

RUKOVANJE INFORMACIJAMA

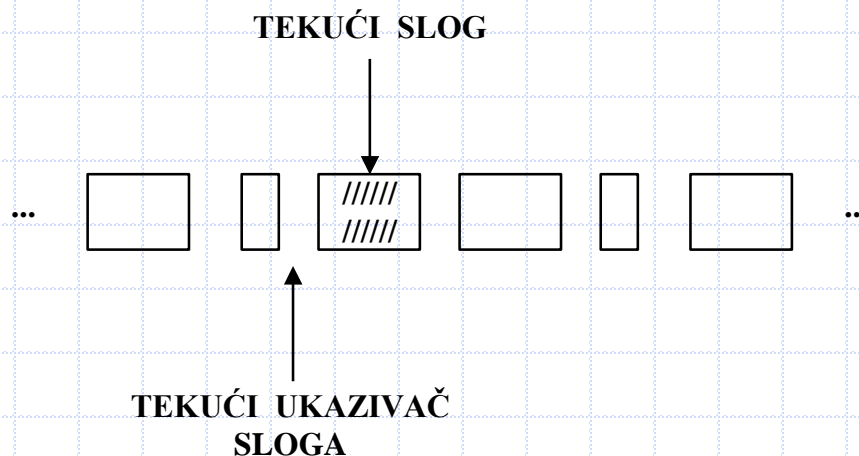
II DEO

- ❖ Metodi pristupa datoteci
- ❖ Baferisanje
- ❖ Hijerarhijski model FS-a

Metodi pristupa datoteci

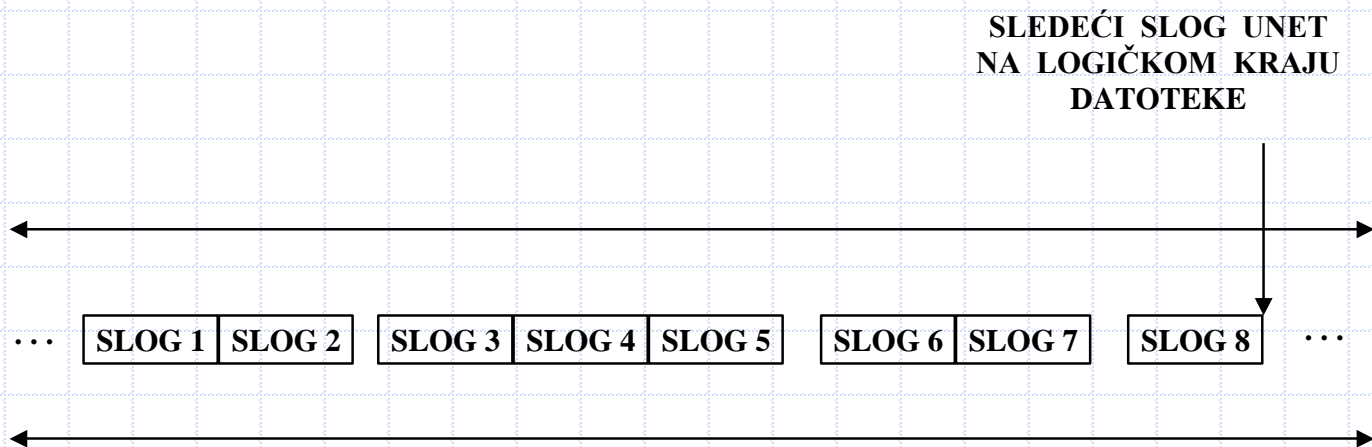
- ◆ Pristup slogovima na 2 načina:
 - Sekvencijalno: po redosledu po kom su u datoteci
 - Slučajno: po ključu, na osnovu koga metod pristupa datoteci određuje željeni slog
- ◆ Tekući slog je zadnji uspešno očitani slog
 - Tekući ukazivač sloga je ID tekućeg sloga u datoteci

SLOG UNUTAR DATOTEKE



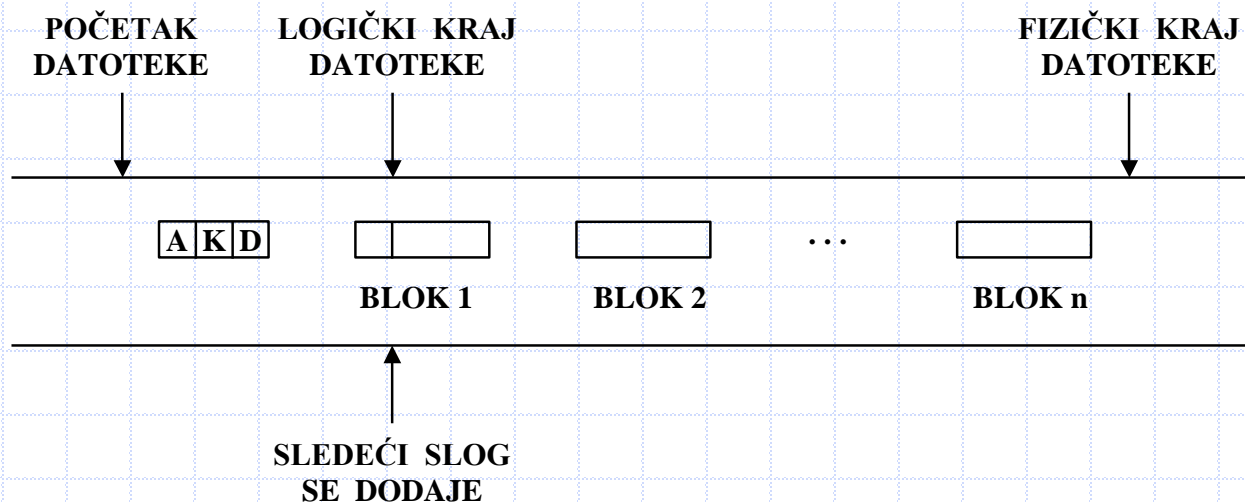
Sekvencijalna (serijska) datoteka (1/2)

- ◆ Def: je datoteka čiji je slog podataka smešten neposredno iza prethodnog sloga
 - Ne zavisi od medijuma na koji je smešten
 - Slogovi se smeštaju prema momentu nailaska, dodavanjem slogova, sve do logičkog kraja datoteke
 - Dat. je formirana od korisničkih slogova koji imaju svoj prethodni i svoj naredni slog



Sekvencijalna (serijska) datoteka (2/2)

- Novi slog može se dodati samo na logičkom kraju dat., a program može očitati samo sledeći slog iz dat.
- Slogovi mogu biti fiksne i promenjive dužine
- U drugom sl. slogu prethodi ukazivač njegove dužine
- Može se fizički realizovati na jedinici spoljnje mem. ili sekvencijalno ili pomoću ukazivača (na naredni slog)

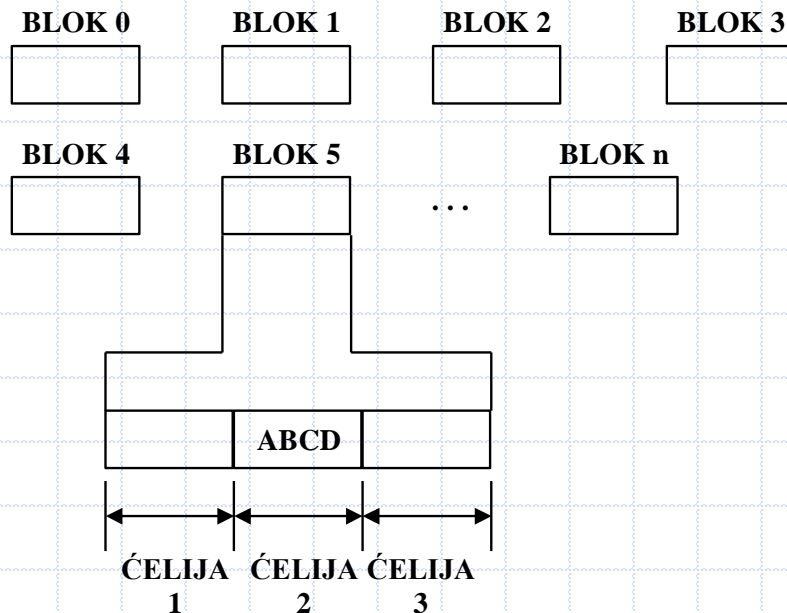


Relativna datoteka (1/2)

- ◆ Def: je datoteka u kojoj se pojedinačnim slogovima pristupa slučajno (direktno)
 - Slogu se može pristupiti na nivou jedne UI operacije, svaki slog može da se ažurira ili odstrani, uključujući i promene dužine ažuriranog sloga
 - Slogu se pridružuje celobrojni ključ, koji nije fizički deo sloga
 - Ključ sloga je jedinstven broj sloga u dat: 1..n
 - Broj n je max br. sloga (MBS)
 - Prvi blok je blok naslova datoteke (BND), koji sadrži svu informaciju potrebnu za pristup slogovima

Relativna datoteka (2/2)

- Svaki blok je podeljen na jednu ili više ćelija fiksne dužine, dovoljno velike da se u njih smesti slog najveće dužine
- Ćelije su numerisane brojevima od 1 do n, bez obzira na to, da li posebna ćelija sadrži ispravan slog

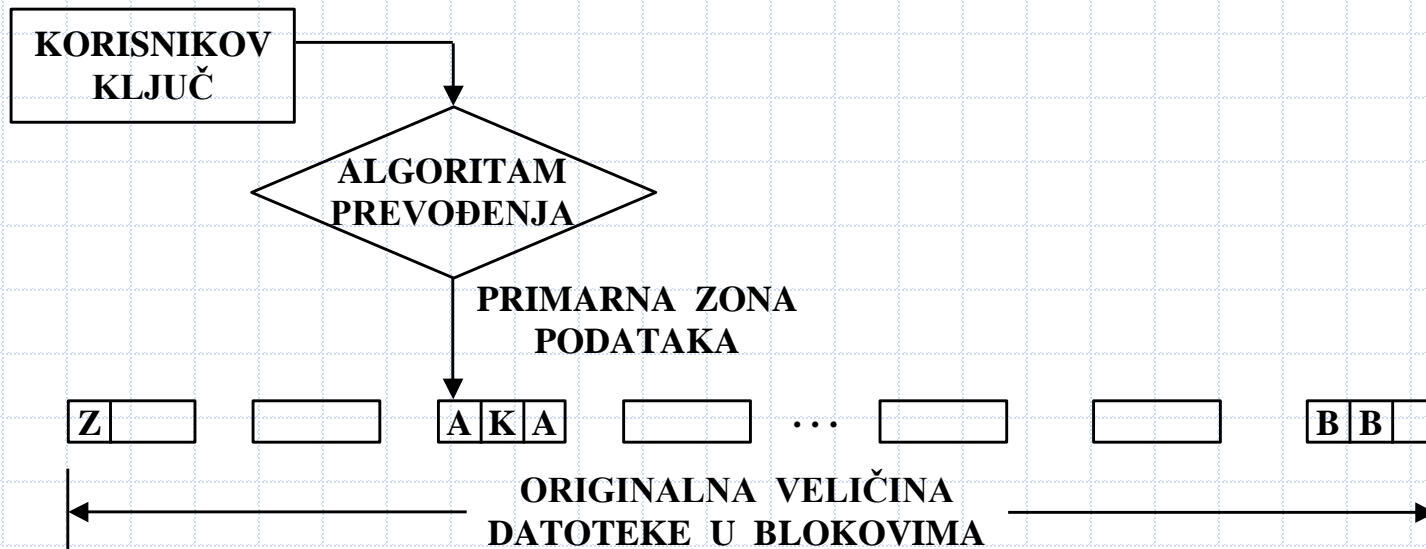


Direktna datoteka (1/3)

- ◆ Def: je datoteka koja omogućava slučajan pristup proizvoljnom slogu, prevođenjem vrednosti ključa u adresu sloga
 - Nedostaci relativne dat: (1) celobrojni ključ (2) dodela dovoljnog prostora radi čuvanja MBS slogova
 - Direktna koristi bilo koji ključ, numerički/alfanumerički
 - Direktna dat. se sastoji od blokova fiksne dužine
 - Početna veličina dat. je deo algoritma prevođenja
 - Primarna zona podataka: je polazni prostor datoteke
 - Oblast prekoračenja: prostor za slogove prekoračioce (koji ne mogu da se smeste u primarnu zonu)

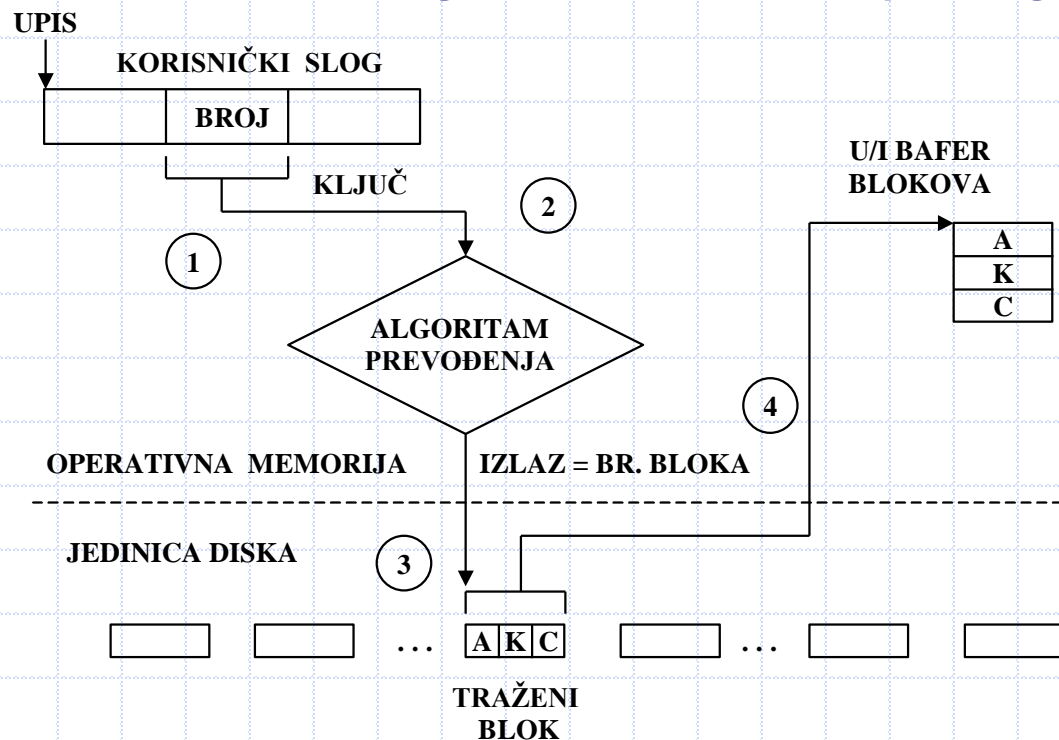
Direktna datoteka (2/3)

- Slogovi se umeću i pronalaze prevođenjem ključa
- Nema unapred određene lokacije za svaki slog
- Blok u direktnoj datoteci može da sadrži jedan, ili više slogova sa identičnom vrednošću ključa
- Nema ćelija, pa ako je blok u koji treba uneti slog pun, novi slog se smešta u zonu prekoračenja



Direktna datoteka (3/3)

- Ključ je fizički deo sloga (polje, npr. broj zaposlenog)
- Rezultat prevođenja je adr. bloka, koji se zatim učitava u UI bafer
- Potom se utvrđuje da li u bloku postoji slog

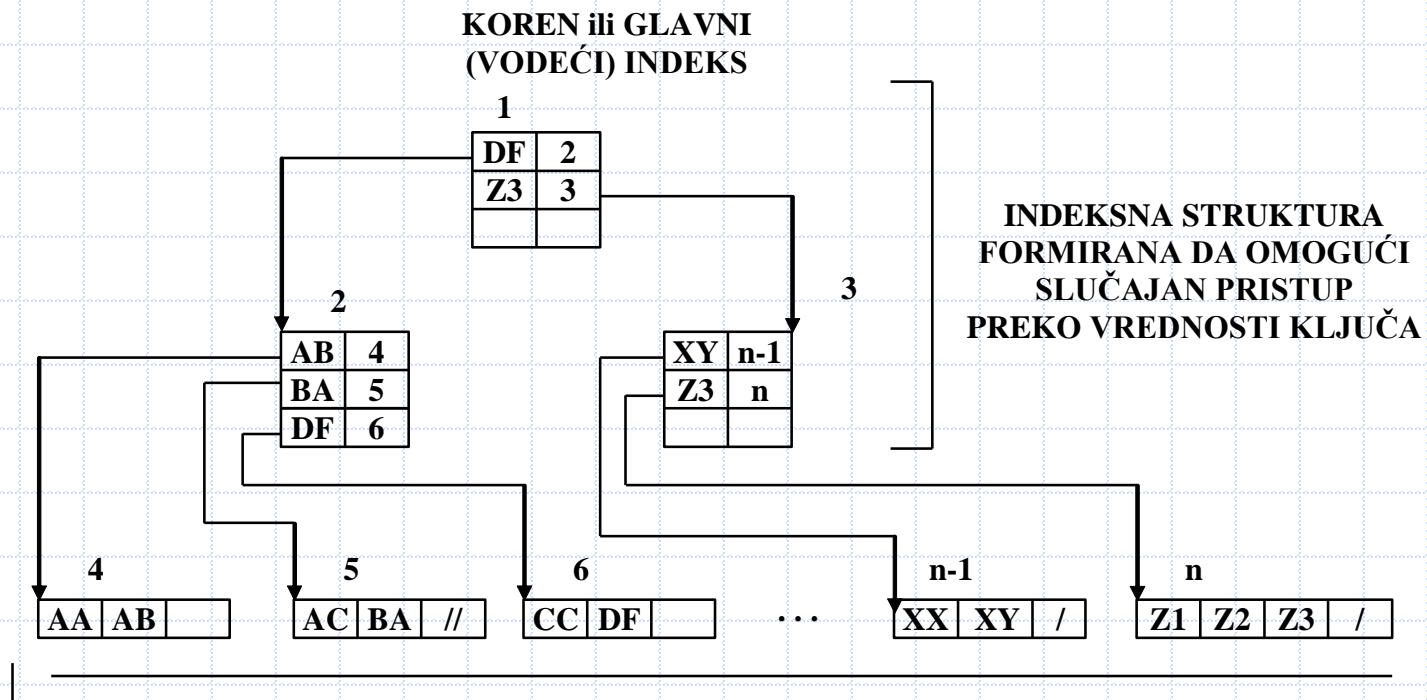


Indeksno-sekvencijalna dat. (1/10)

- ◆ Def: je dat. sa pristupom slogu, sekvencijalno ili slučajno, uz mogućnost definisanja ključeva koji najbolje odgovaraju datoj primeni
 - Korisnički slogovi su sortirani po vrednosti ključeva
 - Indeksna struktura postavljena je iznad ovako uređene datoteke
 - Indeksi dele uređenu sekvencijalnu datoteku prema opsegu ključeva
 - Slučajan pristup se postiže prolaskom duž indeksa, i dalje sekvencijalno pomerajući se unutar uređene sekvencijalne datoteke

Indeksno-sekvencijalna dat. (2/10)

- Npr., blok 2 deli blokove 4, 5 i 6 u tri različita opsega ključeva, blok 5 sadrži sve slogove sa vrednošću ključa većom od AB i manjim ili jednakim BA



SEKVENCIJALNI PRISTUP SE POSTIŽE DRŽANJEM SLOGOVA PODATAKA U NIZU
SA KLJUČEVIMA U SLOGOVIMA

Indeksno-sekvencijalna dat. (3/10)

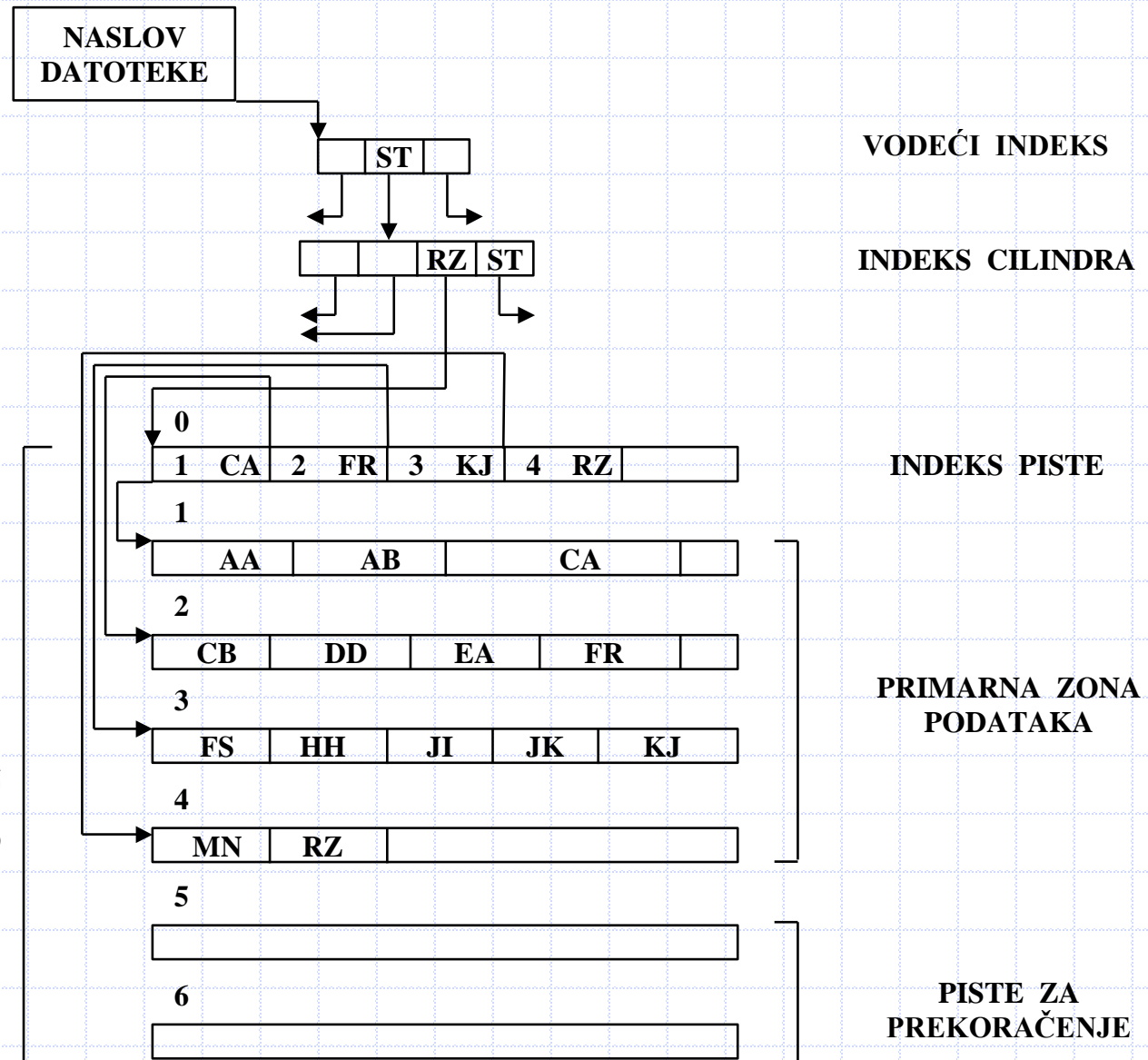
- Indeksna struktura je organizovana u više nivoa
- Prvi nivo: VODEĆI INDEKS (koren IX strukture)
- Dat. može da ima više od jedne IX strukture, ali samo jednu kopiju korisničkog sloga
- Primarni indeks je uređen po primarnom ključu
 - ◆ Alternativni indeksi – alternativni ključevi
- Vodeći indeks služi kao početna tačka za slučajno pronalaženje korisničkog sloga
 - ◆ Indeksni slog sadrži broj cilindra/piste, oznaku da li je to pista indeksa ili podataka, i max ključ na pisti

Indeksno-sekvencijalna dat. (4/10)

- INDEKS CILINDRA je sledeći nivo
- Indeks cilindra može da sadrži ukazivače na druge cilindre ili na najnižem nivou u indeksu mogu ukazivati na indekse piste na cilindru
- Slogovi u indeksu cilindra sadrže broj cilindra i najveći ključ nekog indeksa sloga na tom cilindru

- INDEKS PISTE sadrži po jedan indeksni slog za svaku pistu na definisanom cilindru
- Svaki indeksni slog ukazuje na posebnu pistu unutar cilindra, kao i najveću vrednost ključa na pisti

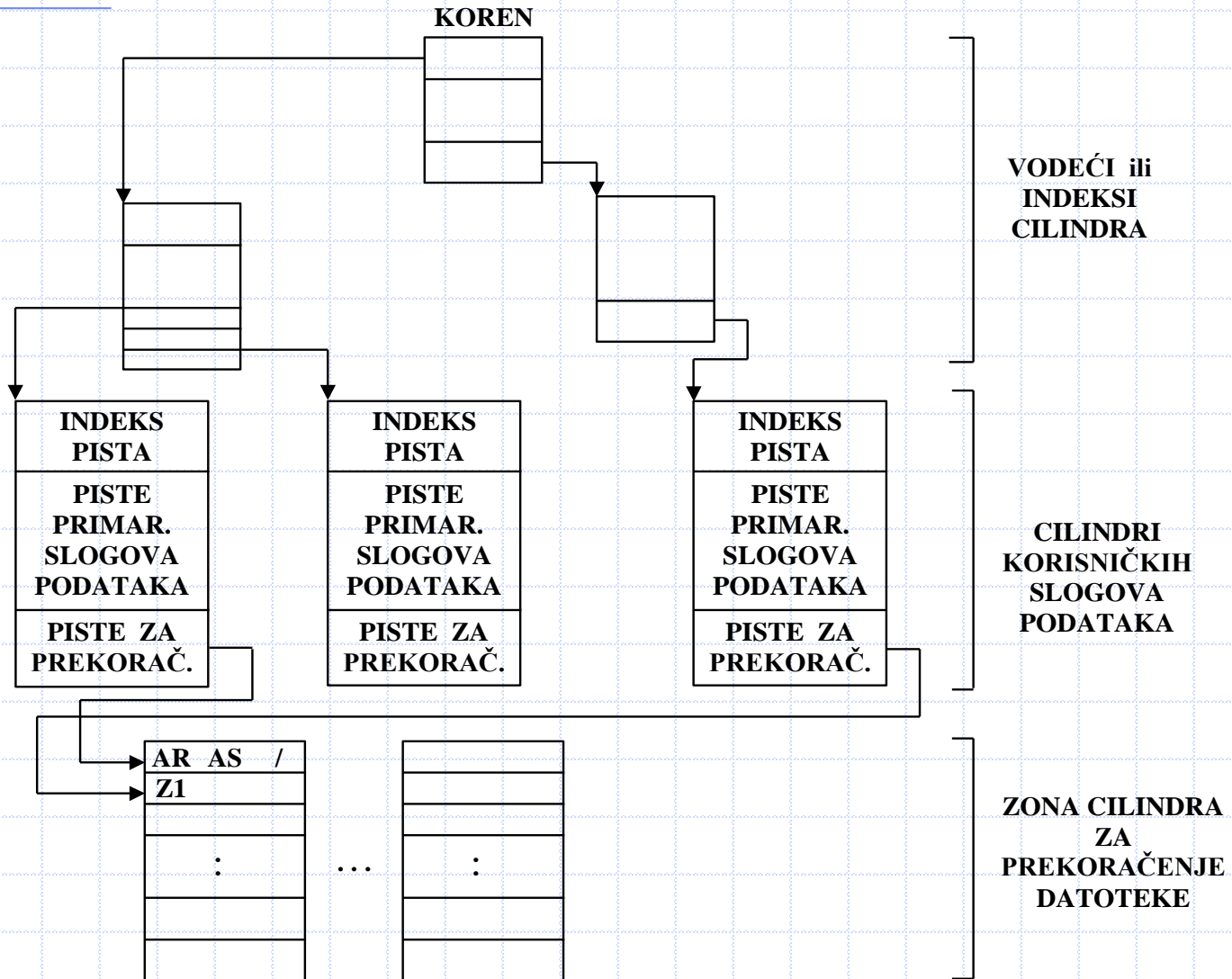
Indeksno-sekvencijalna dat. (5/10)



Indeksno-sekvencijalna dat. (6/10)

- PRIMARNA ZONA PODATAKA predstavlja skup pista koje sadrže slogove uređene po primarnom ključu
- Indeksi pista se ne mogu promeniti sve do reorganizacije datoteke, pa će neki od slogova biti usmeravani u zonu prekoračenja
- ZONA PREKORAČENJA CILINDRA je zona koja se sastoji od jedne ili više pista unutar cilindra namenjenih za čuvanje slogova prekoračioca
- ZONA PREKORAČENJA DATOTEKE je namenjena za prihvatanje svih slogova koji ne mogu stati u zonu prekoračenja cilindara

Indeksno-sekvencijalna dat. (7/10)



Indeksno-sekvencijalna dat. (8/10)

KORISNIČKI SLOG

	KAG	
--	-----	--

KLJUČ

1

VODEĆI INDEKS

CRZ	
HAE	
POZ	16
ZRT	

2

16

JHA	
MRB	28
PQZ	

3

INDEKS
CILINDRA

4

28

1	HZA	IBR	IXX
2	KNH	6	
3			
4			
5			
6	JAR	KAG	KNH
		⋮	
20			

INDEKS
PISTE

5

Indeksno-sekvencijalna dat. (9/10)

- VRSTE PRETRAŽIVANJA:
 - ◆ tačno, generativno i približno pretraživanje ključeva
- Tačno pretraživanje: pretraživanje punim ključem
- Generativno pretraživanje: pretraživanje delom ključa (npr. prvih n znakova u polju ključa)
- Približno pretraživanje: pretraživanje u opsegu ključeva

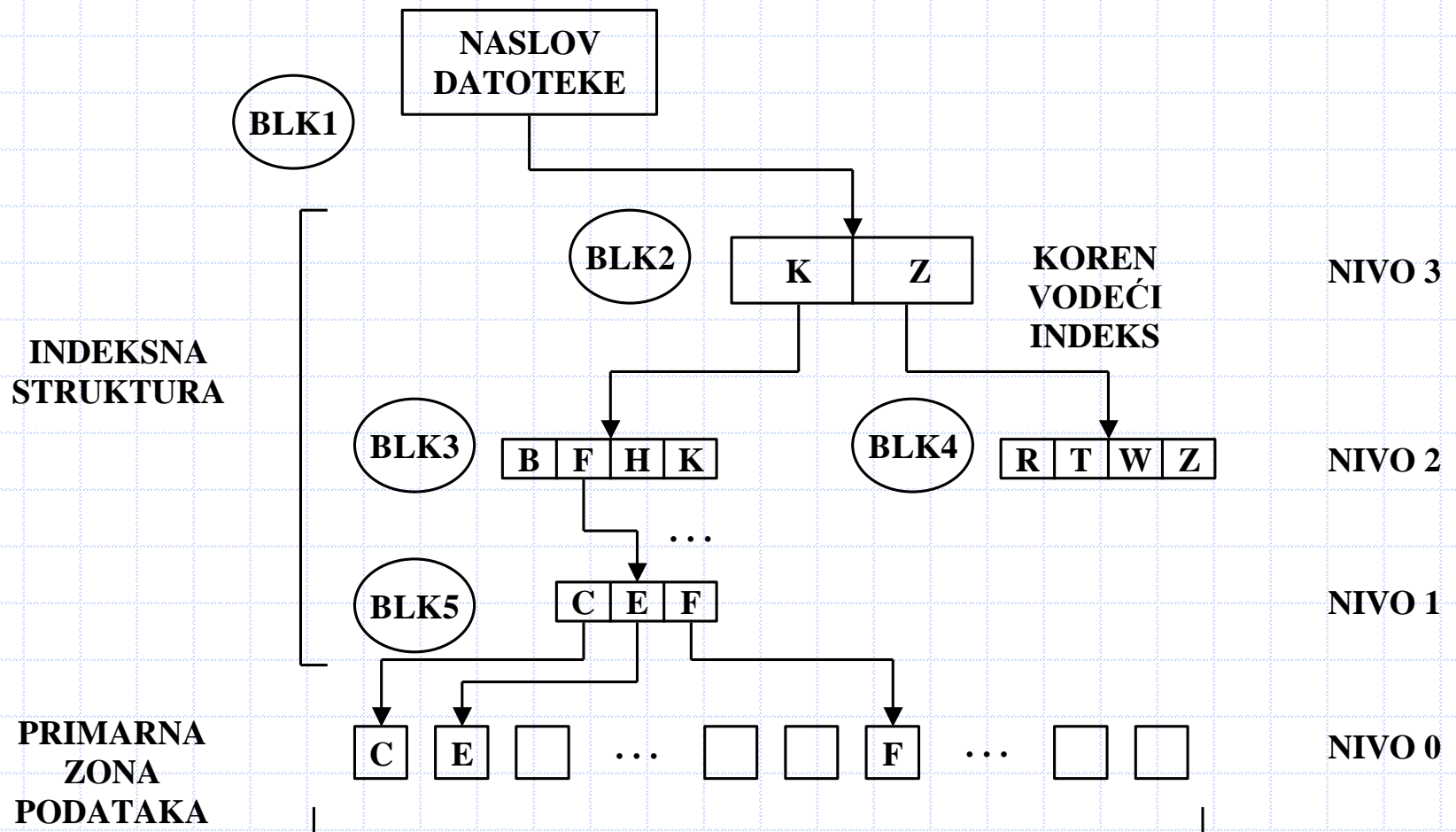
Indeksno-sekvencijalna dat.(10/10)

- ◆ Nedostaci indeksno-sekvencijalne datoteke:
 - Pre formiranja dat. mora se odrediti veličina indeksa, zone podataka, i zona prekoračenja
 - Zone za prekobrojne slogove unose usporenje
 - Struktura datoteke čvrsto je vezana za konfiguraciju jedinice spoljne memorije (diska)
 - Ukoliko se menja konfiguracija, neophodno je ponoviti sav proračun, a datoteka se mora reorganizovati

Indeksna datoteka (1/5)

- ◆ Def: je dat. sa sekvencijalnim ili slučajnim pristupom slogu, pri čemu se indeksni slogovi održavaju u rastućem redosledu ključeva
 - Hijerarhija indeksa ima promenljiv broj nivoa
 - Nivo 0 uvek sadrži korisnički slog, a nivoi od 1 do n su nivoi unutar indeksne strukture
 - Koren indeks je blok na najvišem nivou n
- ◆ Razlike u odnosu na indeksno-sekvencijalnu:
 - Ne postoji vodeći indeks, indeks cilindra i indeks piste; isto tako nema zone za prekobrojne slogove

Indeksna datoteka (2/5)



Indeksna datoteka (3/5)

◆ Održavanje korisničkih slogova:

- Korisnički slogovi se ne održavaju u redosledu porasta ključeva
- Svi novi slogovi se dodaju na kraj datoteke, jer je nivo podataka u datoteci sekvencijalna datoteka
- Održavanje slogova po redosledu nailaska traži jedan slog indeksa po korisničkom slogu podataka

◆ Nedostaci u odnosu na indeksno-sekvencijalnu:

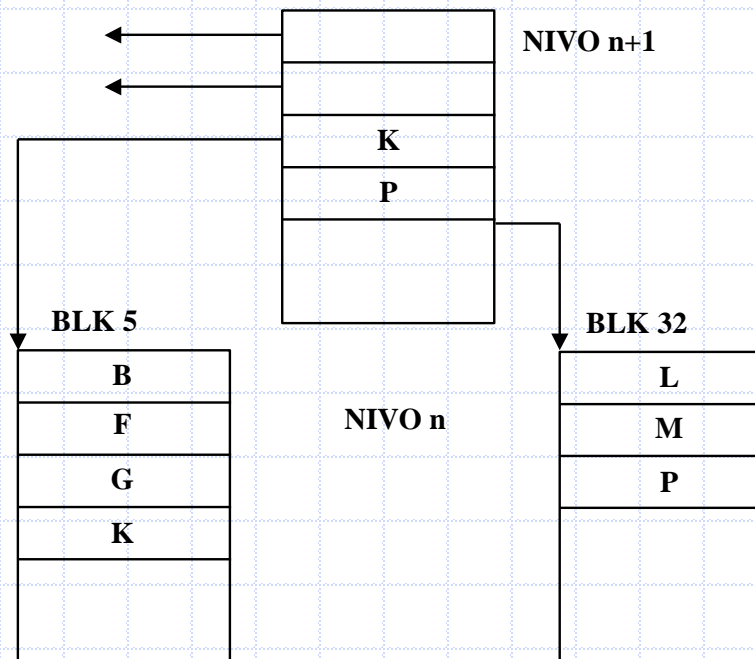
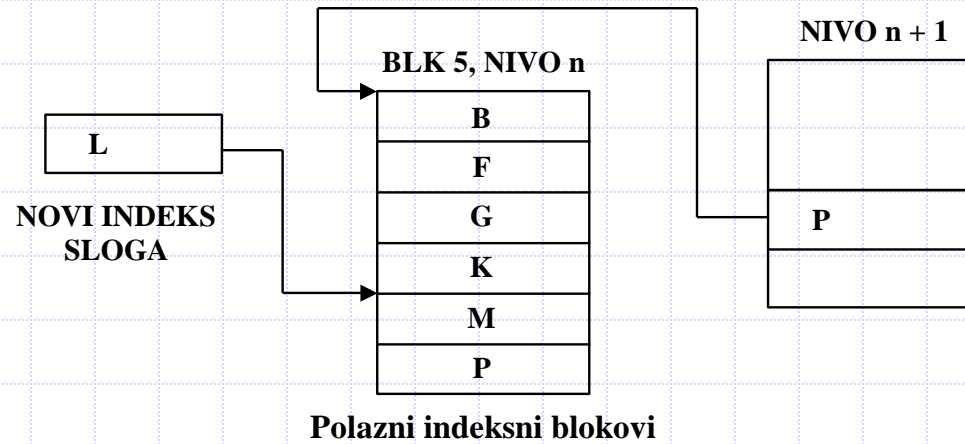
- Veći indeks (jer je tamo bio IX po pisti), veći prostor na disku, i sporiji sekvencijalni pristup

Indeksna datoteka (4/5)

◆ DELJENJE BLOKOVA:

- Javlja se u slučaju kad se u indeksnim blokovima ne može smestiti novi indeksni slog koji je potreban
- Deljenje blokova je proces u kome se sadržaj jednog punog bloka proširuje na dva bloka
- Da bi se indeksni blok podelio, najpre se mora pronaći indeksni slog koji ukazuje na taj indeksni blok
- To je dinamička reorganizacija datoteke, koja se dešava bez prekidanja obrade podataka

Indeksna datoteka (5/5)



Indeksni blokovi posle podele

Tehnike prihvata UI blokova (1/5)

◆ Ulazno-izlazni baferi:

- su zone u OM koje su namenjene za prenos blokova informacija između datoteke i operativne memorije
- FS formira bafere na osnovu atributa datoteke (većina datoteka ima blokove fiksne dužine)

◆ Dva su moguća slučaja u vezi dodele bafera:

- Svakoј datoteci se dodeljuje poseban UI bafer
- Svi UI baferi pod kontrolom sistema, odnosno, izdvaja se jedna jedinstvena, deljiva, zona za sve korisnike
- Nedostatak prvog je neekonomično korišćenje OM, a nedostatak drugog je pojava fragmentacije

Tehnike prihvata UI blokova (2/5)

◆ Koeficijent bafera (KB):

- Pošto se u bafer smešta jedan blok, koeficijent bafera je jednak koeficijentu bloka (br. slogova u bloku)
- Primer sa dva ulazna bafera i $KB = 6$
- Kad se obradi I blok, FS u njega može da prenese sl. blok, odnosno 1-vi slog I bloka postaje 13-ti, itd.

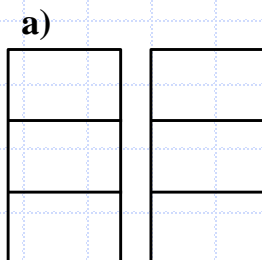
BROJ SLOGA	BROJ SLOGA
1	7
2	8
3	9
4	10
5	11
6	12

Tehnike prihvata UI blokova (3/5)

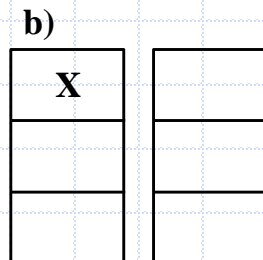
- ◆ Dva načina pristupa konkretnom slogu u baferu:
 - Bafer I-og tipa: slog se kopira u/iz bafera
 - Bafer II-og tipa: slogu u baferu se pristupa direktno preko adrese (bez kopiranja)
- ◆ Rukovalac baferima održava stanje bafera
- ◆ Stanja ulaznog bafera I-og tipa:
 - Stanje A: Ukoliko aplikacija želi da preuzme ulazni slog koji nije zadnji slog tekućeg bafera
 - Stanje B: Aplikacija želi zadnji slog tekućeg bafera
 - Stanje C: Aplikacija želi prvi slog bloka iz bafera, dok je proces popune tog bafera narednim blokom u toku

Tehnike prihvata UI blokova (4/5)

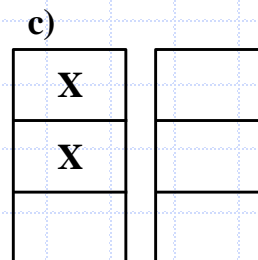
◆ Primer rada sa dvostrukim baferom I-og tipa i $KB=3$:



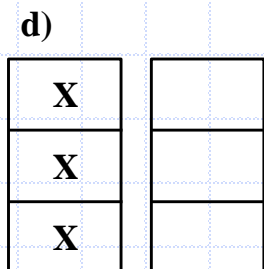
BROJAČ BAFERA=2
BROJAČ SLOGA=3
STANJE A



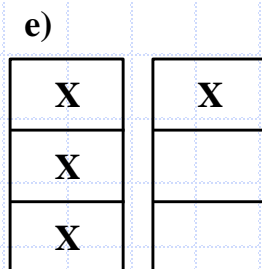
BROJAČ BAFERA=2
BROJAČ SLOGA=2
STANJE A



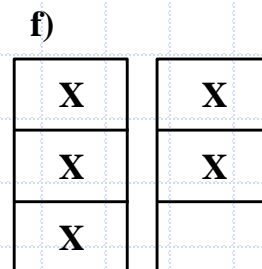
BROJAČ BAFERA=2
BROJAČ SLOGA=1
STANJE B



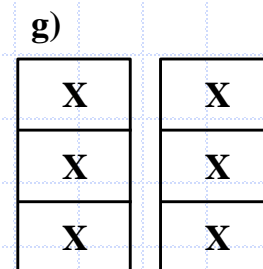
BROJAČ BAFERA=1
BROJAČ SLOGA=3
STANJE A



BROJAČ BAFERA=1
BROJAČ SLOGA=2
STANJE A



BROJAČ BAFERA=1
BROJAČ SLOGA=1
STANJE B



BROJAČ BAFERA=0
BROJAČ SLOGA=0
STANJE C

Tehnike prihvata UI blokova (5/5)

◆ Stanja izlaznog bafera I-og tipa:

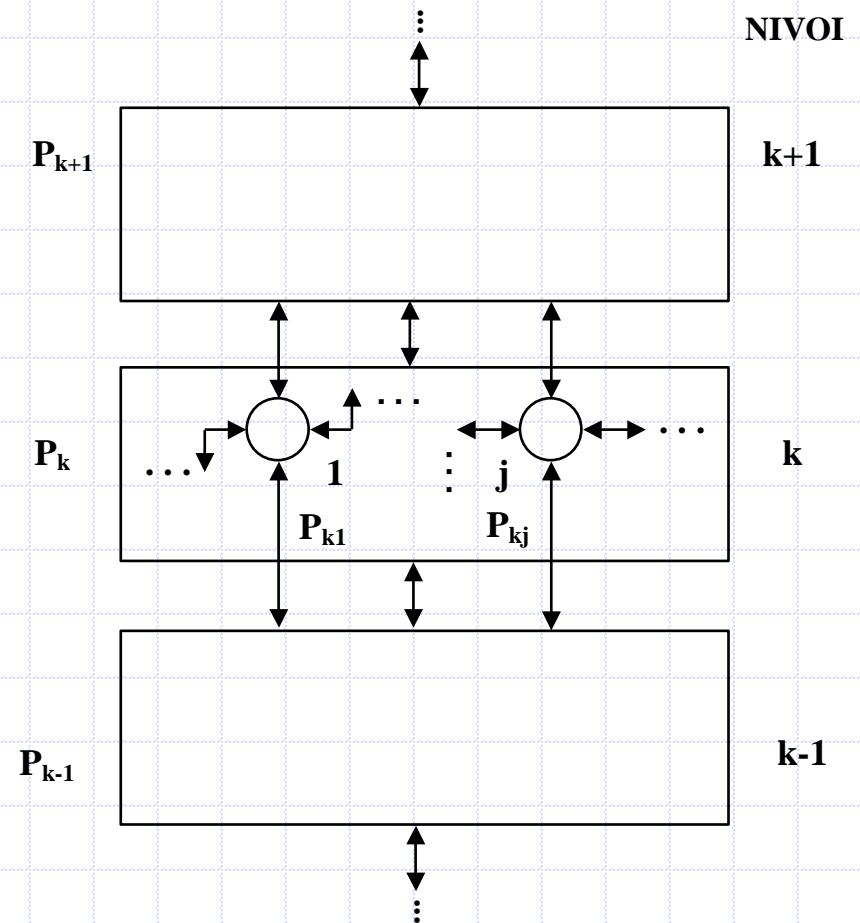
- Ista kao za ulazne bafera, samo je umesto operacije čitanja u pitanju operacija upisa

◆ Stanja ulaznog bafera II-og tipa:

- Stanje A: Ako u baferu postoji najmanje jedan slog
- Stanje B: Ako u baferu nema slogova
- Stanje C: Ako slog ne može da se dobije, jer je to prvi slog u narednom bloku koji još nije popunjen

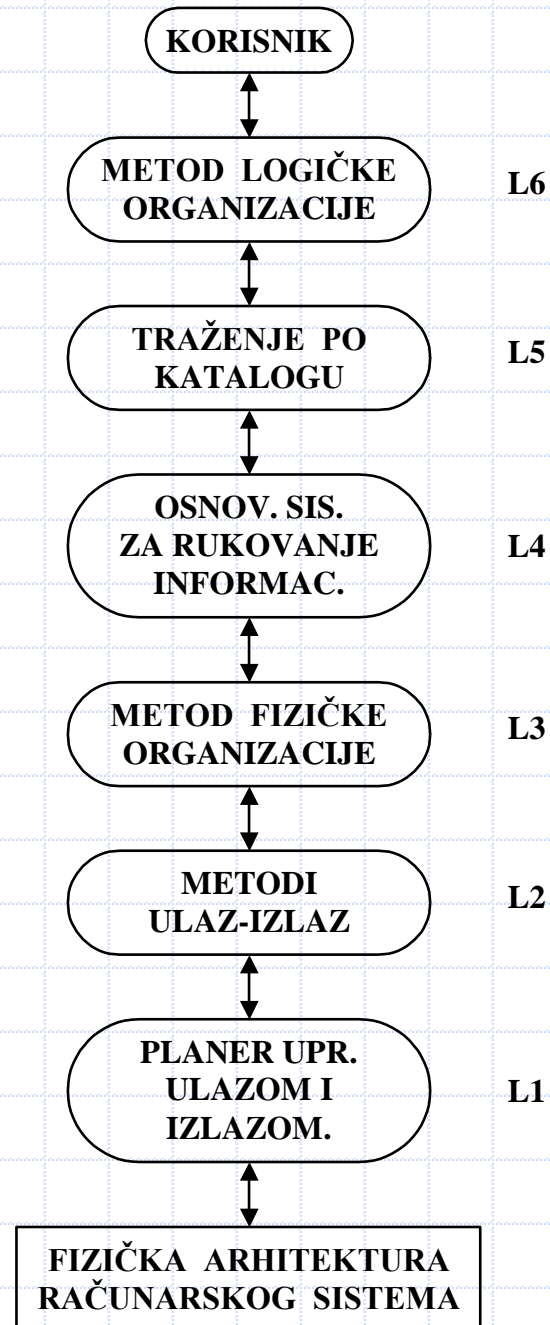
Hijerarhijski model FS-a (1/6)

- ◆ FS je moguće podeliti na više nivoa
 - od najnižeg nivoa, koji počinje od rukovoaca U/I, do najvišeg nivoa, sprege korisnika sa sistemom
 - Svaki nivo je apstraktna mašina (skup procesa)
 - Proces na nivou k može slati poruku samo drugim procesima na tom nivou, na nivou $k+1$ ili na nivou $k-1$



Hijerarhijski model FS-a (2/6)

- ◆ Hijerarhijski model ima šest funkcionalnih nivoa, koji su označeni kao L1-L6



Hijerarhijski model FS-a (3/6)

- ◆ L6 (Metod logičke organizacije / Simbolički FS):
 - Barata sa neformatizovanim virtuelnim datotekama
 - Koristi dva kataloga: simbolički (SK) i osnovni (OK)
 - U SK-u se uz naziv datoteke pridružuje specijalni interni ID broj, kojim se pretražuje OK
 - OK sadrži: veličinu logičkog sloga i njihov broj, adresu prvog fizičkog bloka i informaciju vezanu za zaštitu i kontrolu pristupa
- ◆ L5 (Traženje po katalogu):
 - Pretraživanje OK se vrši na osnovu ID broja, radi pronalaženja tačne pozicije dat. ili njenog deskriptora

Hijerarhijski model FS-a (4/6)

- ◆ L4 (Osnovni sistem za rukovanje informacijama):
 - Dostavlja mu se od gornjeg nivoa ID broj datoteke
 - Ako dat. nije otvorena, informacija o datoteci (njen deskriptor) se smešta u tabelu aktivnih datoteka
 - Zatim se proveru prava pristupa (pomoću matrica kontrole pristupa)
- ◆ L3 (Metod fizičke organizacije):
 - Pretvaranje adrese logičkog sloga u adresu logičkog bajta
 - Zatim se na osnovu adrese logičkog bajta pronalazi fizička adresa bajta

Hijerarhijski model FS-a (5/6)

◆ L2 (Metodi ulaz-izlaza):

- Pretvaranje tražene operacije u niz UI naredbi kanala i kontrolnih signala neophodnih rukovaocu kanalom
- Optimizacije (min kašnjenja pri lociranju piste itd.)

◆ L1 (Planiranje ulaz-izlaza):

- Rukovanje listama zahteva za UI, odnosno planiranje, aktiviranje i upravljanje zahtevima za UI radnjama
- Na ovom nivou se isto tako vrši obrada UI prekida

Hijerarhijski model FS-a (6/6)

◆ Jednostavna sprege između nivoa:

- Tok ka nižim nivoima su zahtevi za izvršavanjem operacije sa potrebnim podacima
- Tok ka višim nivoima su poruke o načinu završetka tražene operacije

