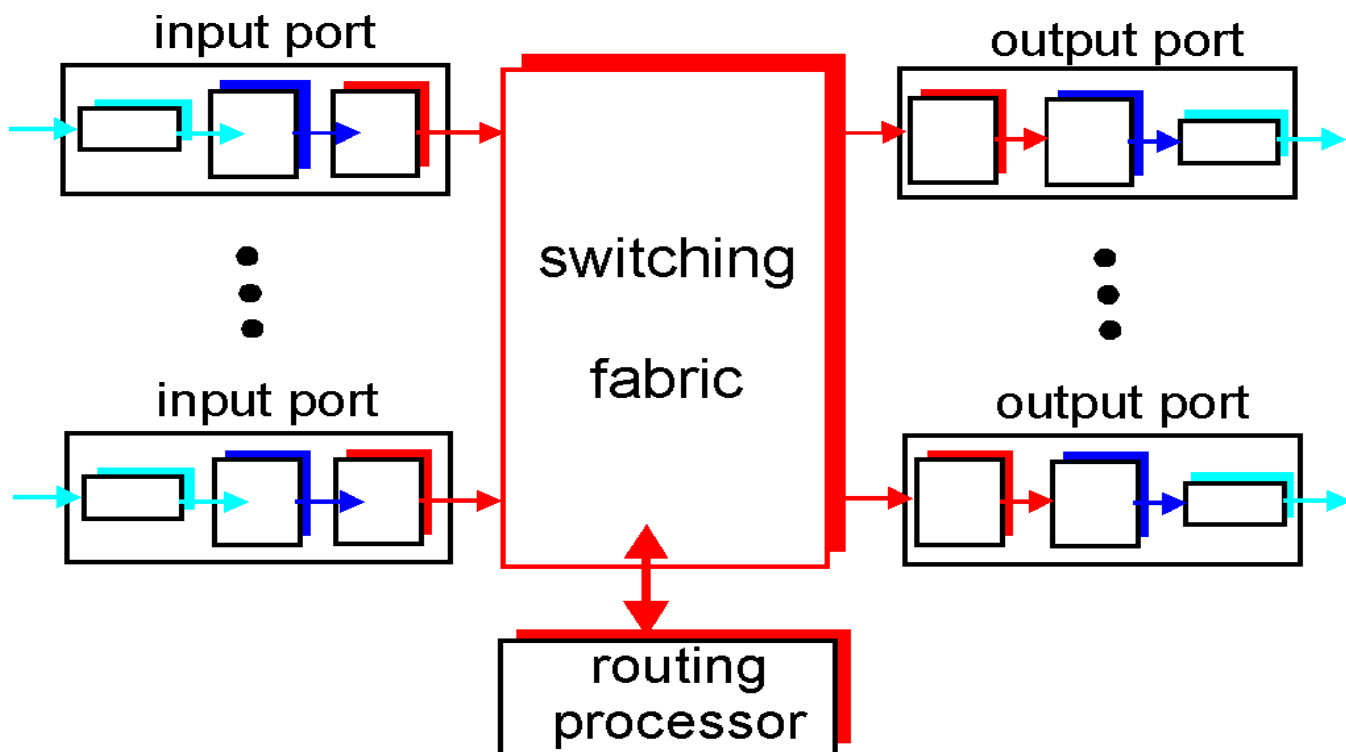
- 2.1 Softversko inženjerstvo
- 2.2 metode i metodologije (UML)
- 2.3 Zadaci



## 4.3. Pregled arhitekture rutera

Dve ključne funkcije rutera:

- *startuje* rutinng algoritme/protokole (RIP, OSPF, BGP)
- *Prosleđuje* datagrame od dolaznog do odlaznog linka

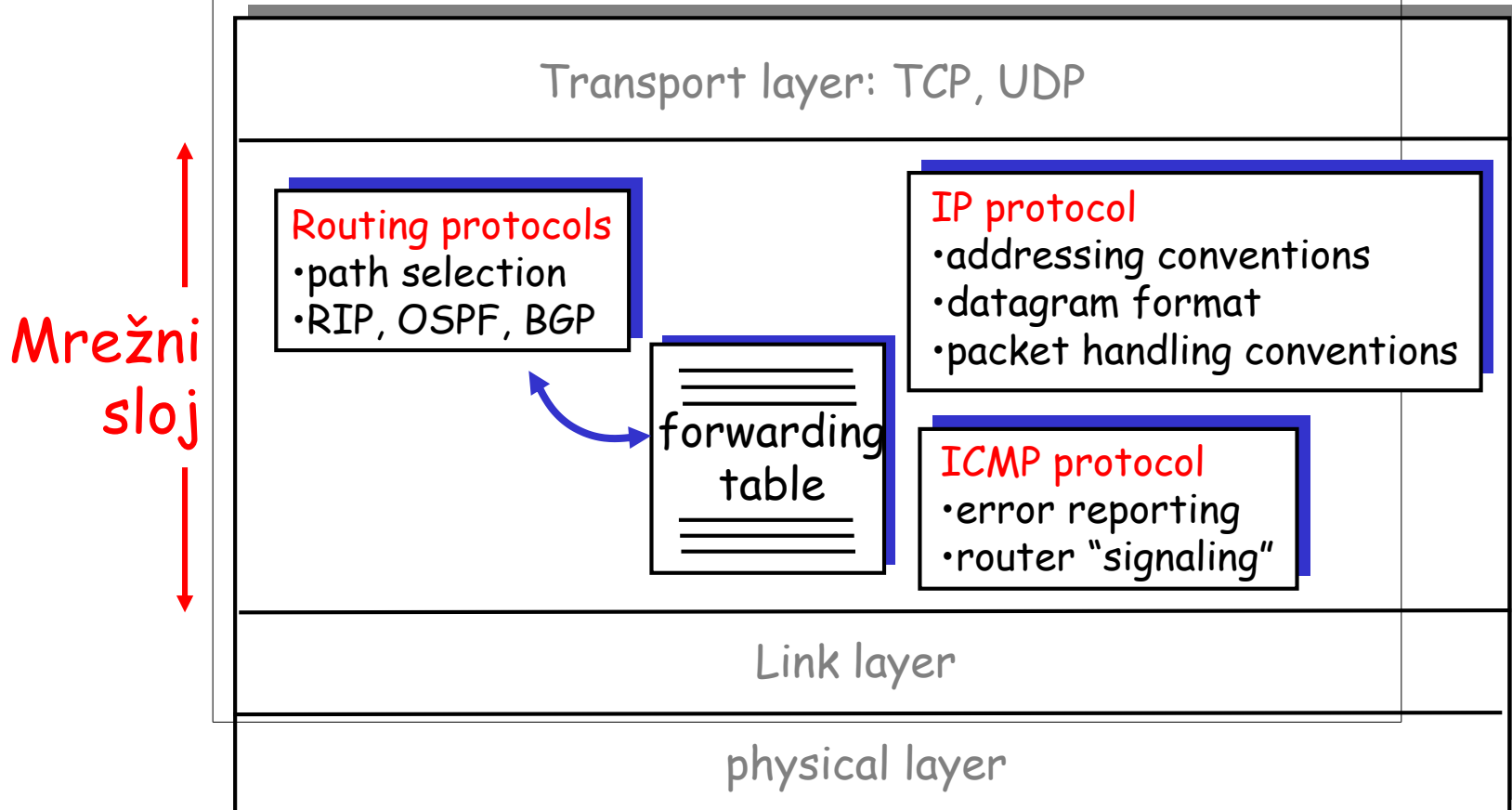


- 1.1 Uvod
- 1.2. Mrežni sloj
- 1.3 Virtuelna kola i datagram mreže
- 1.4 Šta je u unutrašnjosti rutera
- 1.5 IP: Internet protokol
  - format datagrama
  - IPv4 adresiranje
  - IPv6
- 1.6 zadaci

- 2.1 Softversko inženjerstvo
- 2.2 metode i metodologije (UML)
- 2.3 Zadaci

# Internet mrežni sloj

## Funkcije mrežnog sloja hosta i rutera



- 1.1 Uvod
- 1.2. Mrežni sloj
- 1.3 Virtuelna kola i datagram mreže
- 1.4 Šta je u unutrašnjosti rutera
- 1.5 IP: Internet protokol
  - format datagrama
  - IPv4 adresiranje
  - IPv6
- 1.6 zadaci

- 2.1 Softversko inženjerstvo
- 2.2 metode i metodologije (UML)
- 2.3 Zadaci

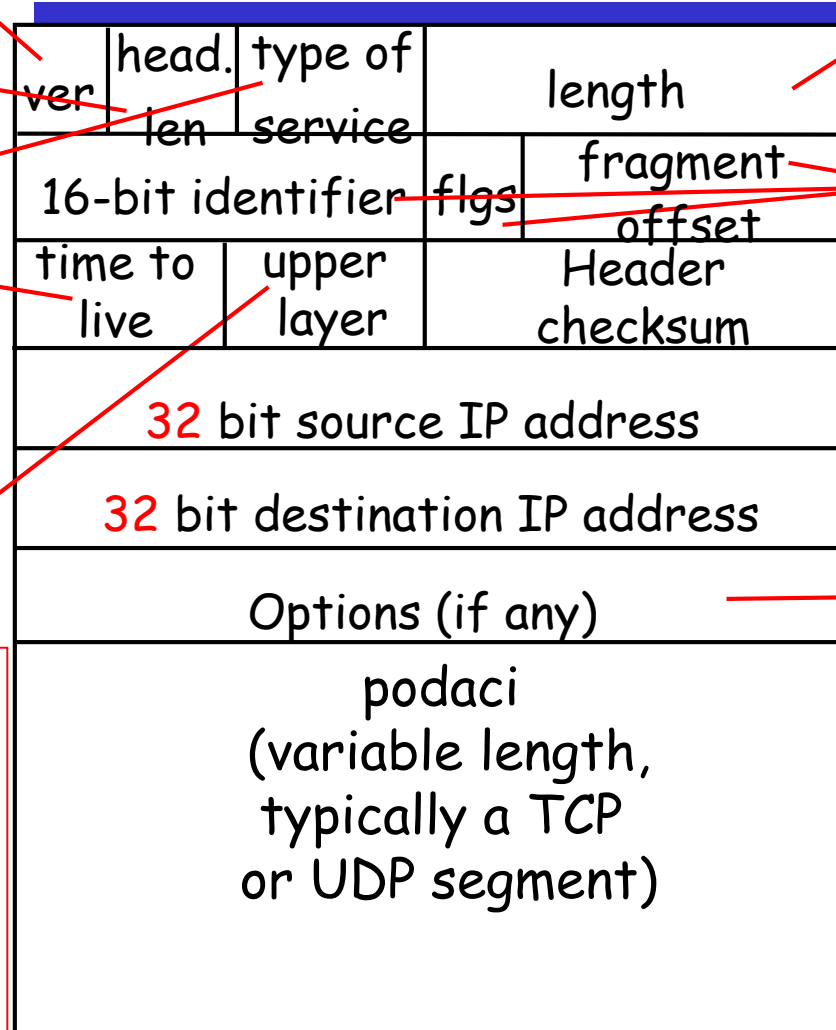
# Format IP datagrama

Broj verzije  
IP protokola  
dužina zaglavlja  
(bytes)  
"TOS"

max number  
remaining hops  
(decremented at  
each router)

protokol gornjeg sloja  
kome se isporučuje payload

← 32 bits →



ukupna dužina  
datagrama (bytes)

for  
fragmentation/  
reassembly

npr. vremenska markica,  
tokena zapisa putanje,  
određuje listu rutera

## Overhead sa tcp strukturuom

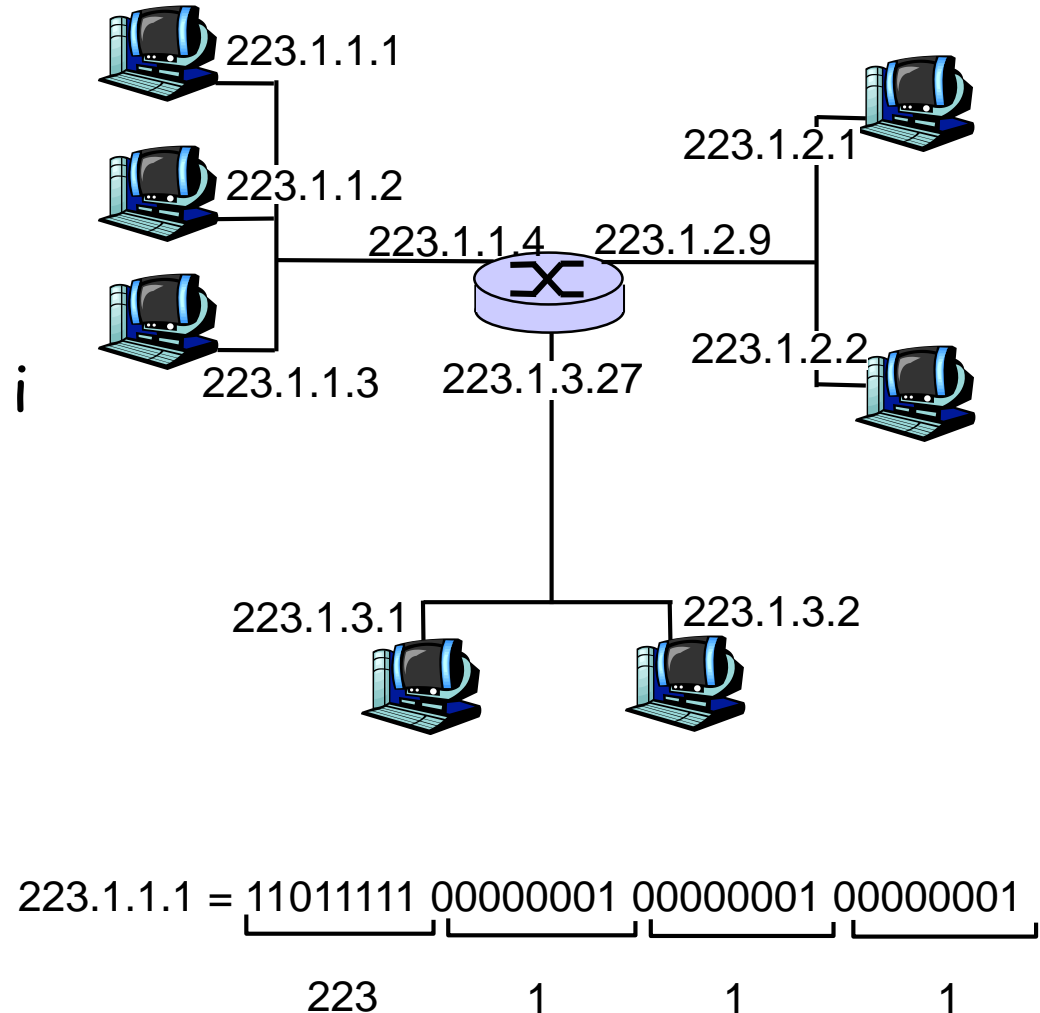
- 20 bytes of TCP
- 20 bytes of IP
- = 40 bytes + app layer message - overhead

- 1.1 Uvod
- 1.2. Mrežni sloj
- 1.3 Virtuelna kola i datagram mreže
- 1.4 Šta je u unutrašnjosti rutera
- 1.5 IP: Internet protokol
  - format datagrama
  - IPv4 adresiranje
  - IPv6
- 1.6 zadaci

- 2.1 Softversko inženjerstvo
- 2.2 metode i metodologije (UML)
- 2.3 Zadaci

# IP adresiranje: uvod

- IP adresa: 32-bit-ni identifikator za host, ruter *interfejs*
- *interfejs*: konekcija između host-a/router-a i fizičkog linka
  - ruteri tipično imaju više interfejsa
  - host može imati više interfejsa
  - IP adrese su pridružene svakom interfejsu

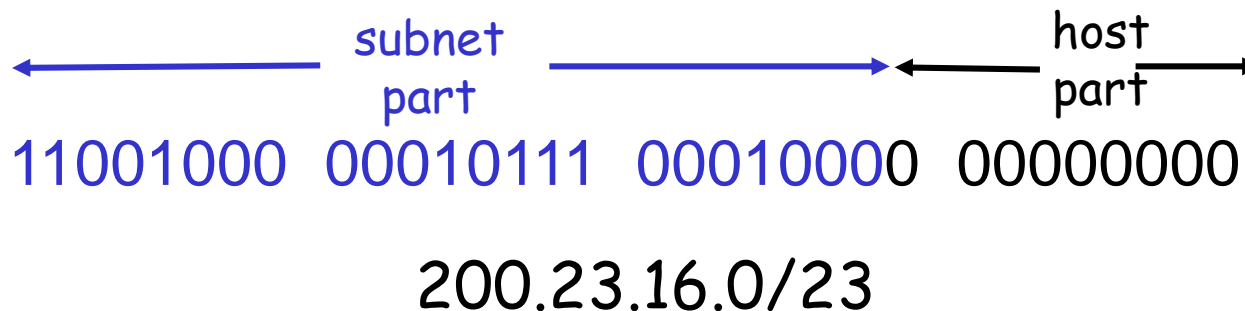


približno 4 milijarde mogućih  
IP adresa

# IP adresiranje: CIDR

## CIDR: Classless InterDomain Routing

- podmrežni deo adrese
- format adrese: **a.b.c.d/x**, gde je x broj bitova u prvom delu adrese - podmrežnom delu adrese - prefiks





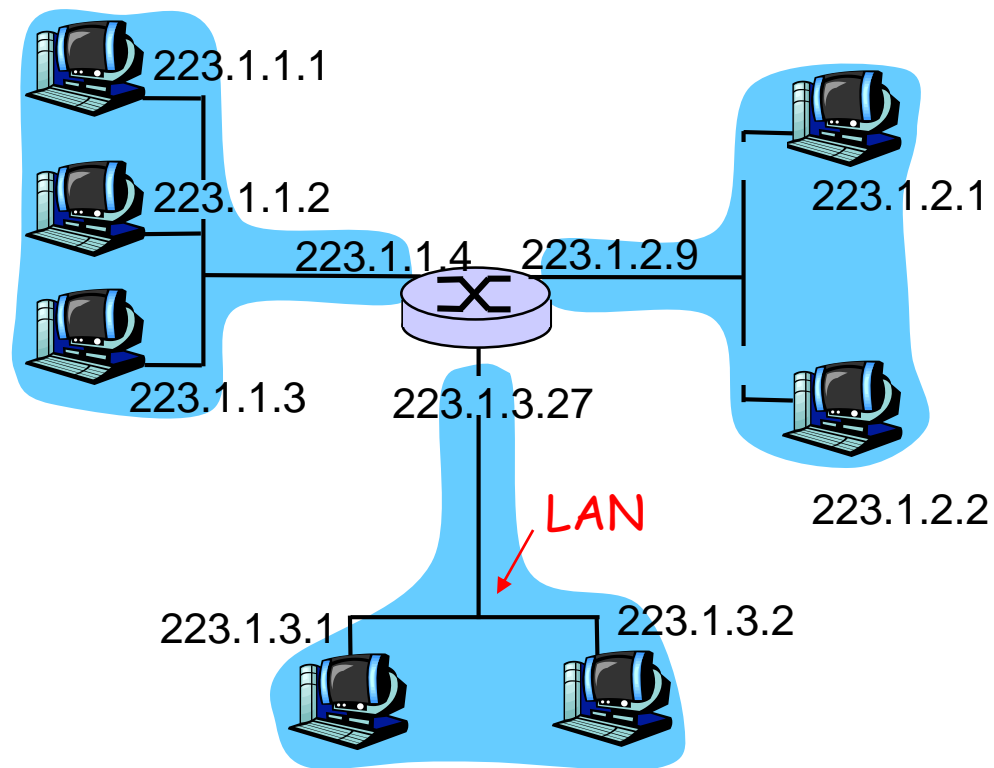
# Podmreže

## □ IP adresa:

- deo za podmrežu (bitovi višeg reda)
- deo za host (bitovi nižeg reda)

## □ Šta je podmreža ?

- interfejsi uređaja sa istim podmrežnim delom IP adrese
- interfejsi su povezani preko ethernet hub-a ili switch-a

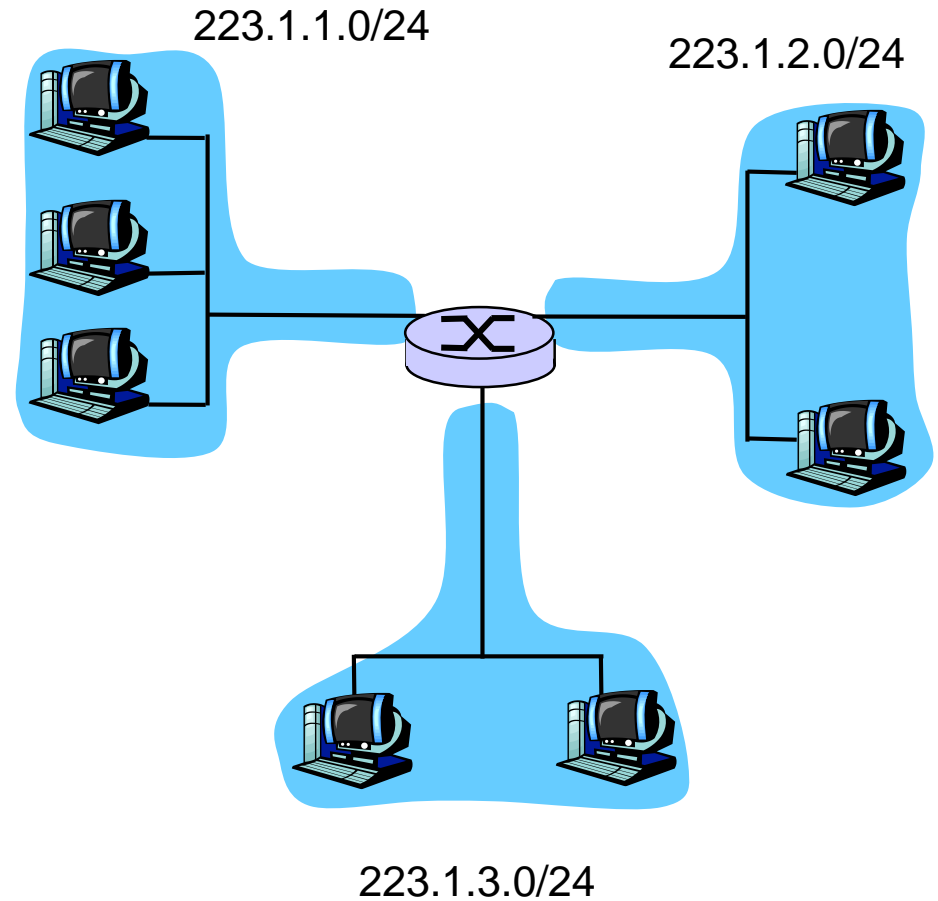


mreža sastavljena od 3 podmreže

# Podmreže

## Prijem

- Da bi odredili podmreže, treba rastaviti svaki interfejs od njegovog hosta ili rutera, kreirajući ostrva izolovanih mreža. Svaka izolovana mreža se zove **podmreža**.
- /24 maska podmreže pokazuje da leva 24 bita od 32 definišu adresu podmreže
- pre usvajanja CIDR-a, classful addressing:
  - 8, 16, 24 podmreže - klase: A, B (/16, 65534), C (/24, 254 hosta) mreže (pogledati sledeći slajd)



Maska podmreže: /24

# Adresni prostor pre usvajanja **CIDR-a**

- starija klasifikacija u odnosu na CIDR
- x**-evi definišu Netid (identifikuje mrežu)
- y**-oni definišu Hostid (identifikuje host u mreži)
  - Klasa-Class A: 0**xxxxxxx**.yyyyyyyyy.yyyyyyyyyy.yyyyyyyyyy
  - Klasa-Class B: 10**xxxxxx**.**xxxxxxx**.yyyyyyyyy.yyyyyyyyyy
  - Klasa-Class C: 110**xxxxx**.**xxxxxxxx**.**xxxxxxxx**.yyyyyyyyy
  - Klasa-Class D: 1110.....multicast address.....
- Klasa se definiše sa nekoliko prvih bitova u prvom bajtu
- Multicast (klasa D) - identifikuje grupe uređaja na mreži
- Unicast - identifikuje jedan uređaj na mreži (klasa A, B, C)

- 1.1 Uvod
- 1.2. Mrežni sloj
- 1.3 Virtuelna kola i datagram mreže
- 1.4 Šta je u unutrašnjosti rutera
- 1.5 IP: Internet protokol
  - format datagrama
  - IPv4 adresiranje
  - IPv6
- 1.6 zadaci

- 2.1 Softversko inženjerstvo
- 2.2 Metode i metodologije (UML)
- 2.3 Zadaci

# IPv6

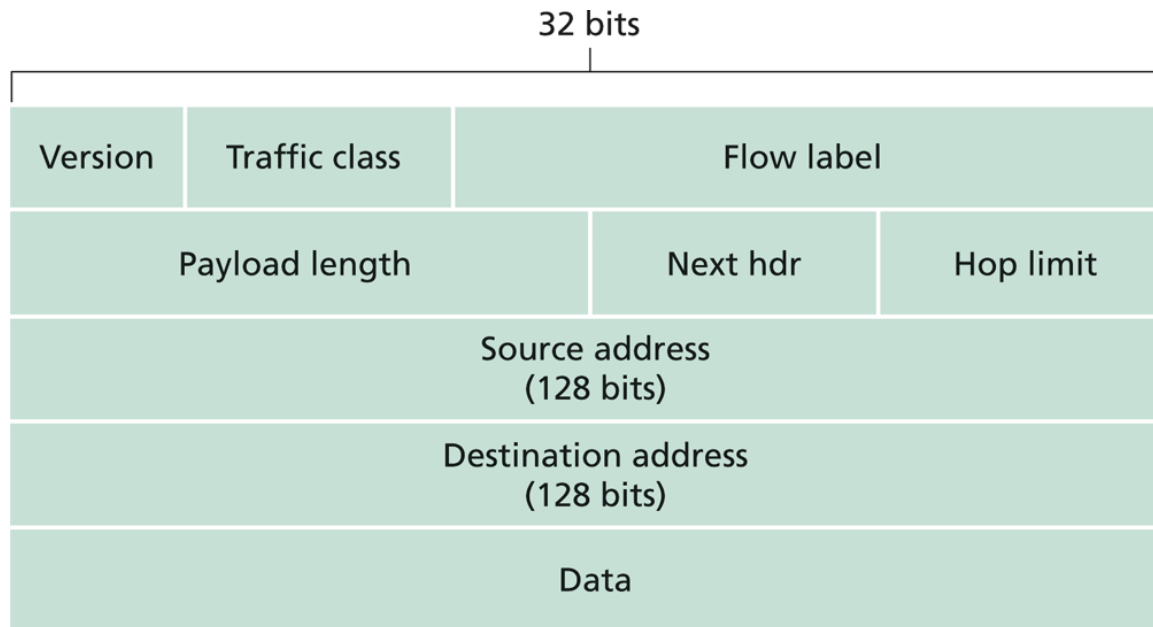
- ❑ **Početna motivacija:** 32-bitni adresni prostor biće uskoro potpuno dodeljen -IP<sub>new generation</sub>- koristi se 128 bit-ni adresni prostor,  $3.4 \times 10^{38}$  čvorova.
- ❑ **Dodatna motivacija:**
  - format zaglavlja pomaže brzom procesiranju/prosleđivanju
  - heder se menja da bi omogućio QoS real-time servisa
- ❑ **Format IPv6 (RFC 2460) datagrama:**
  - fiksne dužine hedera od 40 byt-ova
  - nije dozvoljena fragmentisanje između rutera;
    - to može biti urađeno samo na izvoru i odredištu; usporava se sistem podelom (i ponovnim spajanjem) na manje IP datagrame

# IPv6 zaglavlje

**Prioritet-klasa saobraćaja:** identifikuje prioritet između datagrama koji se prenose istim tokom, npr. ICMP.

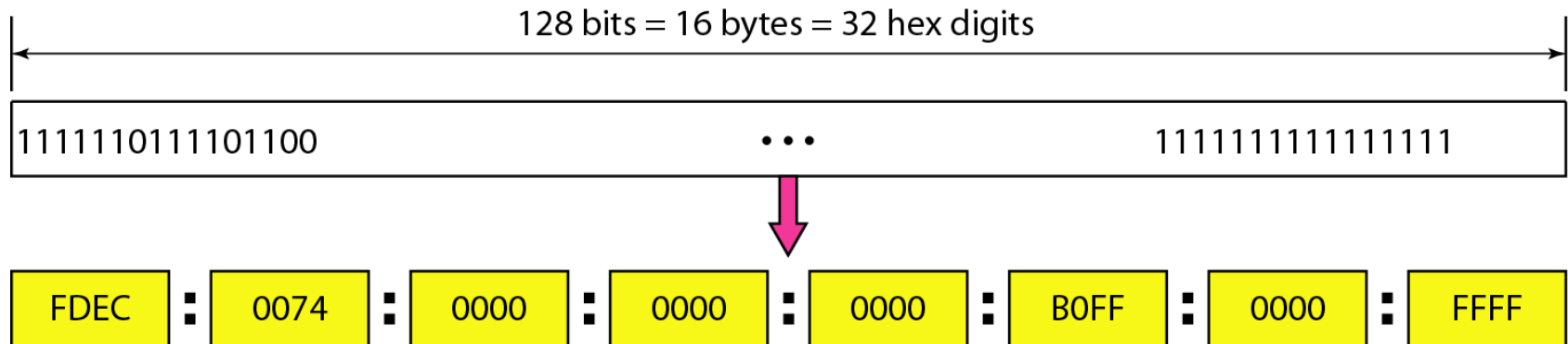
**Oznaka toka:** identifikuje datagrame u istom "toku" (još nije utvrđeno precizno značenje toka-"flow").

**Sledeće zaglavlje:** identifikuje protokol kome će se sadržaji datagrama isporučiti (npr. TCP ili UDP)



**Figure 4.22** ♦ IPv6 datagram format

# *IPv6 address in binary and hexadecimal colon notation*



# Druge promene u odnosu na IPv4

- ❑ **Checksum:** uklanja se potpuno da bi se redukovalo vreme procesiranja u svakom skoku
- ❑ **Opcije:** dozvoljene, ali izvan hedera, indicirano poljem "Next Header"
  - fiksne dužine hedera od 40 byt-ova (bez option field)
- ❑ **ICMPv6:** nova verzija ICMP-a
  - dodatni tipovi poruka, npr. "Packet Too Big"
  - funkcije upravljanja multicast grupom (InternetGroupManagementProtocol)



# Prelazak sa IPv4 na IPv6

- ❑ Ne mogu svi ruteri da budu nadograđeni istovremeno
  - nema "flag day" - određeni datum i vreme kada bi se sve mašine na Internetu isključile zbog prelaska IPv4 na IPv6 (od 1995 god., oko 15 god. prelazak)
  - Kako će mreža da radi sa izmešanim IPv4 i IPv6 ruterima ?
- ❑ **Dvostruki stek:** IPv6 čvorovi imaju takođe i kompletnu implementaciju protokola IPv4
- ❑ **Tunelovanje:** IPv6 se zbrinjava kao payload u IPv4 datagramu između IPv4 rutera

- 1.1 Uvod
- 1.2. Mrežni sloj
- 1.3 Virtuelna kola i datagram mreže
- 1.4 Šta je u unutrašnjosti rutera
- 1.5 IP: Internet protokol
  - format datagrama
  - IPv4 adresiranje
  - IPv6
- 1.6 zadaci

- 2.1 Softversko inženjerstvo
- 2.2 metode i metodologije (UML)
- 2.3 Zadaci

# 1.6. Zadaci

## Zadatak 1.

Odrediti klase datih IP adresa (koristeći klasno adresiranje-stariju notaciju adresnog prostora), koristiti tabele sa slajdova 45 i 46:

- a. 230.12.14.85      b. 200.14.56.20      c. 18.23.120.9  
d. 241.5.15.101      e. 190.11.78.46

Rešenje:

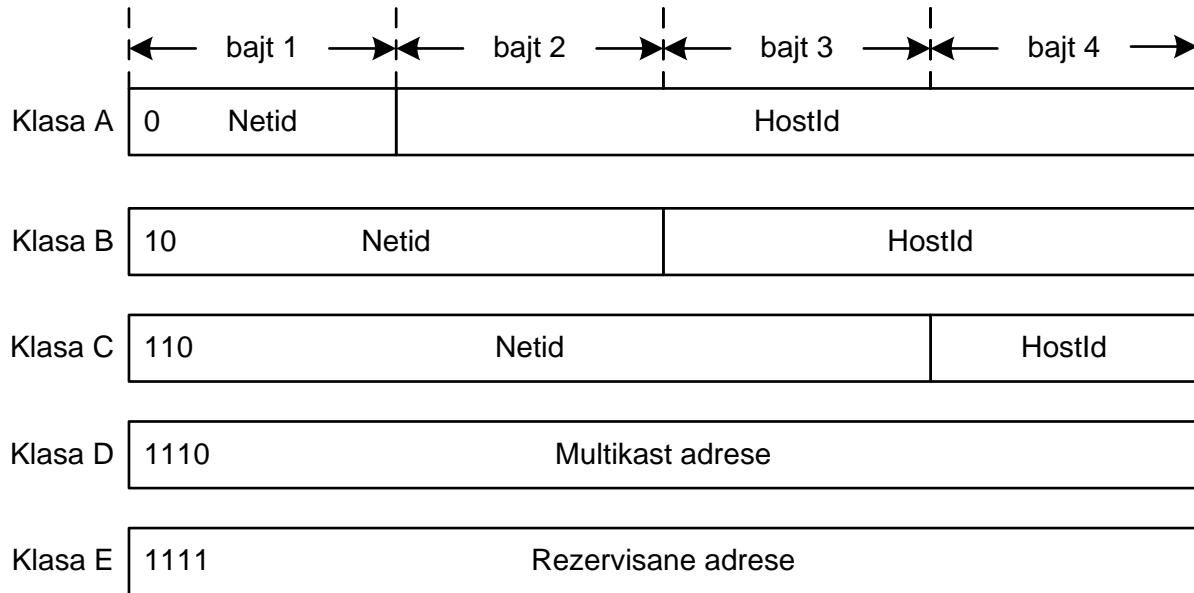
- a. Prvi bajt je 230 (između 224 i 239) - klasa D.  
b. Prvi bajt je 200 (između 192 i 223) - klasa C.  
c. Prvi bajt je 18 (između 0 i 127) - klasa A.  
d. Prvi bajt je 241 (između 240 i 255) - klasa E.  
e. Prvi bajt je 190 (između 128 i 191) - klasa B.

# klase IP adresa-starija notacija

Ako je adresa data u decimalnoj notaciji:

	Prvi bajt	Drugi bajt	Treći bajt	Četvrti bajt
Klasa A	0 - 127			
Klasa B	128 - 191			
Klasa C	192 - 223			
Klasa D	224 - 239			
Klasa E	240 - 255			

# Netid i Hostid



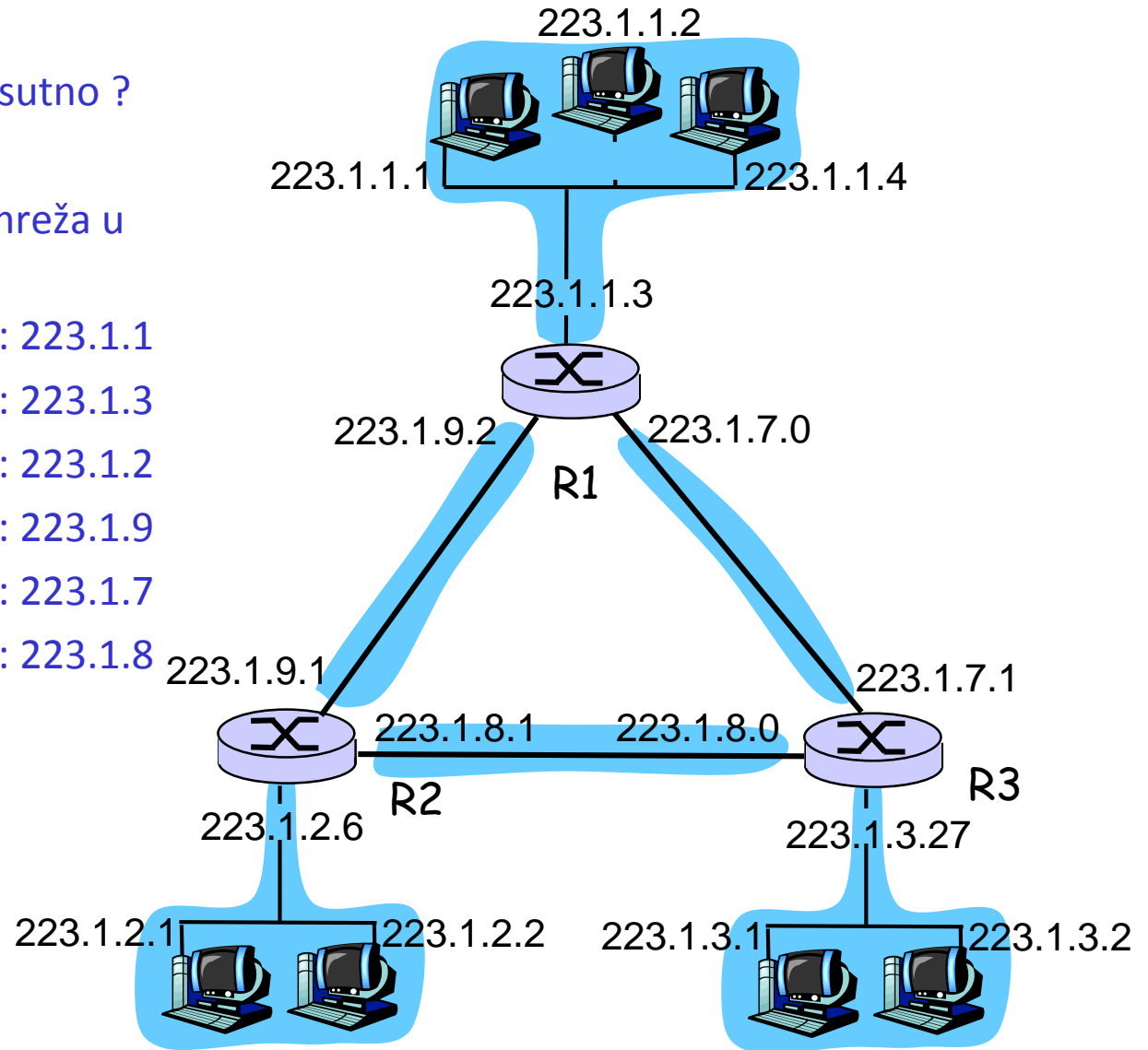
**Netid** - identifikuje mrežu

**Hostid** - identifikuje host u mreži

**Blok** - skup adresa sa istim netid

## Zadatak 2.

- 1) Koliko podmreža je prisutno ?
  - a) 6
- 2) Navedite adrese podmreža u decimalnom obliku
  - a) Mrežna adresa -1: 223.1.1
  - b) Mrežna adresa -2: 223.1.3
  - c) Mrežna adresa -3: 223.1.2
  - d) Mrežna adresa -4: 223.1.9
  - e) Mrežna adresa -5: 223.1.7
  - f) Mrežna adresa -6: 223.1.8



## Zadatak 3.

- ❑ Broj adresa u bloku mora biti stepen dvojke (2, 4, 8, 16, 32,...).
- ❑ Ako blok ima 4 adrese, prva adresa odnosno krajnji desni bajt mora biti deljiv sa 4.
- ❑ Ako blok ima 16 adrese, prva adresa odnosno krajnji desni bajt mora biti deljiv sa 16.
- ❑ Ako blok ima 256 ( $2^8$ ) ili manje adresa, dovoljno je proveriti samo krajnji desni bajt, koji mora biti deljiv sa 256, te je logično da krajnji bajt mora biti 0, jer su samo brojevi od 0 do 255 dostupni u datom sistemu.
- ❑ Ako blok ima 65536 ( $2^{16}$ ) adresa krajnji bajtovi moraju biti deljiv sa 256 te je logično da krajnji bajt mora biti 0, jer su samo brojevi od 0 do 255 dostupni u datom sistemu, a drugi po redu bajt sa 4, jer je  $1024=4 \times 256$ .



□ Koja od sledećih adresa može biti početna adresa bloka koji sadrži 16 adresa.

○ a. 200.16.38.32

b. 205.16.42.31

c. 16.16.35.96

d. 121.35.24.41

○ Prva adresa tj. krajnji desni bajt u bloku mora biti deljiva bez ostatka brojem adresa u bloku u ovom slučaju sa 16.

○ adrese 188.11.25.32 i 223.1.1.96 su početne adrese zato što je 32 i 96 deljivo sa 16.

## Zadatak 4.

- Odrediti broj adresa u bloku ako je 200.23.18.0/20 jedna od adresa iz bloka.
  - a.b.c.d/x - dužina sufiksa može se predstaviti na sledeći način:  $32 - x = \text{dužina sufiksa}$
  - Dužina prefiksa je  $x = 20$  te sledi  $32 - x = 32 - 20 = 12$  (u pitanju je broj adresa koji se koristi za host uređaje, dati sufiks je promenljiv u funkciji generisanja međusovno različitih adresa) tj.  $2^{12} = 4096$ , broj potencijalnih adresa

## Zadatak 5.

- Odrediti prvu adresu u bloku ako je 200.23.20.0/20 jedna od adresa iz bloka
  - a.b.c.d/x - dužina sufiksa može se predstaviti na sledeći način:  $32 - x = \text{dužina sufiksa}$ . Dužina prefiksa je  $x = 20$  te sledi  $32 - x = 32 - 20 = 12$  je dužina sufiksa
  - Dužina prefiksa je 20, prvih 20 bita se ne menjaju stalni su, dok preostale bitove tj. 12 bitova "menjamo" na 0.
  - Adresa u binarnom obliku je: 11001000 00010111  
00010000 00000000
  - prvih 20 bita ostaju stalni te sledi: 11001000  
00010111 000010000 00000000
  - Prva adresa je: 200.23.16.0/20

## Zadatak 6.

□ Odrediti poslednju adresu u bloku ako je 200.23.15.0/20 jedna od adresa iz bloka.

○ Maska ima 20 jedinica i 12 nula. Komplement ("suprotnost") maske je 20 nula i 12 jedinica

• Maska odnosno "mrežni deo" IP adrese je:

– 11111111 11111111 11110000 00000000 - (20 jedinica i 12 nula)

– 00000000 00000000 00001111 11111111 - (20 nula i 12 jedinica) u decimalnom obliku je: 0.0.15.255

○ poslednja adresa u datom bloku dobija se zbirom početne adrese i komplementa maske:

$$\begin{array}{r} 200.23.16.0 \\ 0.0.15.255 \\ \hline 200.23.31.255 \end{array}$$

○ Poslednja adresa u bloku je: 200.23.31.255/20

- 1.1 Uvod
- 1.2. Mrežni sloj
- 1.3 Virtuelna kola i datagram mreže
- 1.4 Šta je u unutrašnjosti rutera
- 1.5 IP: Internet protokol
  - format datagrama
  - IPv4 adresiranje
  - IPv6
- 1.6 zadaci

- 2.1 Softversko inženjerstvo
- 2.2 metode i metodologije (UML)
- 2.3 Zadaci

## □ 2.1. Softversko inženjerstvo

- Termin “Softversko inženjerstvo” prvi se put pojavljuje na NATO konferenciji 1968. godine. Dati pojam podrazumeva skup metoda odnosno tehnika i alata za projektovanje softvera.

□ 2.2. U objektno orijentisanom pristupu razdvajaju se metode i metodologije

- ZA MODELE SE DEFINIŠU STANDARDI - NAJPOZNATIJI STANDARD odnosno grafički jezik je: UML (Unified Modeling Language) koji omogućava prikazivanje konceptijskih i konkretnih aspekata.

- 1.1 Uvod
- 1.2. Mrežni sloj
- 1.3 Virtuelna kola i datagram mreže
- 1.4 Šta je u unutrašnjosti rutera
- 1.5 IP: Internet protokol
  - format datagrama
  - IPv4 adresiranje
  - IPv6
- 1.6 zadaci

- 2.1 Softversko inženjerstvo
- 2.2 metode i metodologije (UML)
- 2.3 Zadaci

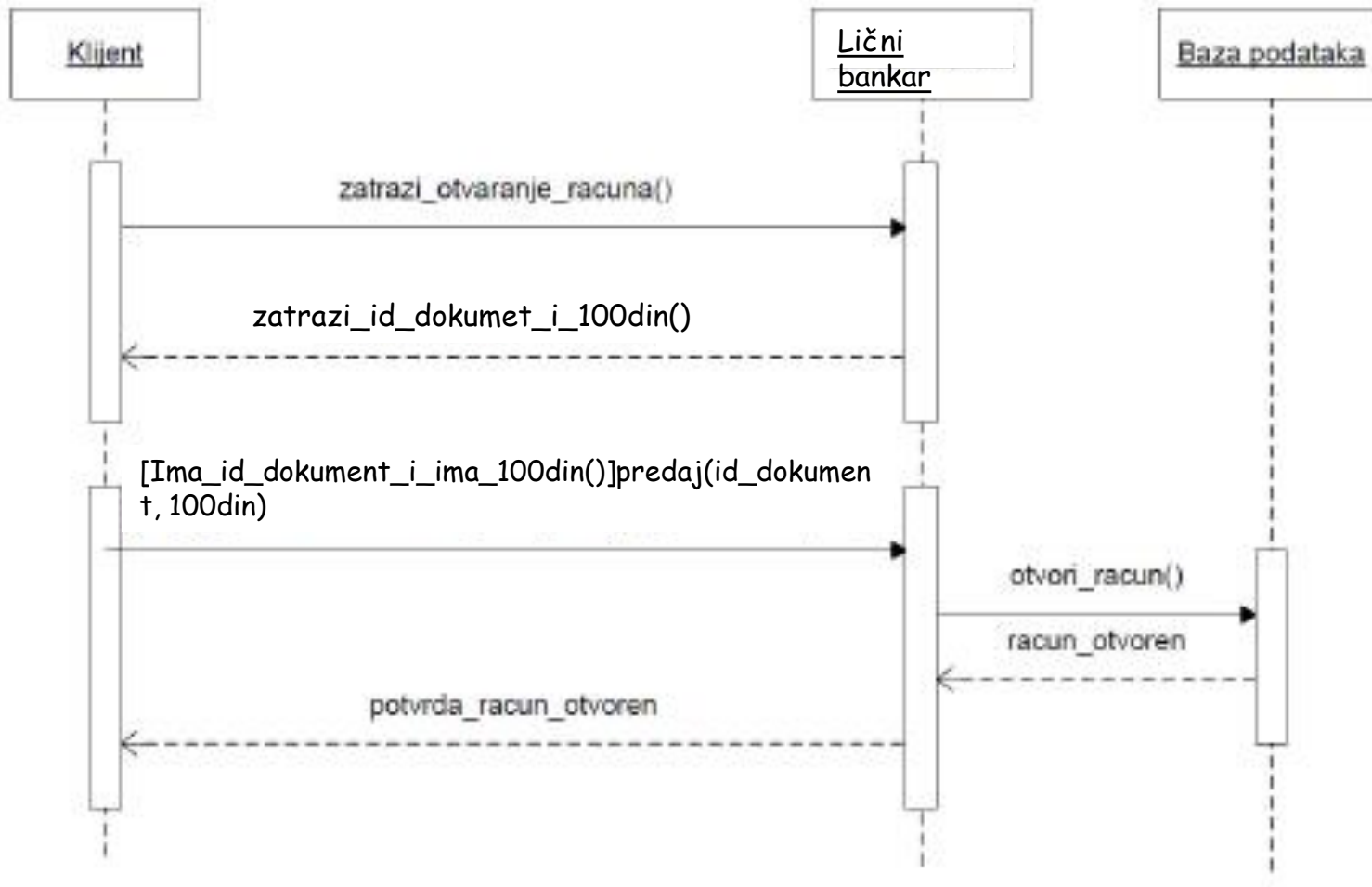


## Zadatak 1.

- U ovom zadatku modelujemo otvaranje računa u banci sekvencijalnim dijagramom. Klijent traži otvaranje novog računa. Ličnom bankaru je neophodno lična karta (id - dokument) i 100 dinara. Ukoliko potencijalni klijent nema ličnu kartu ili 100 dinara, interakcija se stopira. Ukoliko poseduje ličnu kartu i 100 dinara predaje dati id-dokument i novac. Zatim, bankar otvara račun u bazi podataka. Nakon otvaranja računa, službenik banke tj. bankara potvrđuje klijentu i interakcija je završena.

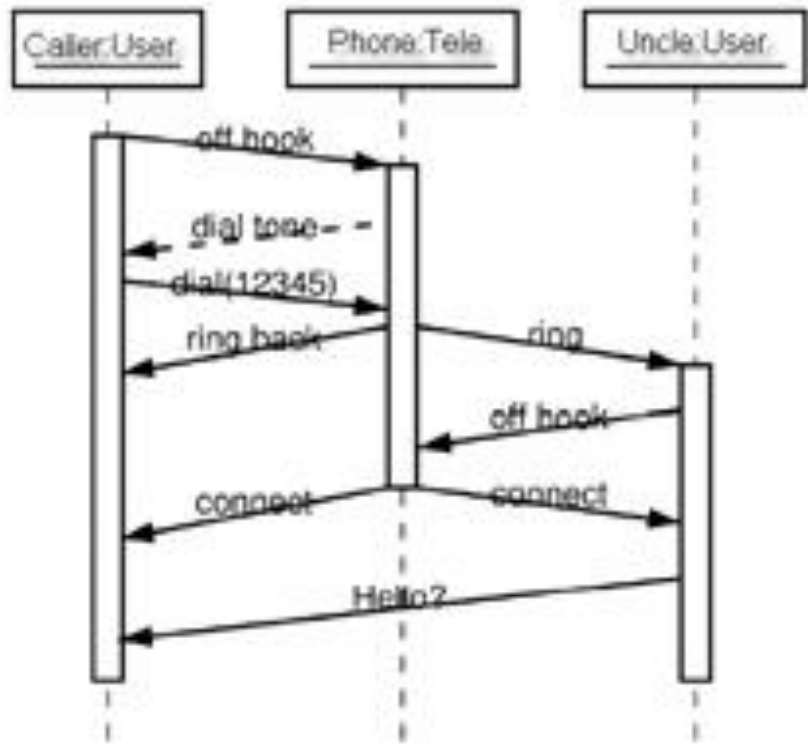
Ugalste zagrade ([ ] - na primer:

[Ima\_id\_dokument\_i\_ima\_100din()]predaj(id\_dokument, 100din) definišu mogućnost odnosno opciju. U ovom slučaju potencijalni klijent može, a ne mora imati novac odnosno id - dokument.



## Zadatak 2.

- U ovom zadatku modelujemo telefonski poziv između dva korisnika. Korišćenjem sekvencijalnog dijagrama.



## □ Literatura

- Umrežavanje računara od vrha ka dnu sa Internetom u fokusu, 2005, James F. Kurose, Keith W. Ross
- Savremene komunikacione tehnologije i mreže, 2004, William A. Shay
- Software Engineering, 2001, Roger S. Pressman
- UML vodič za korisnike, 2000, Grady Booch, James Rumbauch, Ivar Jacobson

- ❑ Termin polaganja II kolokvijuma -19.6. (termin predavanja, grupe naknadno na oglasnoj tabli)
- ❑ Struktura II kolokvijuma: 6 teorijskih pitanja na zaokruživanje ili dopunu i 2 zadatka
- ❑ Na kolokvijum poneti id karticu ili indeks i digitron
- ❑ Vreme -20min.

# POSLOVNA INFORMATIKA

## 1- GODINA

ZADACI ZA PRVI KOLOKVIJUM-1-VEŽBE

# 1. Brojčani sistem

- **1.1. Decimalni sistem**

- U okruženju koristimo decimalni sistem koji se bazira na decimalnim ciframa  $\{0, 1, 3, 4, 5, 6, 7, 8, 9\}$  - datih cifara ima 10.

- *Primer 1.*

- Broj 83 možemo napisati kao zbir osam desetica i tri jedinice.

- $83 = 8 \times 10 + 3$  ili  $83 = 8 \times 10^1 + 3 \times 10^0$ .



– *Primer 2.*

- Broj 4728 - četiri hiljade, sedam stotina, dve desetice i osam jedinica:

$$- 4728 = 4 \times 1000 + 7 \times 100 + 2 \times 10 + 8 \text{ ili } 4728 = 4 \times 10^3 + 7 \times 10^2 + 2 \times 10^1 + 8 \times 10^0 \quad .$$

Na osnovu prethodnih primera možemo reći za decimalni sistem da ima bazu (osnovu) 10. To znači da se svaka cifra množi brojem 10 podignutim na stepen koji odgovara položaju date cifre.

– *Primer 3.*

- Isto pravilo važi i za decimalne razlomke. U ovom slučaju koristise cifra 10 na negativni stepen.
- Decimalni razlomak:

$$- 0.256 = (2 \times 10^{-1}) + (5 \times 10^{-2}) + (6 \times 10^{-3})$$

- U opštem slučaju, za decimalno predstavljanje broja  $X = \{ \dots d_2d_1d_0, d_{-1}d_{-2}d_{-3} \dots \}$ , možemo koristiti slededi izraz:

$$X = \sum(d_i \times 10^i)$$

- **1.2. Binarni sistem**

- U decimalnom sistemu koristi se deset različitih cifara sa za predstavljanje brojeva sa osnovom 10. U binarnom sistemu imamo samo dve cifre, 1 i 0 - zaključak je da su brojevi u binarnom sistemu predstavljeni osnovom 2.
- U sledećim primerima koristidemo indeks kako bismo označili osnovu datog broja. Na primer, ovi brojevi su decimalni brojevi:  $82_{10}$ ,  $83_{10}$ ,  $175_{10}$ ,  $4728_{10}$ .
- Cifre 1 i 0 imaju isti binarni i decimalni oblik:  
 $0_2 = 0_{10}$ ,  $1_0 = 1_{10}$

– *Primer 4.*

- $10_2 = 1 \times 2^1 + 0 \times 2^0 = 2_{10}$
- $11_2 = 1 \times 2^1 + 1 \times 2^0 = 3_{10}$
- $100_2 = 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 = 4_{10}$

– *Primer 5.*

- U binarnom sistemu razlomke možemo predstaviti na sledeći način:

$$- 1001,101 = 2^3 + 2^0 + 2^{-1} + 2^{-3} = 9.625_{10}$$

U opštem slučaju, binarno predstavljanje broja  $Y = \{ \dots b_2 b_1 b_0, b_{-1} b_{-2} b_{-3} \dots \}$  poprima sledeći oblik:

$$Y = \sum b_i \times 2^i$$

## • 1.2.1. Pretvaranje binarnih u decimalne brojeve

– Celebrojni delovi

$$b_{m-1}b_{m-2}\dots b_2b_1b_0 \quad b_i = 0 \text{ ili } 1$$

$$(b_{m-1} \times 2^{m-1}) + (b_{m-2} \times 2^{m-2}) + \dots + (b_1 \times 2^1) + b_0$$

– Ako podelimo  $N$  sa 2, u decimalnom sistemu dobijamo količnik  $N_1$  i ostatak  $R_0$ :

$$N = 2 \times N_1 + R_0, \quad R_0 = 0 \text{ ili } 1$$

– Zatim delimo sa 2 i dobijamo količnik i novi ostatak:

$$N_1 = 2 \times N_2 + R_1, \quad R_1 = 0 \text{ ili } 1$$

– Možemo napisati:

$$N = 2(2N_2 + R_1) + R_0 = (N_2 \times 2^2) + (R_1 \times 2^1) + R_0$$

– Ukoliko nastavimo i dalje sa datim izračunavanjem, dobijamo:

$$N_2 = 2N_3 + R_2$$

- Daljim izračunavanjem dobijamo uopšteniji oblik:  $N = (N_3 \times 2^3) + (R_2 \times 2^2) + (R_1 \times 2^1) + R_0$ ,
- Pošto N ima najveću vrednost u datom nizu, možemo napisati sledede:  $N > N_1 > N_2 \dots > N_{m-1}$ ; pritom poslednji član niza (kolicnik) ima vrednost  $N_{m-1} = 1$  (izuzimajući cele brojeve 0 i 1 koji imaju isti oblik i u binarnom sistemu), i ostatak  $R_{m-2}$  koji ima vrednost 0 ili 1, te se može napisati:

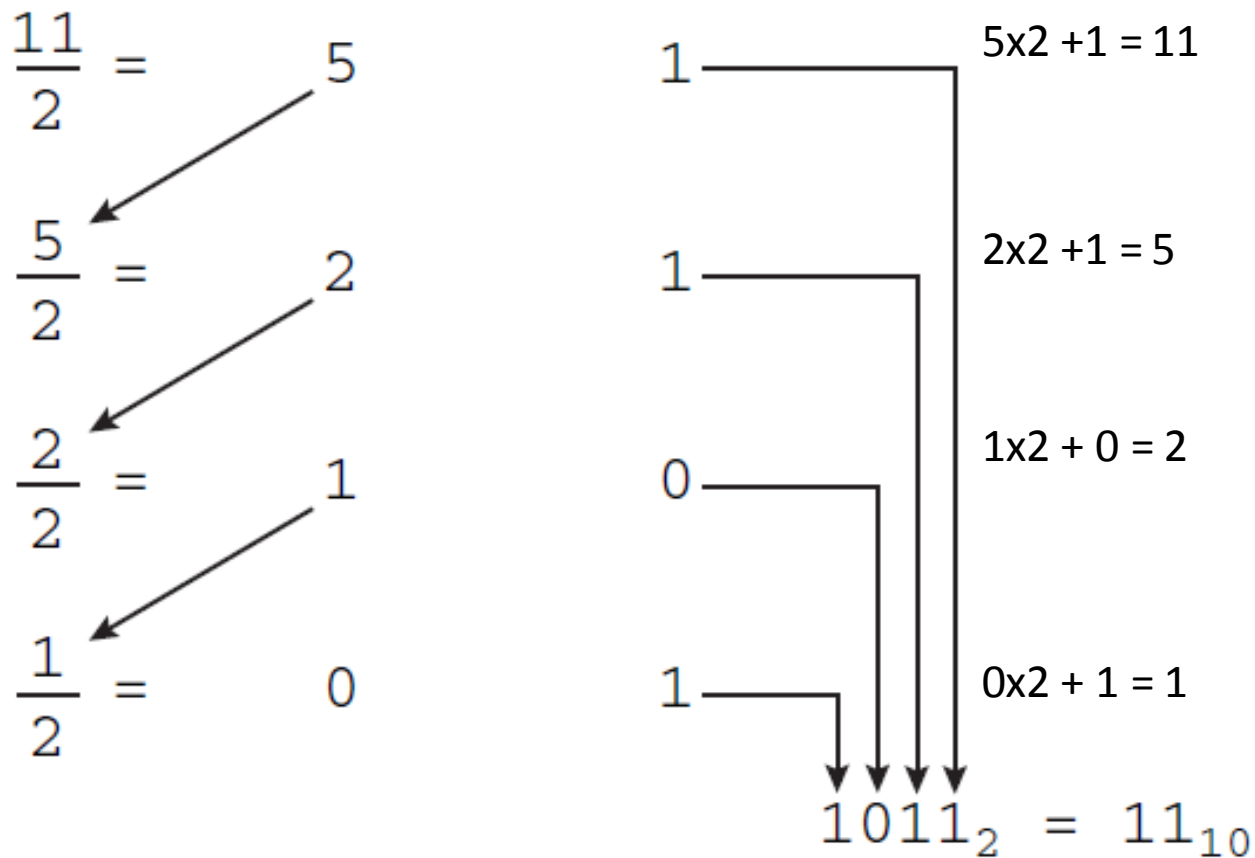
$$N = (1 \times 2^{m-1}) + (R_{m-2} \times 2^{m-2}) + \dots + (R_2 \times 2^2) + (R_1 \times 2^1) + R_0$$

Primer 6.

Količnik

Ostatak

**Quotient    Remainder**

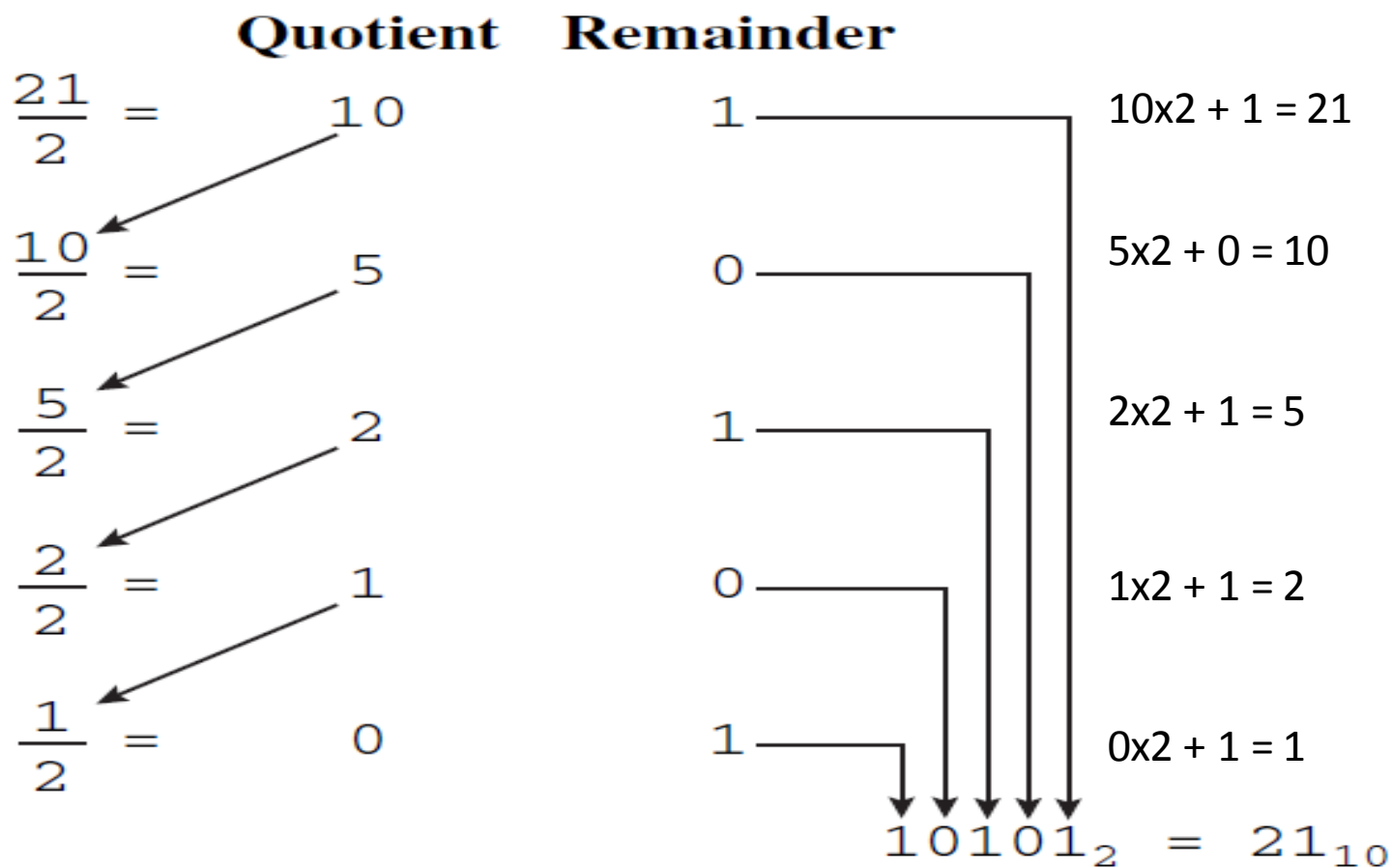


(a) 11<sub>10</sub>

# Primer 7.

Količnik

Ostatak



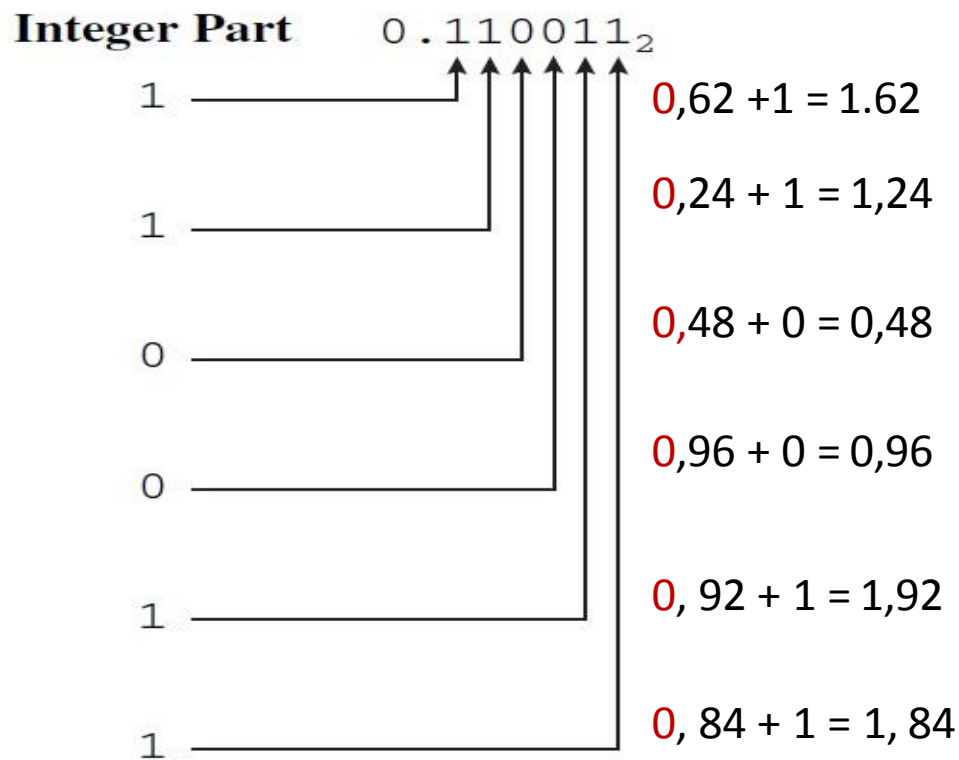
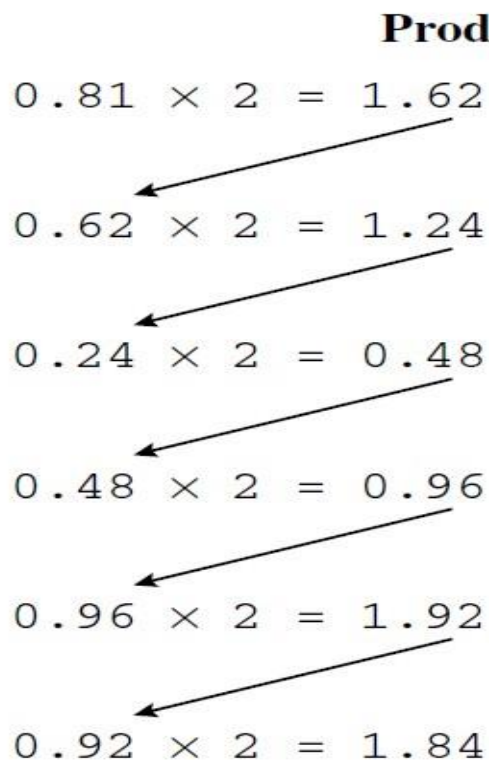
(b)  $21_{10}$



# Primer 8.

Proizvod

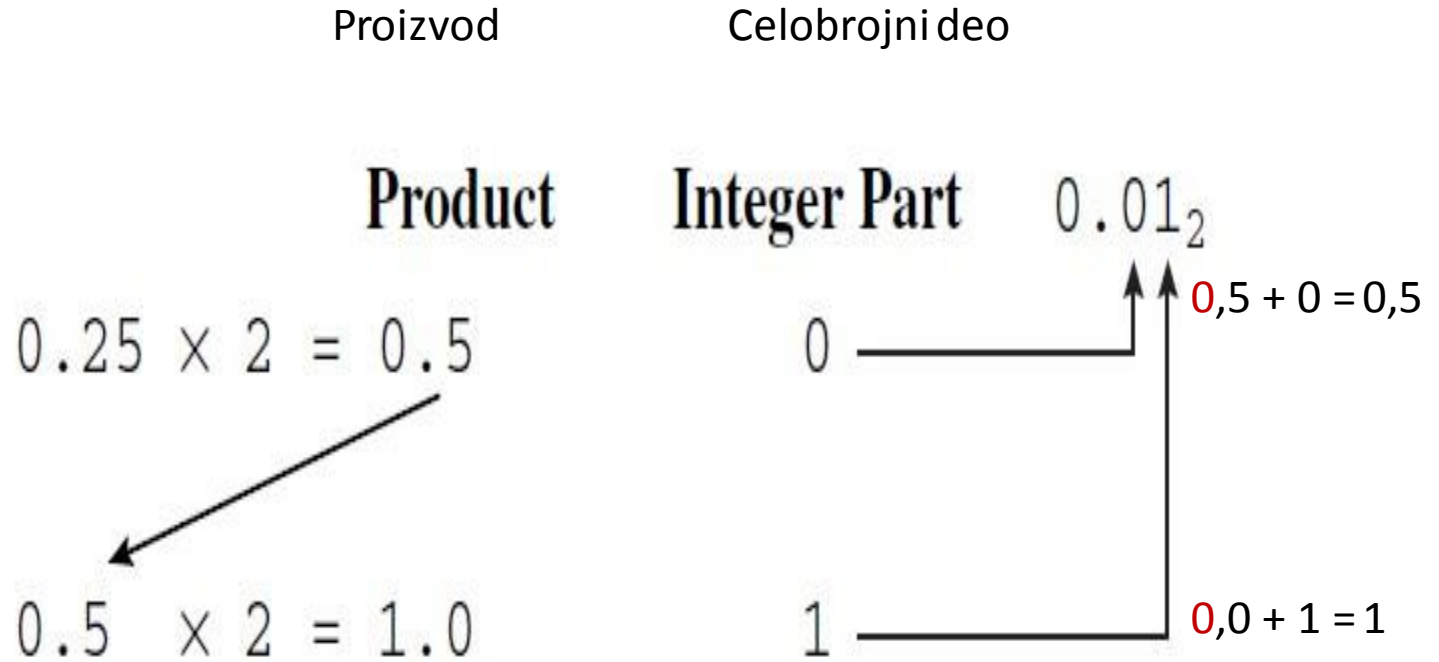
Celobrojni deo



(a)  $0.81_{10} = 0.11011_2$  (approximately)

Približna vrednost, te sledi da je neophodno prekinuti izračunavanje-a u određenom koraku.

# Primer 9.



$$(B) 0.25_{10} = 0.01_2 \text{ (exactly)}$$

Tačna vrednost, te sledi da se izračunavanje odvijalo do “kraja”.

- **1.2.2. Binarno sabiranje**

- Binarno sabiranje obavlja se tako što se prenos na sledeće značajno mesto obavlja nakon postignutog zbira (1+1) - “pamti se jedinica”. Cifre obojene u crveno su napisane kao deo rezultata. U pitanju su cifre koje se prenose u sledeću kolonu (videti primer 10).

## Primer 10.

- $0 + 0 = 0$
- $1 + 0 = 1$
- $0 + 1 = 1$
- $1 + 1 = 10$
- $1 + 1 + 1 = 11$

Bitovi koji se prenose u  
slededu kolonu

	11	1 <-----bit ----->	11
1001101	1001001		1000111
+0010010	+0011001		+0010110
-----	-----		-----
1011111	1100010		1011101

## Primer 11.

Sabrati brojeve  $11011_2$   
i  $1011_2$ .

Prenos	1	1		1	1	
		1	1	0	1	1
+			1	0	1	1
	1	0	0	1	1	0

		1	+	1	=	0 i 1 dalje
1	+	1	+	1	=	1 i 1 dalje
1	+	0	+	0	=	1
		1	+	1	=	0 i 1 dalje
1	+	1			=	0 i 1 dalje
1					=	1

– *Primer 12.*

- Sabrati date brojeve.

$$\begin{array}{r} 1001011_2 = 75_{10} \\ + 1011110_2 = 94_{10} \\ \hline 10101001_2 = 169_{10} \end{array}$$

$$\begin{array}{r} 1111001_2 = 121_{10} \\ + 110100_2 = 52_{10} \\ \hline 10101101_2 = 173_{10} \end{array}$$

### Primer 13.

- Izračunati zbirove brojeva u binarnom sistemu.
- Postavka-e:

$$\begin{array}{r} 1011101 \\ + 1000000 \\ \hline \end{array}$$

$$\begin{array}{r} 10011 \\ + 1111101 \\ \hline \end{array}$$

- Rešenja-e:

$$\begin{array}{r} 10011001 \\ + 100111 \\ \hline \end{array}$$

$$\begin{array}{r} 11000011 \\ + 101111 \\ \hline \end{array}$$

$$\begin{array}{r} 1001100 \\ + 1100101 \\ \hline \end{array}$$

$$\begin{array}{r} 1011101 \\ + 1000000 \\ \hline \end{array}$$

$$\begin{array}{r} 10011 \\ + 1111101 \\ \hline \end{array}$$

**10011101**

**10010000**

$$\begin{array}{r} 10011001 \\ + 100111 \\ \hline \end{array}$$

$$\begin{array}{r} 11000011 \\ + 101111 \\ \hline \end{array}$$

$$\begin{array}{r} 1001100 \\ + 1100101 \\ \hline \end{array}$$

**11000000**

**11110010**

**10110001**

- **1.3. Heksadecimalni sistem**

- U matematici i informatici se česti koristi heksadecimalni brojevni sistem – brojni sistem sa osnovom 16 (sistem sa 16 različitih cifara). Obično se koristi 10 arapskih cifara i dodaju se slova A-F (a-f)
- **Heksadecimalni** sistem je IBM uveo u upotrebu 1963.godine.
- Npr., broj 10101010 u binarnom sistemu, koji se piše 170 u dekadnom sistemu, u heksadecimalnom je AA.



- **1.3. Heksadecimalni sistem**

- Heksadecimalni sistem je pogodno koristiti u računarskim sistemima iz razloga jednostavnijeg pretvaranja binarnog u heksadec.sistem. Tako svaka 4 bita mogu da se napišu kao jedna cifra heksadecimalnog sistema, što znači da se 1 bajt može napisati kao dve cifre u heksadecimalnom sistemu.

Tabela koja definiše vezu između dekadnog i binarnog sistema

0000 = 0	1000 = 8
0001 = 1	1001 = 9
0010 = 2	1010 = A
0011 = 3	1011 = B
0100 = 4	1100 = C
0101 = 5	1101 = D
0110 = 6	1110 = E
0111 = 7	1111 = F

– *Primer 14.*

$$2C_{16} = 2_{16} \times 16^1 + C_{16} \times 16^0 = 2_{10} \times 16^1 + 12_{10} \times 16^0 = 44$$

– *Primer 15.*

$$1101 \quad 1110 \quad 0001 = DE1_{16}$$

D

E

1

– *Primer 16.*

$$A2F_{16} = A \times 16^2 + 2 \times 16^1 + F \times 16^0 = 10 \times 256 + 2 \times 16 + 15 \times 1 = 2607$$

$$A2F_{16} = 2607_{10}$$

## 2. Digitalna logička kola

- Rad digitalnih računara zasniva se na skladištenju i obradi binarnih podataka. Elementi za skladištenje podataka, odnosno kola nalaze se u jednom od dva stabilna stanja.
- Upravljački signali definišu funkciju elemenata koji se nalaze na “nižem” nivou, a samim tim i funkcije računarskog sistema kao celine.

- Elementi za skladištenje podataka, tj. kola mogu se implementirati u obliku digitalnih logičkih kola.
- Digitalna logička kola javljaju se u obliku kombinatornih i sekvencijalnih kola.
- Na osnovu navedenog, može se prepoznati važnost poznavanja osnovnih komponenata kompjuterskog sistema.

P	Q	NE P	P I Q	P I L I Q	P X O R Q	P N I Q	P N I L I Q
0	0	1	0	0	0	1	1
0	1	1	0	1	1	1	0
1	0	0	0	1	1	1	0
1	1	0	1	1	0	0	0

$$A \text{ I } B = A \cdot B$$

$$A \text{ I L I } B = A + B$$

$$\text{NE } A = \overline{A}$$

$$A \text{ N I } B = \text{NE}(A \text{ I } B) = \overline{A \cdot B}$$

$$A \text{ N I L I } B = \text{NE}(A \text{ I L I } B) = \overline{A + B}$$

$$A + B \cdot C = A + (B \cdot C) = A + BC$$

## Osnovni postulati

$$A \cdot B = B \cdot A$$

$$A + B = B + A$$

zakoni komutacije

$$A \cdot (B + C) = (A \cdot B) + (A \cdot C)$$

$$A + (B \cdot C) = (A + B) \cdot (A + C)$$

zakoni distributivnosti

$$1 \cdot A = A$$

$$0 + A = A$$

jednakost elemenata

$$A \cdot \bar{A} = 0$$

$$A + \bar{A} = 1$$

inverznost elemenata

## Ostali identiteti

$$0 \cdot A = 0$$

$$1 + A = 1$$

$$A \cdot A = A$$

$$A + A = A$$

$$A \cdot (B \cdot C) = (A \cdot B) \cdot C$$

$$A + (B + C) = (A + B) + C$$

zakon asocijacije

$$\overline{A \cdot B} = \bar{A} + \bar{B}$$

$$\overline{A + B} = \bar{A} \cdot \bar{B}$$

De Morganove teoreme

$$A \text{ NI LI } B = \bar{A} \text{ I } \bar{B}$$

$$A \text{ NI } \bar{B} = \bar{A} \text{ ILI } B$$

$$A + B = \overline{\bar{A} \cdot \bar{B}}$$

$$A \text{ ILI } B = \text{NE}((\text{NE } A) \text{ I } (\text{NE } B))$$



- Periodični signal: signal koji se ponavlja u jednakim vremenskim intervalima
- Ciklus: “deo” procesa koji se periodično ponavlja
- Period (p): vreme trajanja jednog ciklusa
- Frequency (f): 1/period (1 Hertz (Hz) = 1 cycle per second (cps) tj. 1ciklus/1sekund )
- $f = 1/p$  sledi  $p = 1/f$
- Bandwidth: (frequency opseg)
- bit rate: # bitova u sekundi (bps or Kbps or Mbps or Gbps)

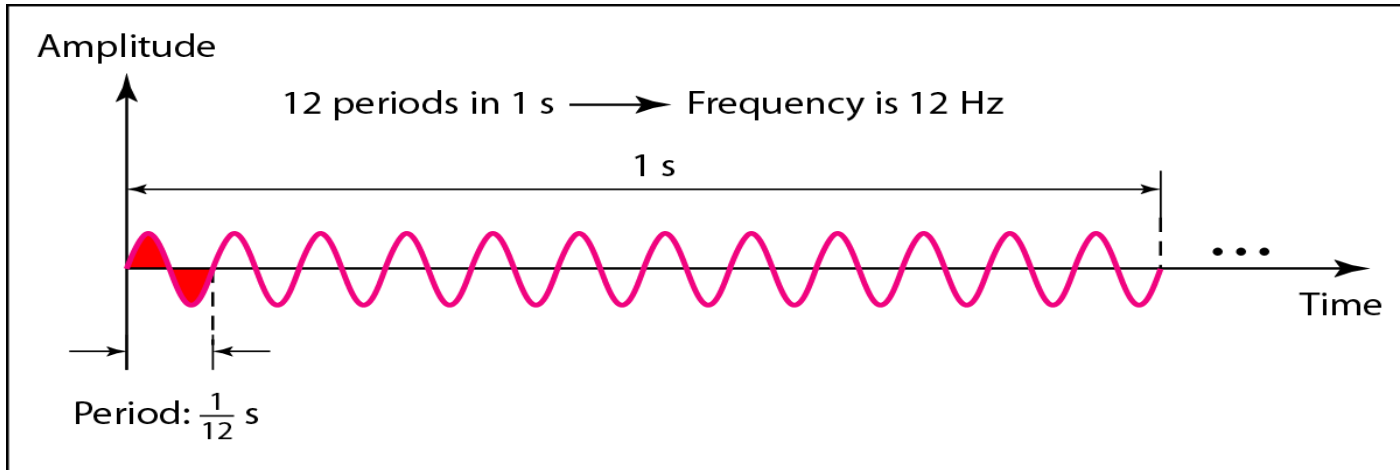
- *Primer 17.*

Nacrtati signale proizvoljne faze i amplitude sa frekvencijama od 12Hz i 6Hz (Primer-i dva signala mogu imati iste amplitude i faze, radi preglednosti, ali različite frekvencije).

a) prvi primer signala se karakteriše sa 12 perioda, odnosno 12 ciklusa u 1s, te sledi da prvi signal poseduje frekvenciju od 12 Hz.

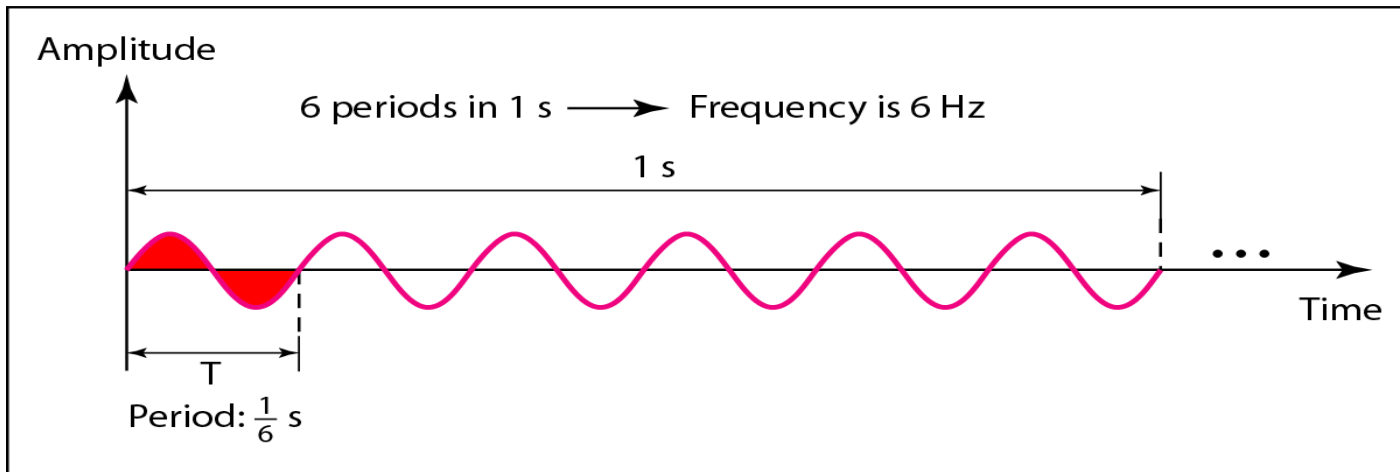
b) Drugi primer signala se karakteriše sa 6 perioda, odnosno 6 ciklusa u 1s, te sledi da drugi signal poseduje frekvenciju od 6 Hz.

1.



a. A signal with a frequency of 12 Hz

2.



b. A signal with a frequency of 6 Hz

# I primeri...



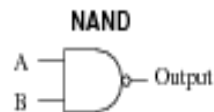
A	B	Output
0	0	
0	1	
1	0	
1	1	



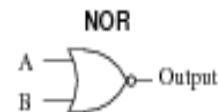
A	B	Output
0	0	
0	1	
1	0	
1	1	



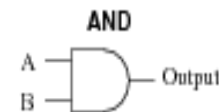
A	B	Output
0	0	
0	1	
1	0	
1	1	



A	B	Output
0	0	1
0	1	1
1	0	1
1	1	0



A	B	Output
0	0	1
0	1	0
1	0	0
1	1	0



A	B	Output
0	0	0
0	1	0
1	0	0
1	1	1



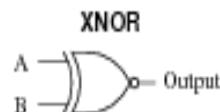
A	B	Output
0	0	
0	1	
1	0	
1	1	



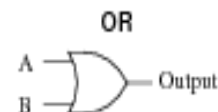
A	B	Output
0	0	
0	1	
1	0	
1	1	



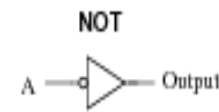
A	Output
0	
1	



A	B	Output
0	0	1
0	1	0
1	0	0
1	1	1



A	B	Output
0	0	0
0	1	1
1	0	1
1	1	1



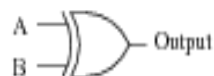
A	Output
0	1
1	0



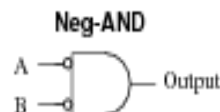
A	B	Output
0	0	
0	1	
1	0	
1	1	



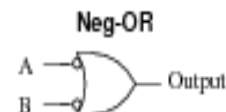
A	B	Output
0	0	
0	1	
1	0	
1	1	



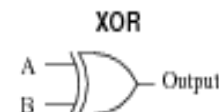
A	B	Output
0	0	
0	1	
1	0	
1	1	



A	B	Output
0	0	1
0	1	0
1	0	0
1	1	0



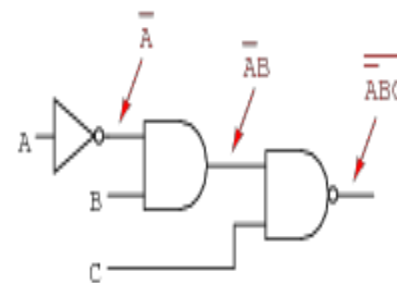
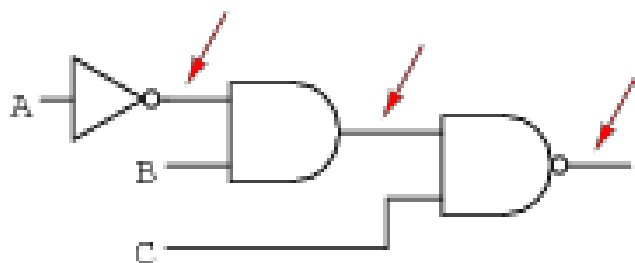
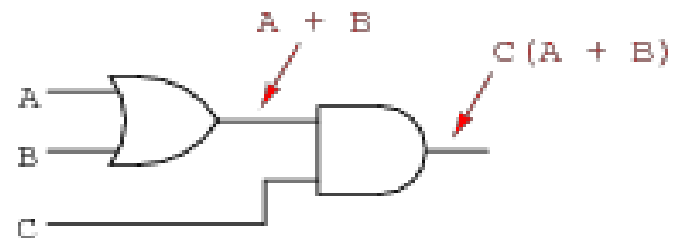
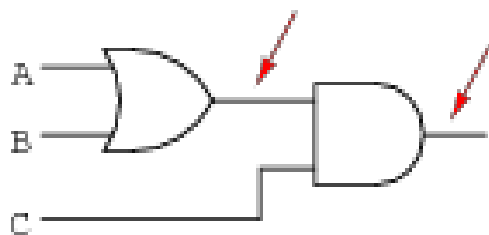
A	B	Output
0	0	1
0	1	1
1	0	1
1	1	0



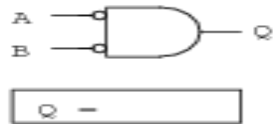
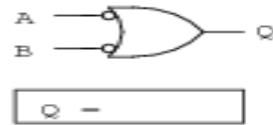
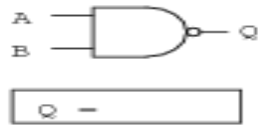
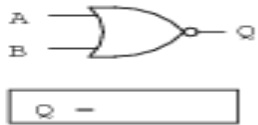
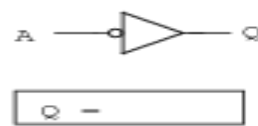
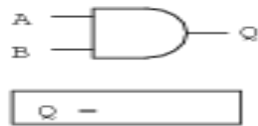
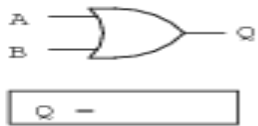
A	B	Output
0	0	0
0	1	1
1	0	1
1	1	0

• Postavka:

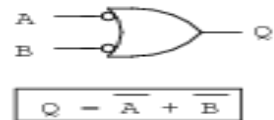
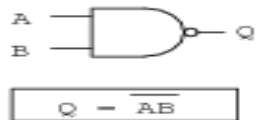
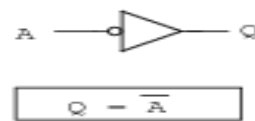
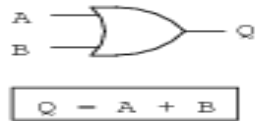
Rešenje:



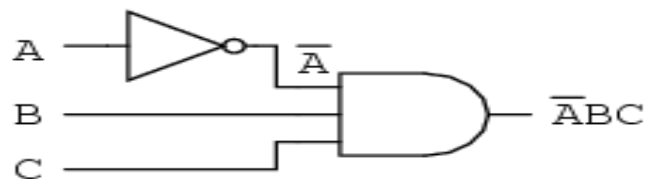
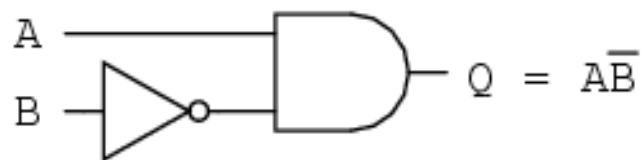
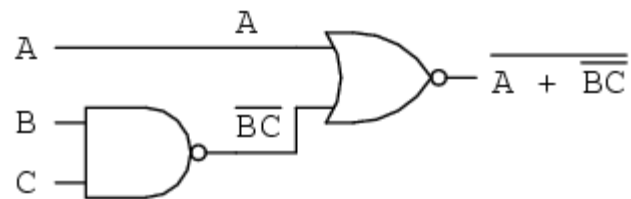
# Postavka:



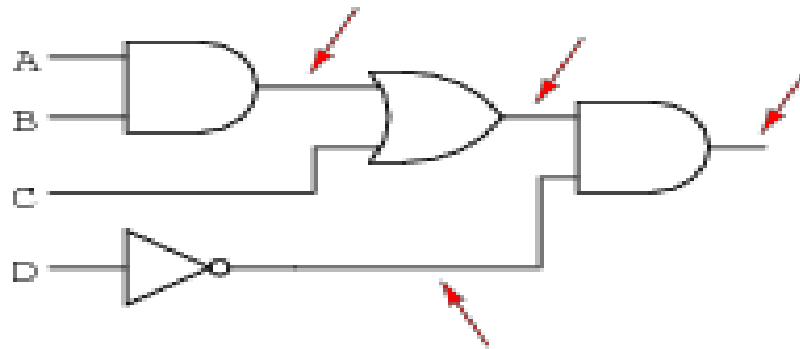
# Rešenje:



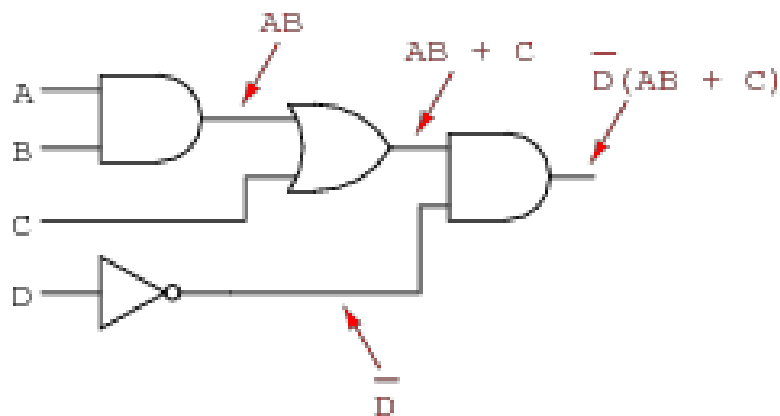
● ...



- Postavka:

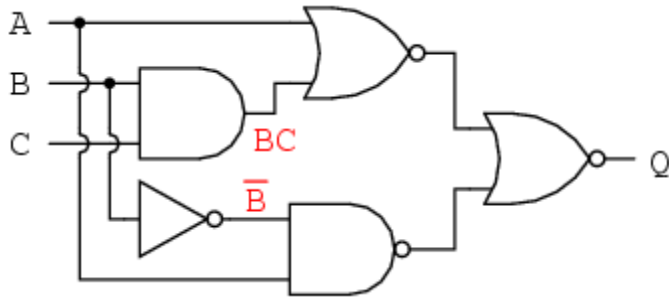


- Rešenje:

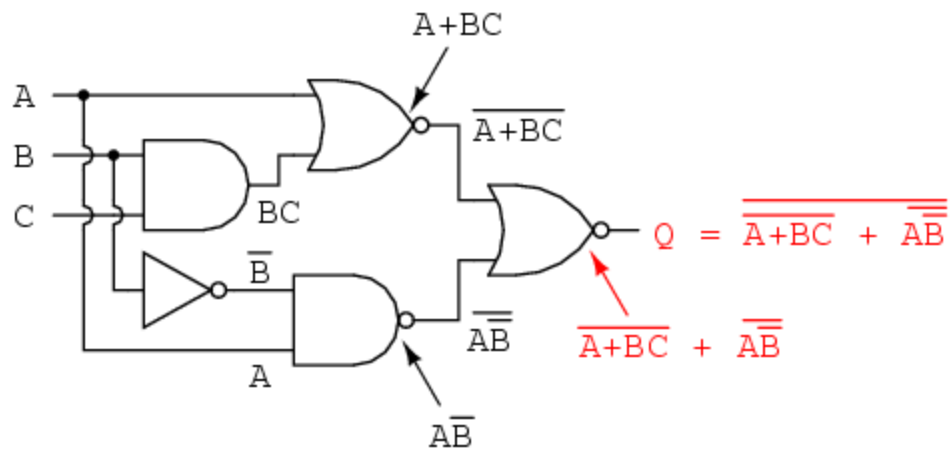




- Postavka:



- Rešenje:



- Termin polaganja I kolokvijuma – (termin predavanja, grupe naknadno na oglasnoj tabli)
- Struktura I kolokvijuma: 6 teorijskih pitanja na zaokruživanje ili dopunu i 2 zadatka