OPERATIVNI SISTEMI

IX - Sekundarne memorije

IX - Sekundarne memorije

S A D R Ž A J

9.1 Struktura diskova

Geometrija diskova

- disk čini skup **rotacionih kružnih ploča** koje rotiraju oko zajedničke ose
- površine ploča su presvučene **magnetnim materijalom**
- svaka površina ima pridruženu **glavu za čitanje i pisanje**
  - čitaju ili upisuju podatke sa magnetnih ploča
  - linearno se pokreću pomoću sopstvenog servo-sistema
  - na taj način im je uz rotaciju ploča omogućen pristup svim delovima magnetne površine
- računar i disk komuniciraju putem **disk kontrolera** (*disk controller)*
  - disk kontroleri pružaju **interfejs ka ostatku računara**
  - računar ne mora da zna način rada niti da kontoliše elektro-mehaniku diska

9.1 Struktura diskova

Geometrija diskova

- dodatne funkcije kontrolera
  - baferovanje podataka koje treba upisati na disk
  - keširanje diskova
  - automatsko obeležavanje neispravnih sektora diska
- površina diska je podeljena u koncentrične prstenove - **staze** (*tracks)*
- svaka staza je dalje podeljena na **sektore** (*sectors)*
- tipična količina podataka koja se može upisati u jedan sektor je **512 bajtova**
  - to je najmanja količina podataka koja se može upisati na disk ili pročitati sa diska
- sve površine magnetnih ploča su jednako podeljene na staze i sektore
  - to znači da se glave za čitanje i pisanje na svim pločama diska u jednom vremenskom trenutku nalaze na istim stazama

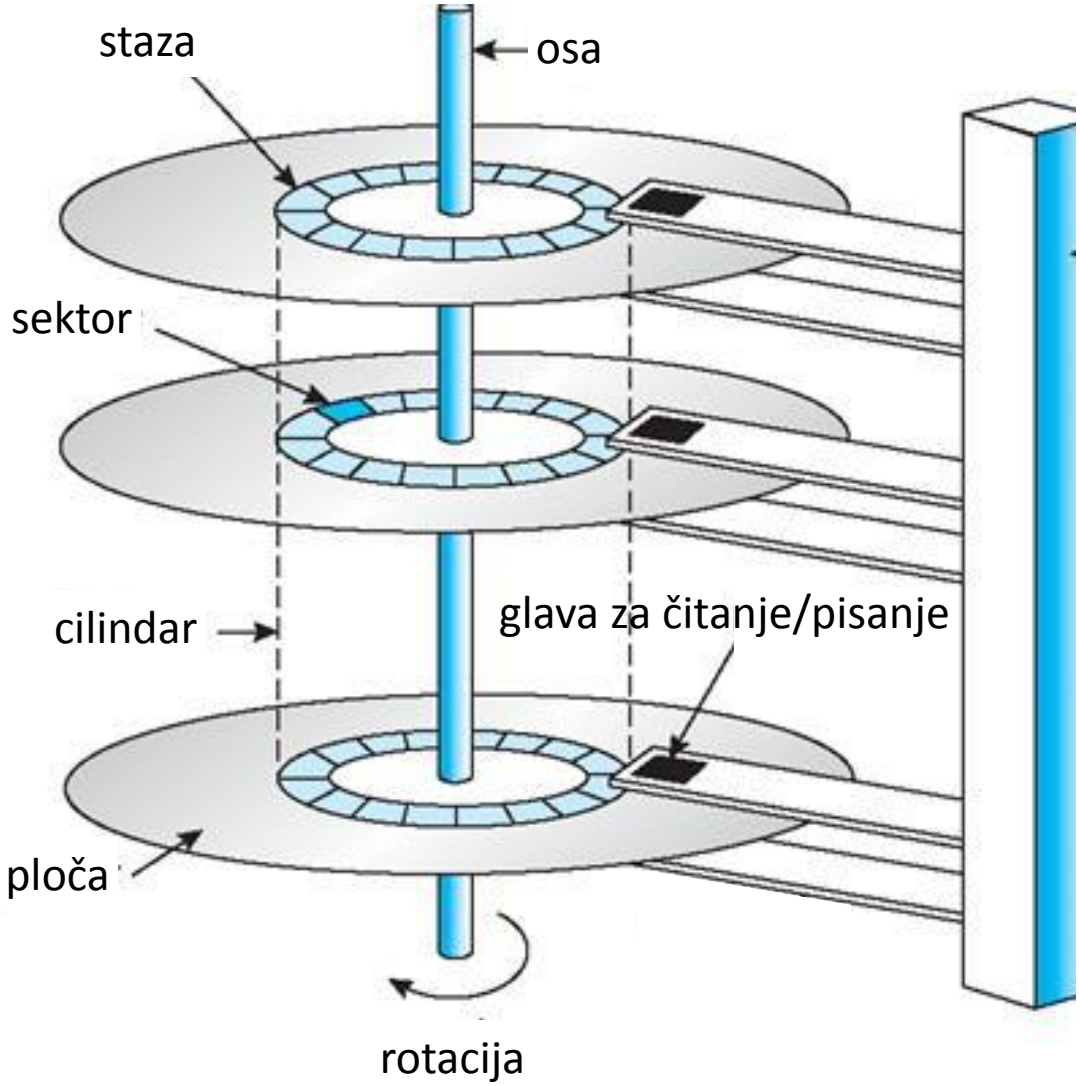
9.1 Struktura diskova

Geometrija diskova

- ekvidistantne staze svih ploča čine jedan **cilindar** (*cylinder)*
- datoteke koje nisu smeštene u okviru jednog cilindra su **fragmentisane**
  - pomeranje glava sa jedne staze na drugu prilikom čitanja ovakvih datoteka unosi **kašnjenje**
  - performanse diska se mogu uvećati smeštanjem datoteke **u okviru jednog cilindra** kad god je to moguće
- geometrija diska je u opštem slučaju određena
  - brojem magnetnih površina (odnosno glava za čitanje i pisanje)
  - brojem cilindara
  - brojem sektora
- **trodimenzionalnim adresiranjem** (*head-cylinder-sector addressing)* može se pristupi svim delovima diska
  - primer podatka koji je upisan na drugu površinu, u stazu 3, na sektoru 5: *(head, cylinder, sector)=(2,3,5).*

9.1 Struktura diskova

Geometrija diskova



9.1 Struktura diskova

Savremeni disk uređaji

- **zapažaju se sledeće smernice razvoja disk uređaja**
- smanjivanje vremena pozicioniranja (*seek time)*
- povećanje rotacione brzine
- povećanje gustine magnetnog mediuma
- zonska tehnika (*Bit Zone Recording)*
  - sve veća gustina staza iziskuje efkasnije korišćenje magneta
  - unutrašnje staze (bliže centru) imaju manju površinu pa samim tim i manje magneta
  - zato se uvodi tehnika zona na disku gde se cilindri grupišu u zone iste gustine
    - time se povećava kapacitet diska i brzina čitanja sa medijuma
    - međutim upravljanje promenljivom gustinom staza je složenije

9.1 Struktura diskova

Savremeni disk uređaji

- **zapažaju se sledeće smernice razvoja disk uređaja** (*nastavak*)
- rezervni regioni na disku (*spare regions)* za zamenu defektnih blokova
  - alternativni sektori za upravljanje defektima
  - ovaj prostor se rezerviše na kraju staza, cilindara i zona
  - time se i u slučaju pojava defekata na disku održava 100% naznačenog kapaciteta
- upotreba procesora solidne snage na disku
  - kontrolna ploča diska sadrži sopstveni procesor i RAM
- keširanje na disk uređaju (*on-board cache)*
- obrada komandi u redu čekanja (*command queuing*)
  - disk prima više komandi u redu čekanja a zatim organizuje njihov optimalni redosled izvršavanja
  - interni disk raspoređivač (*scheduler*) uvek mnogo efikasnije radi nego bilo koji eksterni, jer najbolje poznaje svoje karakteristike

9.1 Struktura diskova

IDE, SATA i SCSI klase diskova

- **IDE** (*Integrated Drive Electronics)* tj **ATA** (A*dvanced Technology Attachment)*
  - kontroler integrisan na matičnoj ploči
  - brzina od 33 do 133MB/s
  - dva kanala: primarni i sekundarni
    - na svaki kanal se mogu vezati najviše dva uređaja u odnosu nadređen/podređen (*master/slave*)
    - uređaji vezani na različite kanale mogu istovremeno da šalju ili primaju podatke od računara
    - na jednom kanalu, samo jedan uređaj može biti aktivan u jednom trenutku
- **SATA** (*serial* ATA)
  - kontroler integrisan na matičnoj ploči
  - koriste tanje kablove (olakšava rashlađivanje)
  - mogu se isključivati i priključivati u toku rada računara
  - brži su (do 500MB/s)
  - pouzdaniji su

9.1 Struktura diskova

IDE, SATA i SCSI klase diskova

- **SCSI** (*Small Computer System Interface*)
  - kontroler nije integrisan na matičnoj ploči
  - na kontroler je moguće vezati od 7 do 15 uređaja
  - SCSI uređaji se ne nalaze u master-slave odnosu već se vezuju prema prioritetima
  - prioritet uređaja određen je njegovim ID koji se postavlja preko džampera
  - ID =0 (najviši prioritet), ID =15 (najniži prioriter), ID =7 (rezervisan za SCSI kontroler)

9.2 Priprema diskova za rad

- priprema diskova za rad obuhvata sledeće procedure
  - formatiranje diskova
  - izrada particija
  - formiranje sistema datoteka (opisano detaljno u narednoj temi 10)

Formatiranje diskova

- formatiranje diska na niskom nivou (*low-level formatting*)
  - u prošlosti je bilo neophodno kako bi se disk pripremio za rad
  - upisuje oznake koje predstavljaju granice staza i sektora
    - koncentrični krug se deli na sektore
    - svaki sektor ima zaglavlje (*header*) u kojem je navedena adresa sektora, zatim sledi 512 bajtova podataka i na kraju završni zapis (*trailer*) u kojem se obično navodi ECC (*Error Corecction Fault*) tj. informacija za detekciju i oporavak od greške
  - današnji diskovi su već fabrički formatirani na niskom nivou
- formatiranje na visokom nivou (*high-level formatting*)
  - proces formiranja sistema datoteka
  - korisnicima DOS/Windows sistema poznato kao komanda *format*

9.2 Priprema diskova za rad

Izrada particija

- nakon formatiranja diska, pre formiranja sistema datoteka neophodno je formirati particiju
  - po potrebi, može se praviti jedna ili više particija
  - svaka particija se ponaša kao zasebni disk
- informacije o svim particijama na disku čuvaju se u prvom sektoru prve staze poznatom pod imenom glavni startni zapis (**MBR** - *Master Boot Record)*
  - u startnom sektoru se nalazi program (*bootstrap* rutina) čijim se pokretanjem započinje učitavanje OS-a u RAM
- u prvo vreme bilo je moguće kreirati najviše četiri particije po jednom disku
- problem je rešen uvođenjem produžene particije (*extended*),
  - okvir u kome se mogu kreirati nekoliko logičkih particija
- logičke particije se ponašaju kao primarne, ali se razlikuju po načinu kreiranja

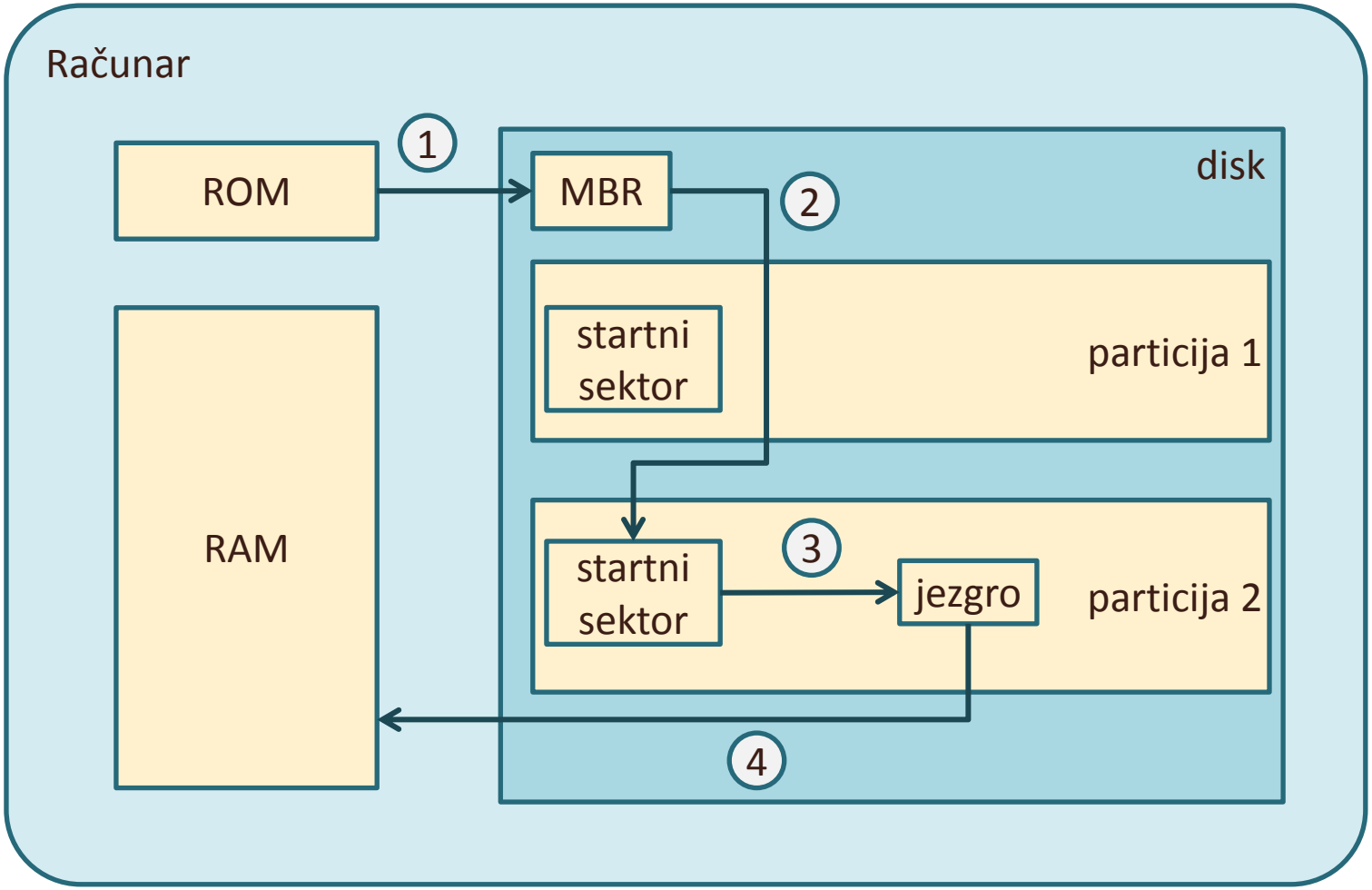
9.2 Priprema diskova za rad

Bootstrap rutina

- kada se računar uključi, BIOS izvršava POST rutinu (*Power On Self Test*)
  - POST predstavlja seriju testova hardvera
- **podizanje sistema** (*boot*) je procedura koja se izvršava u cilju dovođenja sistema u operativno stanje
- primer (MBR)
  - kôd upisan u prvom MBR najpre identifikuje **aktivnu particiju** u particionoj tabeli
  - zatim se izvršava **kôd upisan u boot sektoru** aktivne particije
  - program u *boot* sektoru je zadužen da **pokrene punjenje RAM memorije OS-om**
    - delovi kôda u toj fazi nalaze se na fiksnim područjima diska, a ne u sistemima datoteka
    - u toj fazi nema kernela, pa nemamo podršku za sistem datoteka
  - rana faza podizanja OS-a se završava učitavanjem jezgra

9.2 Priprema diskova za rad

Bootstrap rutina



9.3 Nivoi keširanja diskova

- **keširanje na nivou OS-a** (*built-in file caching)*
  - svaki OS ima sopstveno keširanje
  - keširanje na ovom nivou se naziva *file cashing* zato što keš sadrži kopiju datoteka
- **keširanje na nivou disk kontrolera** (*HBA level caching)*
  - ne koristi se za klasične disk kontrolere budući da baferski kontroleri u kombinaciji sa keširanje na nivou OS-a daju bolje rezultate, po mnogo nižoj ceni
  - nezaobilazni deo u najkvalitetnijim i najsloženijim disk kontrolerima kao što su RAID kontroleri
- **keširanje na nivou disk uređaja** (*disk drive level caching)*
  - diskovi imaju solidne procesore i veliku količinu memorije koja služi za keširanje
  - disk keš memorija optimalno je mesto za tehniku prediktivnog čitanja
    - disk najbolje poznanje svoj servo-sistem i raspored podataka na medijumu

9.3 Nivoi keširanja diskova

- **RAID keširanje** (*caching in RAID)*
  - svaki RAID kontroler pored obezbeđivanja RAID funkcionalnosti predstavlja i potpuni keš kontroler
- **keširanje na nivou aplikacije** (*application level caching)*
  - svi pomenuti keš nivoi su po prirodi opšte namene, generalni
  - zahvaljući dobrom poznavanju sopstvenih potreba u radu sa diskom, kvalitena aplikacija kešira disk saglasno svojim potrebama, znatno bolje nego generalni keš na nivou OS-a

9.4 Raspoređivanje zahteva za rad sa diskom

- U/I uređaji predstavljaju usko grlo računarskog sistema po pitanju performansi
  - vreme pristupa kod diska zavisi od
    - **vremena pozicioniranja glava** sa tekuće pozicije na zahtevani cilindar
    - **vremena rotacionog kašnjenja** (zavisi od brzine okretanja rotacionih površina diska)
    - **brzine transfera podataka** sa magnetnog medijuma (zavisi od gustine medijuma i brzine okretanja rotacionih površina diska)
- **brzina disk transfera** (*bandwidth*) je količnik ukupnog broja prenetih bajtova i ukupnog vremena
- u višeprocesnoj okolini u jednom trenutku postoji veliki broj zahteva za rad sa diskom
- pravilnim **raspoređivanjem ovih zahteva** (*disk scheduling*) ukupno vreme pozicioniranja ili rotacionog kašnjenja se može smanjiti

9.4 Raspoređivanje zahteva za rad sa diskom

- svaki zahtev koji je upućen disku sadrži **sledeće informacije**
  - da li se zahteva operacija čitanja ili pisanja
  - adresu bloka na disku
  - adresu memorijskog bafera
  - broj bajtova koje treba preneti
- više zahteva može stići istovremeno
  - u jednom trenutku disk može obraditi samo jedan
- postoje više algoritama koji će obaviti raspoređivanje zahteva za rad sa diskom: FCFS, SSTF, SCAN, C-SCAN, LOOK i C-LOOK
- relativne performanse algoritama izrazićemo ukupnim brojem cilindara koje glave za čitanje i pisanje prelaze pri opsluživanju zahteva (pomerajima glave)

9.4 Raspoređivanje zahteva za rad sa diskom

Algoritam FCFS (*First Come, First Served)*

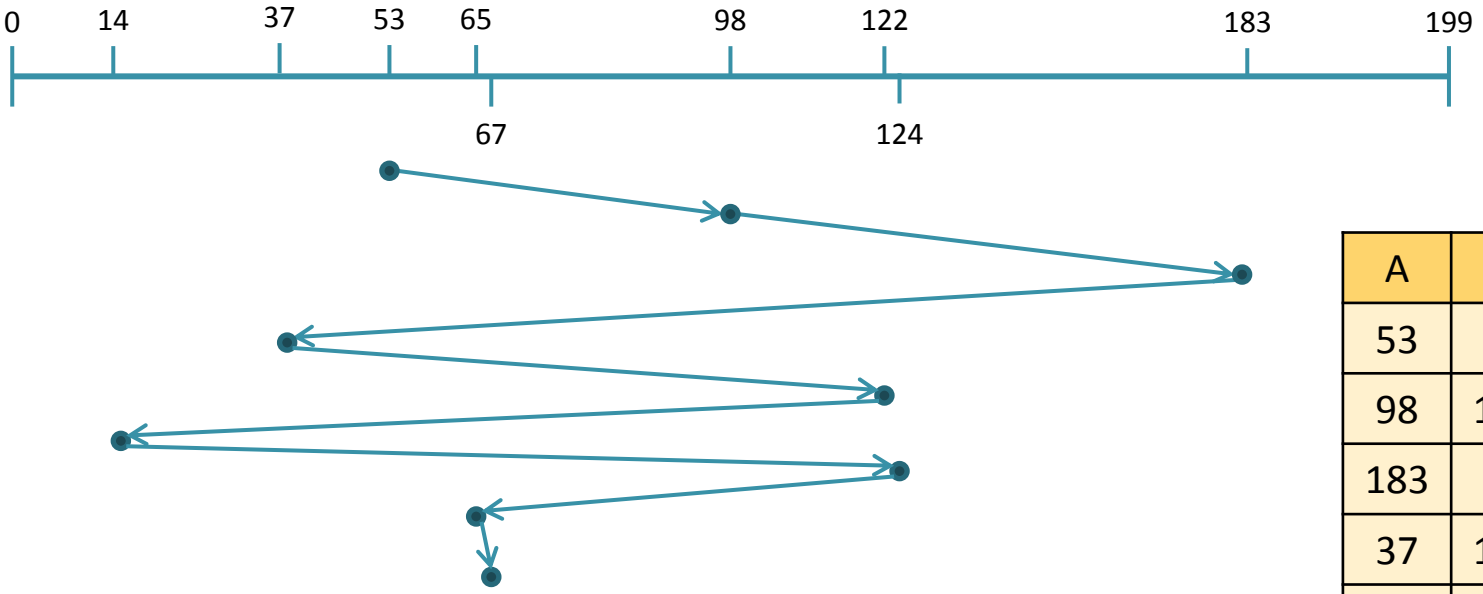
- najjednostavniji algoritam
- zahteve prosleđuje u onom redosledu u kome su stigli
- obezbeđuje krajnje fer odnose prema prispelim zahtevima, ali i znatno loše performanse
- primer
  - trenutna pozicija glava za čitanje i pisanje je na cilindru 53
  - u red čekanja za disk zahtevi pristižu po sledećem redu: 98, 183, 37, 122, 14, 124, 65, 67
  - ukupni pomeraj glava diska iznosi 640 cilindara

9.4 Raspoređivanje zahteva za rad sa diskom

Algoritam FCFS (engl *First Come, First Served)*

početna pozicija glave: 53

red čekanja : 98, 183, 37, 122, 14, 124, 65, 67



A	B	P
53	98	45
98	183	85
183	37	146
37	122	85
122	14	108
14	124	110
124	65	59
65	67	2
ukupno		640

A	početna pozicija
B	ciljna pozicija
P	broj pomeraja

9.4 Raspoređivanje zahteva za rad sa diskom

Algoritam SSTF (*Shortest Seek Time First)*

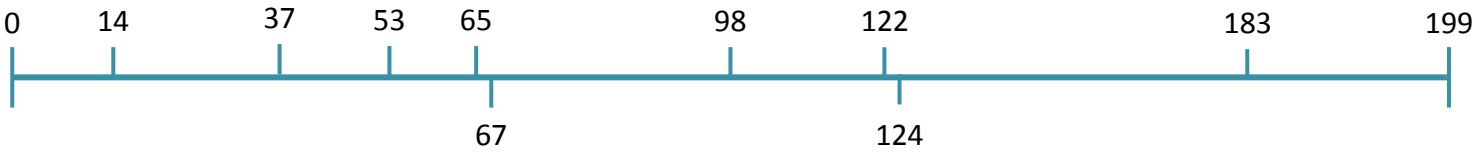
- od prispelih zahteva najpre se uzima onaj koji će izazvati **najmanji pomeraj glava** (*seek time)*
- primer
  - trenutna pozicija glava za čitanje i pisanje je na cilindru 53
  - u red čekanja za disk zahtevi pristižu po sledećem redu: 98, 183, 37, 122, 14, 124, 65, 67
  - ukupni pomeraj glava diska iznosi 236 cilindara
- napomene
  - podseća na SJF (*Shortest Job First)* algoritam za raspoređivanje procesora
  - optimalan je po pitanju **vremena pozicioniranja**
  - kod SSTF algoritma prisutan je problem **zakucavanja** (*starvation)*
    - glave mogu ostati veoma dugo u jednoj zoni opslužujući zahteve koji unose male pomeraje
    - zahtevi čiju su cilindri daleko od trenutne pozicije mogu dugo čekati u redu

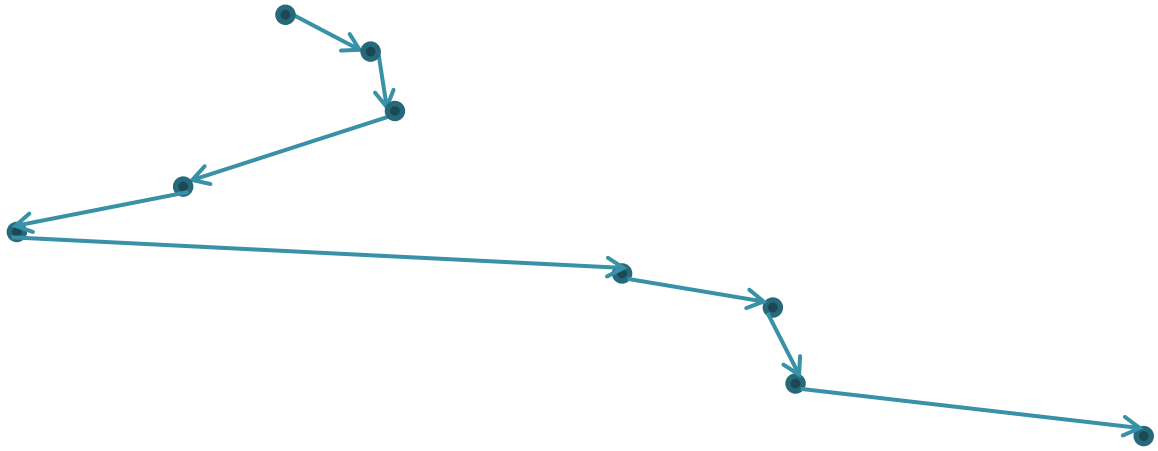
9.4 Raspoređivanje zahteva za rad sa diskom

Algoritam SSTF (*Shortest Seek Time First)*

početna pozicija glave: 53

red čekanja : 98, 183, 37, 122, 14, 124, 65, 67





A	B	P
53	65	12
65	67	2
67	37	30
37	14	23
14	98	84
98	122	24
122	124	2
124	183	59
ukupno		236

A	početna pozicija
B	ciljna pozicija
P	broj pomeraja

9.4 Raspoređivanje zahteva za rad sa diskom

Algoritam SCAN

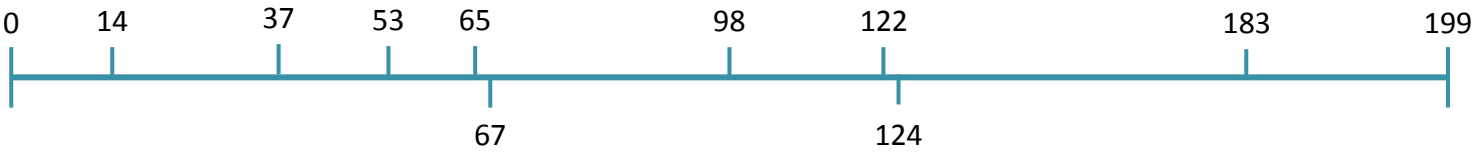
- radi na principu lifta koji se naizmenično kreće od prizemlja do vrha zgrade
- algoritam naizmenično pomera glave od početka do kraja diska i unazad i opslužuje zahteve koji se nalaze na tekućem cilindru
- na ovaj način se rešava problem zakucavanja
- primer
  - trenutna pozicija glava za čitanje i pisanje je na cilindru 53
  - u red čekanja za disk zahtevi pristižu po sledećem redu: 98, 183, 37, 122, 14, 124, 65, 67
  - ukupni pomeraj glava diska iznosi 236 cilindara
- napomena
  - prilikom obrade zahteva, SCAN daje prednost unutrašnjim cilindrima u odnosu na periferne

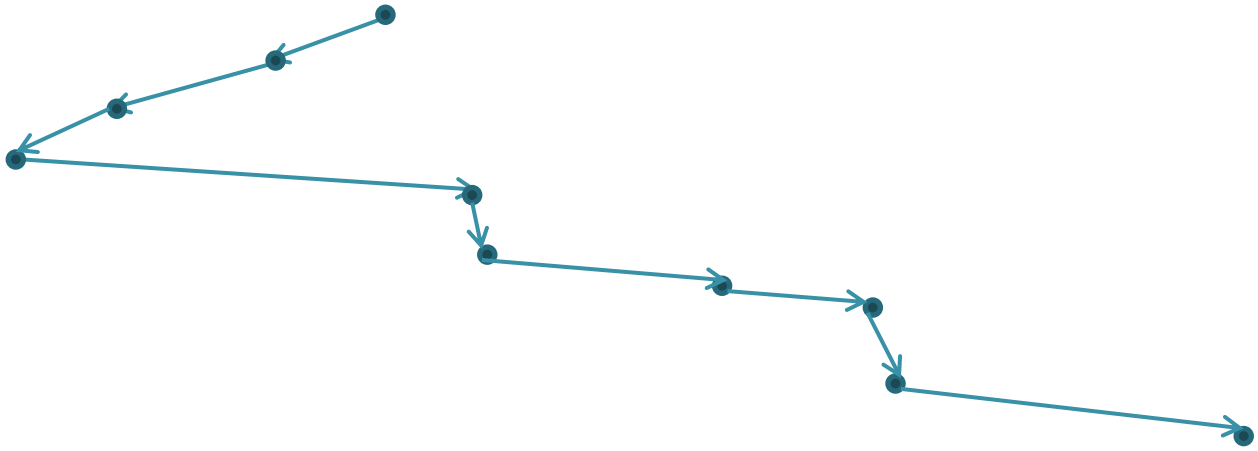
9.4 Raspoređivanje zahteva za rad sa diskom

Algoritam SCAN

početna pozicija glave: 53

red čekanja : 98, 183, 37, 122, 14, 124, 65, 67





A	B	P
53	37	16
37	14	23
14	0	14
0	65	65
65	67	2
67	98	31
98	122	24
122	124	2
124	183	59
ukupno		236

A	početna pozicija
B	ciljna pozicija
P	broj pomeraja

9.4 Raspoređivanje zahteva za rad sa diskom

Algoritam C-SCAN (*Circular SCAN*)

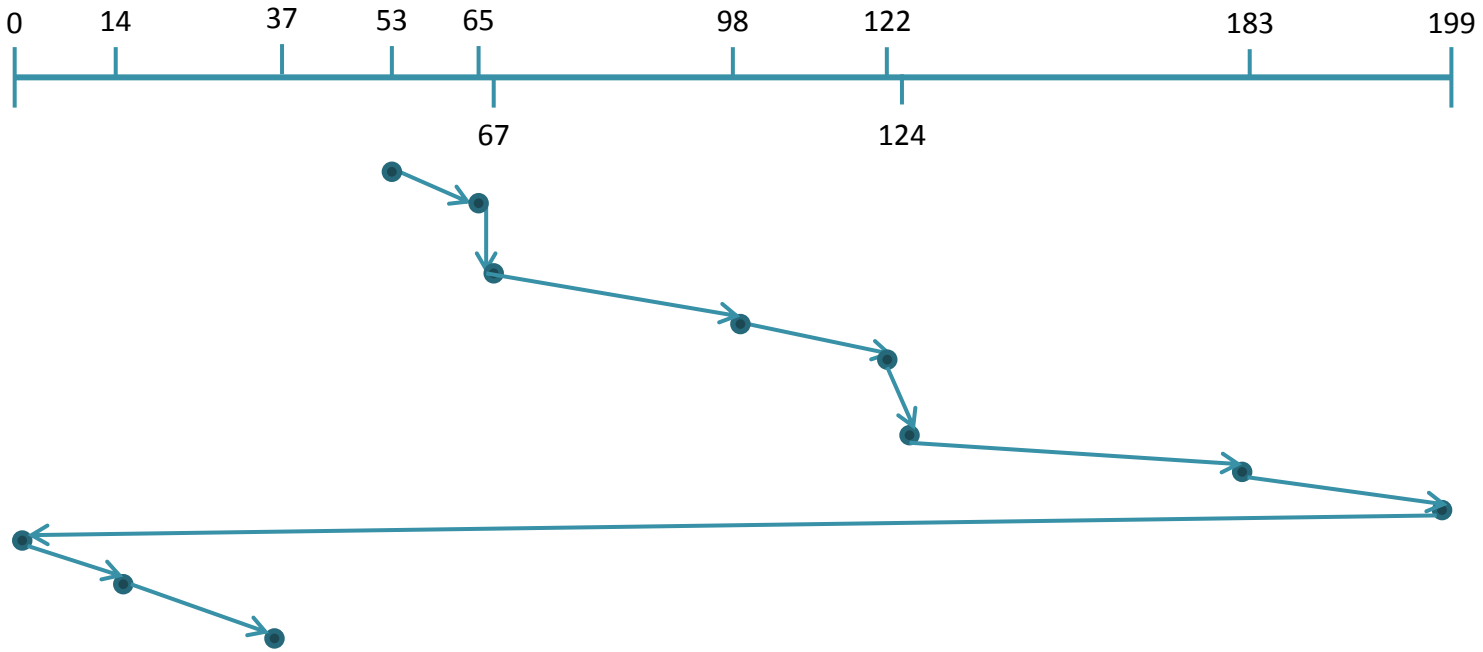
- varijanta SCAN algoritma koja **razrešava problem favorizovanja unutrašnjih cilindara**
- izmena se sastoji u tome da se zahtevi opslužuju samo u jednom smeru
  - kada glave dođu do poslednjeg cilindra, pomeraju se na početak, ne opslužujući zahteve na tom putu
  - posle toga se nastavlja opsluživanje zahteva od početnog do krajnjeg cilindra
- primer
  - trenutna pozicija glava za čitanje i pisanje je na cilindru 53
  - u red čekanja za disk zahtevi pristižu po sledećem redu: 98, 183, 37, 122, 14, 124, 65, 67
  - ukupni pomeraj glava diska iznosi 382 cilindra
    - kod manjeg broja autora, navodi se da se pomeraj 199-0 ne bi trebao računati u krajnji zbir, budući da se on vrlo brzo izvršava

9.4 Raspoređivanje zahteva za rad sa diskom

Algoritam C-SCAN

početna pozicija glave: 53

red čekanja : 98, 183, 37, 122, 14, 124, 65, 67



A	B	P
53	65	12
65	67	2
67	98	31
98	122	24
122	124	2
124	183	59
183	199	16
199	0	199
0	14	14
14	37	23
ukupno		382

A	početna pozicija
B	ciljna pozicija
P	broj pomeraja

9.4 Raspoređivanje zahteva za rad sa diskom

Algoritmi LOOK i C-LOOK (*Circular* LOOK)

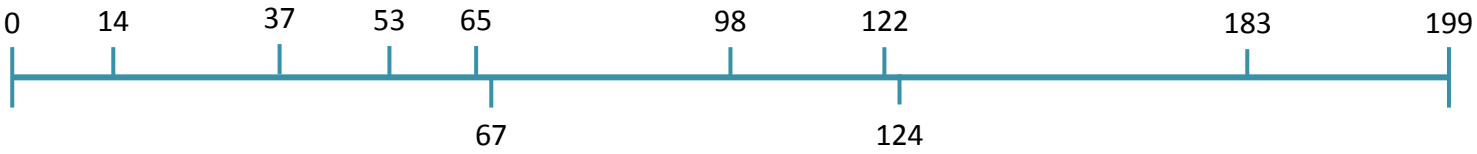
- modifikacije SCAN i C-SCAN algoritama
- glave se ne pomeraju do poslednjeg zahteva koji se nalazi u redu čekanja u tom smeru
  - LOOK opslužuje zahteve u oba smera
  - C-LOOK opslužuje zahteve samo u rastućem smeru do poslednjeg zahteva u redu nakon čega se vraća na zahtev najbliži početku diska
- primeri
  - trenutna pozicija glava za čitanje i pisanje je na cilindru 53
  - u red čekanja za disk zahtevi pristižu po sledećem redu: 98, 183, 37, 122, 14, 124, 65, 67
  - ukupni pomeraj glava diska iznosi 208 za LOOK i 322 za C-LOOK
- napomena
  - naziv su dobili po tome što "gledaju" na kom se cilindru nalazi poslednji zahtev u tom smeru
  - u praksi se umesto SCAN algoritama uvek koriste LOOK algoritmi

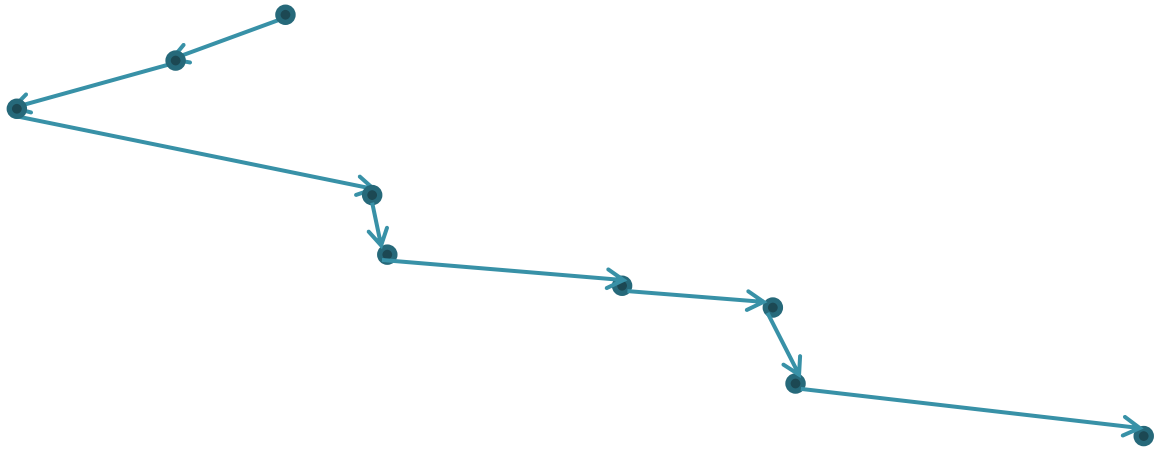
9.4 Raspoređivanje zahteva za rad sa diskom

Algoritam LOOK

početna pozicija glave: 53

red čekanja : 98, 183, 37, 122, 14, 124, 65, 67





A	B	P
53	37	16
37	14	23
14	65	51
65	67	2
67	98	31
98	122	24
122	124	2
124	183	59
ukupno		208

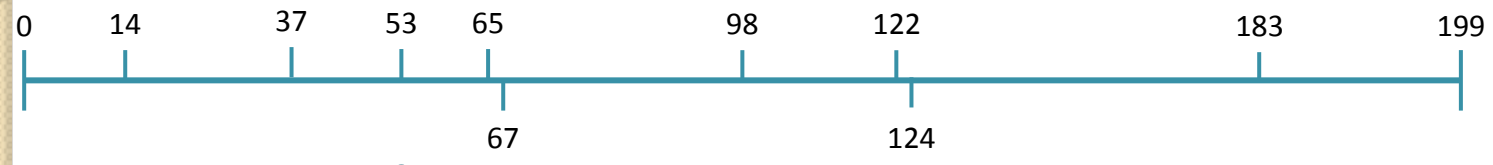
A	početna pozicija
B	ciljna pozicija
P	broj pomeraja

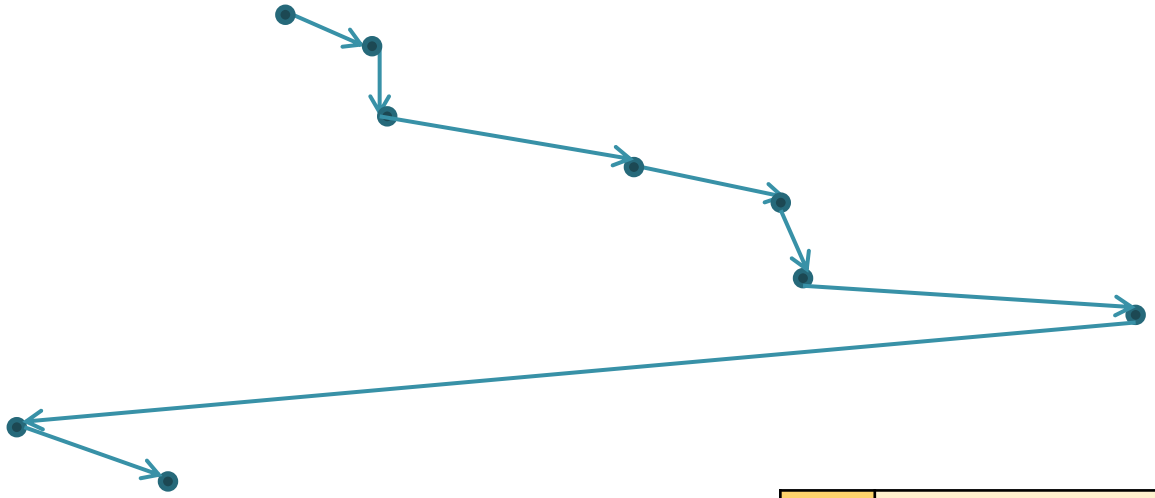
9.4 Raspoređivanje zahteva za rad sa diskom

Algoritam C-LOOK

početna pozicija glave: 53

red čekanja : 98, 183, 37, 122, 14, 124, 65, 67





A	B	P
53	65	12
65	67	2
67	98	31
98	122	24
122	124	2
124	183	59
183	14	169
14	37	23
ukupno		322

A	početna pozicija
B	ciljna pozicija
P	broj pomeraja

9.4 Raspoređivanje zahteva za rad sa diskom

Izbor najboljeg algoritma

- svi algoritmi su bolji od FCFS, ali je teško odrediti koji je od njih najbolji
- performanse samih algoritama **zavise od prispelih zahteva za rad sa diskom**
  - kružne varijante SCAN i LOOK algoritama imaju mnogo bolju raspodelu opsluživanja i nemaju problem zakucavanja (*starvation)*
  - C-LOOK je najbolje rešenje za jako opterećene sisteme
- modernije varijate ovih algoritama **minimiziraju i pozicioniranje i rotaciono kašnjenje**
  - rotaciono kašnjenje ima dominantan uticaj na performanse savremenih diskova
  - jedan takav algoritam je **SATF** (*Shortest Access Time First)*
  - radi na principu SSTF algoritma ali pri odabiru sledećeg zahteva iz reda računa obe mehaničke komponente

9.4 Raspoređivanje zahteva za rad sa diskom

Izbor najboljeg algoritma

- najsavremeniji algoritmi uzimaju u obzir i keširanje na samom disk uređaju
  - C-LOOK u kombinaciji sa ugrađenim disk keširanjem daje najbolje rezultate, što potvrđuju brojne simulacije iz otvorene literature

9.5 RAID strukture - realizacija stabilnih sistema

- upis na disk može da se završi na 3 načina
  - **uspešno okončan upis**
    - svi sektori su uspešno upisani na disk
  - **delimični otkaz**
    - otkaz je nastupio u sredini transfera
    - neki su sektori dobro upisani, a neki su oštećeni
  - **potpuni otkaz**
    - ništa nije upisano jer je ciklus upisa odmah otkazao
- sistem se može oporaviti od otkaza korišćenjem
  - tehnike vođenja dnevnika transakcija (*journalling)*
  - RAID (*Redundant Array of Independent Disks*) strukture koje koriste princip ogledala i parnosti
    - tehnika organizacije sekundarne memorije koja upotrebom više diskova umesto jednog ostvaruje povećanje pouzdanosti i poboljšanje performansi

9.5 RAID strukture - realizacija stabilnih sistema

Osnovne karakteristike RAID sistema

- **paralelizam i konkurentnost operacija**
  - tehnika deljenja podataka između različitih diskova
  - minimalna jedinica podataka koja se može kontinuelno smestiti na jedan disk jeste **traka** (*stripe unit*)
  - paralelno izvršenje više nezavisnih disk operacija u istom trenutku (konkurentnost operacija)
- **povećanje pozdanosti uvođenjem redundanse**
  - svaki disk ima svoju pouzdanost koja se meri srednjim vremenom otkaza
  - verovatnoća otkaza RAID strukture koju čini N diskova uvećava se N puta
  - problem pouzdanosti se rešava uvođenjem redundanse
    - čuvaju se dodatne informacije koje obezbeđuju mogućnost potpunog povratka podataka u slučaju otkaza jednog diska
  - tehnike
    - **ogledala** (*mirroring)*
    - **parnosti** (*parity)*

9.5 RAID strukture - realizacija stabilnih sistema

RAID nivoi

- deljenje podataka na trake poboljšava performanse, ali smanjuje pouzdanost
- tehnika ogledala povećava pouzdanost, ali smanjuje iskorišćenost i performanse upisa
- uvođenje parnosti je dobar kompromis
- postoji šest osnovnih RAID nivoa koji se razlikuju po performansama, pouzdanosti i ceni
- **RAID 0** (*striped set without parity*)
- RAID kontroler deli podatke na blokove koji su ispisani u vidu traka (*stripes*) na više diskova bez korišćenja parnosti
  - podaci će biti brže upisani jer hard disk upisuje manje datoteke
  - čitanje je takođe brže jer se podaci preuzimaju u nizu sa hard diskova a zatim sklapaju u kontroleru
  - brza konfiguracija ali bez ikakve redundanse, tako da otkaz bilo kog diska znači gubitak svih podataka na njemu

9.5 RAID strukture - realizacija stabilnih sistema

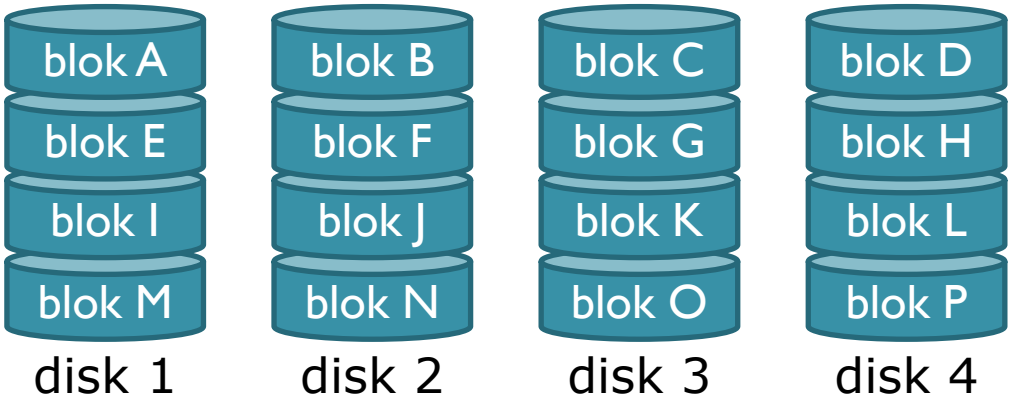
RAID nivoi

- **RAID 0** (*striped set without parity*) (*nastavak*)
- prednosti
  - odlične performanse čitanja i pisanja
  - ne gubi se vreme na proveru parnosti
  - lako i jeftino za implementaciju
- mane
  - nema tolerancije grešaka
  - ne može se koristiti kod sistema osetljivih na grešku
- preporučena upotreba
  - za sve sisteme koji zahtevaju veće performanse a bezbednost osguravaju redovnim uzimanjem bekapa
    - grafički dizajn
    - video obrada...

9.5 RAID strukture - realizacija stabilnih sistema

RAID nivoi

- **RAID 0** (*striped set without parity*) (*nastavak*)



9.5 RAID strukture - realizacija stabilnih sistema

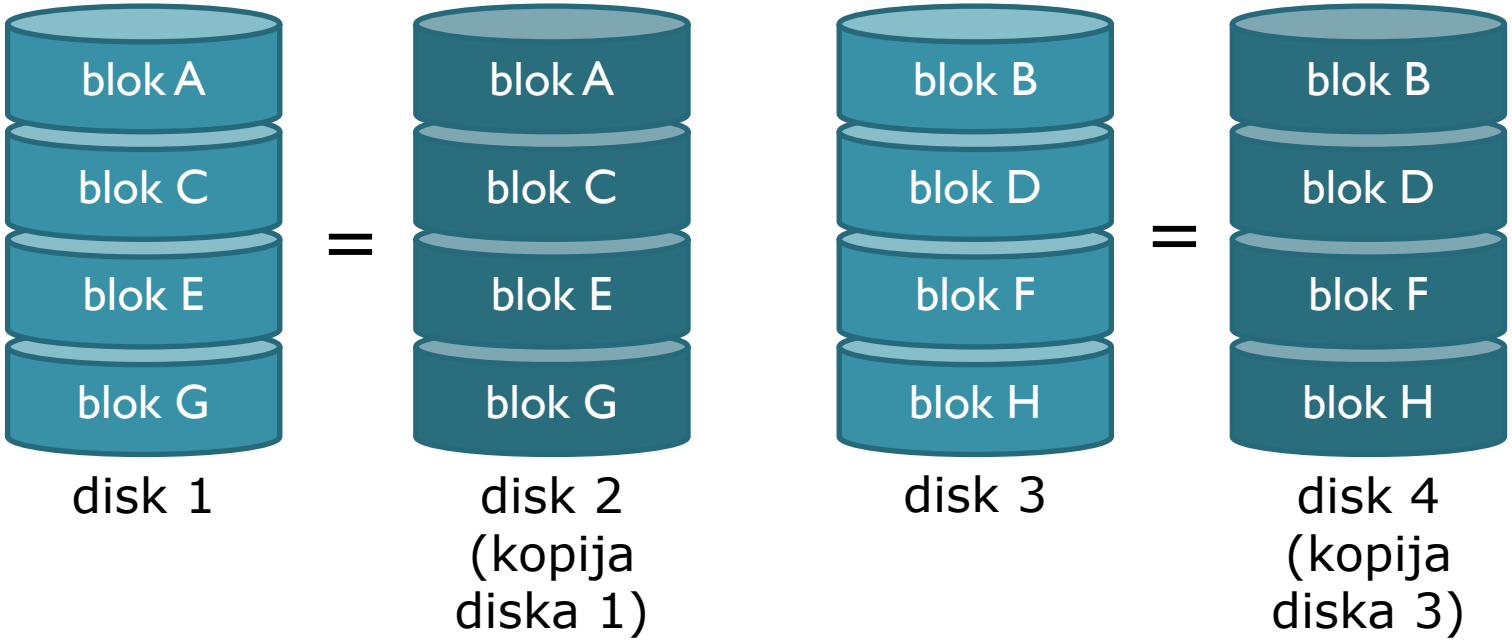
RAID nivoi

- **RAID 1** (*mirroring*)
- svaki disk ima svoju identičnu kopiju (ogledalo) na drugom disku
- prednosti
  - visoka tolerancija grešaka
  - u slučaju greške, podaci se ne rekonstruišu već samo kopiraju
  - ne gubi se vreme na proveru parnosti
- mane
  - neiskorišćeni kapaciteti diskova (50%) zbog dupliranja podataka
  - dupliran broj upisa
- preporučena upotreba
  - za sve sisteme koji zahtevaju bezbednost podataka na prvom mestu
    - veb serveri
    - bankarski sistemi...

9.5 RAID strukture - realizacija stabilnih sistema

RAID nivoi

- **RAID 1** (*mirroring*) *(nastavak)*



9.5 RAID strukture - realizacija stabilnih sistema

RAID nivoi

- **RAID 2** (*memory-style error correcting code organization*)
- poznat je pod nazivom RAID sa memorijskim stilom korekcije
- memorije imaju ECC (*Error Correcting Code*) algoritam koji za svaki bajt ima 3 ekstra bita, potrebna za detekciju i korekciju jednobitnih grešaka
- RAID 2 ima organizaciju deljenja podataka na bit (*bit-striping*) ili bajt nivou
- bez obzira na broj diskova podataka, potrebna su još tri diska za ECC koja mogu sačuvati podatke u slučaju otkaza bilo kog diska
- RAID 2 je dobar po pitanju paralelizma
- bolji je od RAID 0 po pitanju utroška diskova
- praktično se ne koristi, komercijalno neisplativ

9.5 RAID strukture - realizacija stabilnih sistema

RAID nivoi

- **RAID 3**
- deljenje podataka na nivou bita ili bajta
- po pitanju redundanse prvi put se uvodi parnost za diskove (*bit-interleaved parity organization*)
- ukoliko dođe do otkaza jednog diska svi podaci su i dalje dostupni
- za razliku od radne memorije u kojoj je teško odrediti tačnu poziciju greške, kod diskova svaki pojedinačni disk-kontroler zna da li je njegov pročitani sektor korektan ili ne
- zbog toga je bit parnosti (*parity bit*) dovoljan i za detekciju i za korekciju greške, pošto se tačno zna koji bit je pogrešan

9.5 RAID strukture - realizacija stabilnih sistema

RAID nivoi

- **RAID 3** (*nastavak*)
- primer primene bita parnosti
  - koristi se XOR, odnosno isključiva disjunkcija
  - ako je suma bitova svih diskova parna, bit parnosti dobija vrednost 0, a u suprotnom vrednost 1

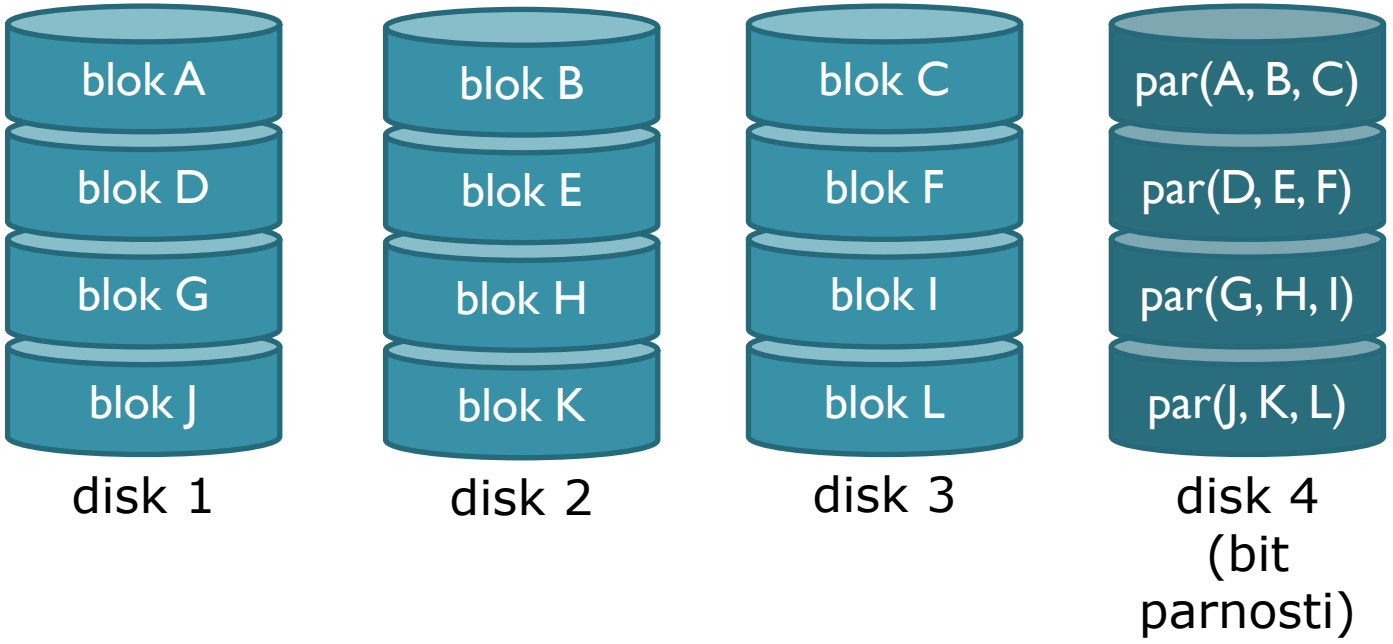
	disk 1	disk 2	disk 3	disk 4	bit parnosti
vrednost bita	0	1	1	1	1

  - zbir vrednosti bitova diskova 1 do 4 je 3, pa je bit parnosti 1
  - ukoliko se na nekom od diskova ne može, usled greške, izvršiti čitanje, njegov sadržaj se lako može rekonstruisati sabiranjem vrednosti bitova ostalih diskova i bita parnosti
    - ukoliko se ne može učitati sadržaj diska 3 na primer, zbir diskova 1, 2, 4 i bita parnosti je neparan broj (3), te se na osnovu toga može zaključiti da je vrednost na disku 3 bila 1

9.5 RAID strukture - realizacija stabilnih sistema

RAID nivoi

- **RAID 3** (*nastavak*)



- njegova primena zahteva najmanje tri diska
  - broju diskova na koje se zapisuju podaci (najmanje dva) potrebno je pridodati i jedan posvećen zapisu bitova parnosti

9.5 RAID strukture - realizacija stabilnih sistema

RAID nivoi

- **RAID 4**
- deljenje podataka na nivou bloka (*block striping*)
- po pitanju redundanse koristi se parnost za diskove na nivou bloka (*block-interleaved parity organization*)
- čitanje i upis velikih segmenata obavlja se paralelizovano
- upis jednog bloka ili nezavisni upisi po jedan blok jesu problem
  - ne mogu se obavljati u paraleli jer svaki zahteva čitanje bloka, izmenu i upis, i to i za podatke i za blok parnosti
- za N blokova dovoljan je jedan blok parnosti koji je jednak veličini trake diska
- velika mana ovog rešenja je što dovodi do preopterećenja diska parnosti koji učestvuje u svakom ciklusu upisa pa tako postaje usko grlo

9.5 RAID strukture - realizacija stabilnih sistema

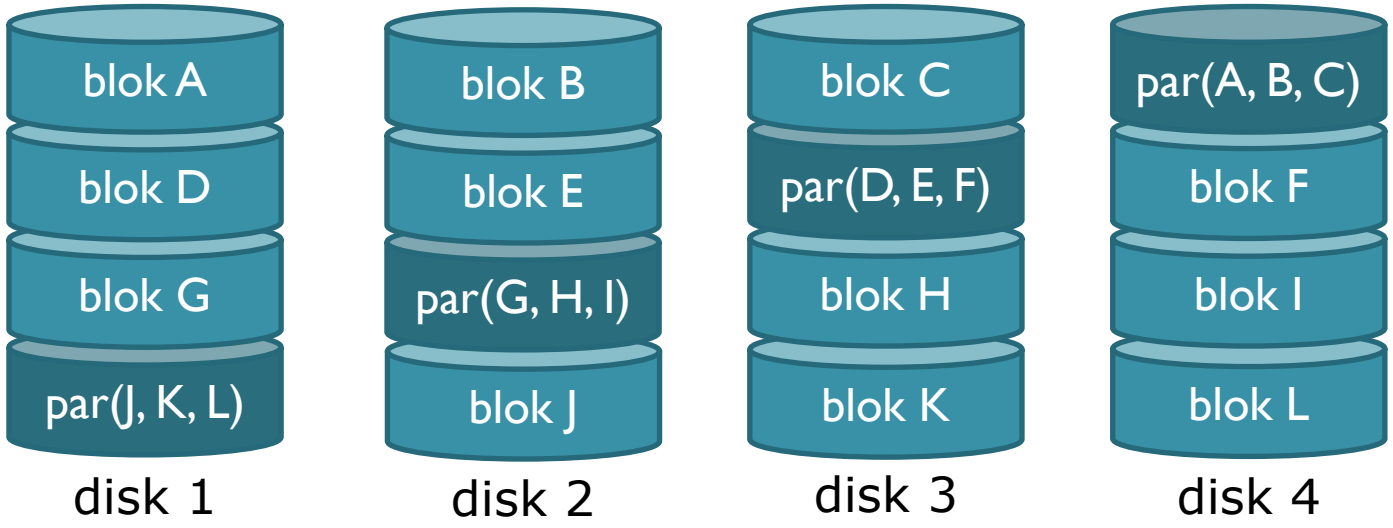
RAID nivoi

- **RAID 5**
- veoma je sličan RAID 4 nivou, samo što podatke i bitove parnosti rasipa po svim diskovima
  - za svaki paket od N blokova, jedan disk (bilo koji) čuva parnost, ostali podatke, ciklično (*block-interleaved distributed parity*)
  - parnost se upisuje u levom simetričnom rasporedu
- u ovom nizu uloga diskova se stalno smenjuje
  - time se eliminiše usporavanje do koga bi došlo da se informacija o parnosti konstantno upisuje na isti disk (RAID 4)
  - zbog toga je potreban hardverski kontroler, jer on ne samo da generiše informaciju o parnosti, već i određuje smenu diskova
- u slučaju otkaza bilo kog diska, niz i dalje može da nastavi da radi
  - mana je ta što će raditi sporije, a i samo rekreiranje niza, posle zamene pokvarenog diska, traje znatno duže nego kod RAID 1

9.5 RAID strukture - realizacija stabilnih sistema

RAID nivoi

- **RAID 5** (*nastavak*)
- najbolja kombinacija
  - poseduje paralelizam
  - konkurentnost
  - diskovi su ravnomerno raspoređeni



9.5 RAID strukture - realizacija stabilnih sistema

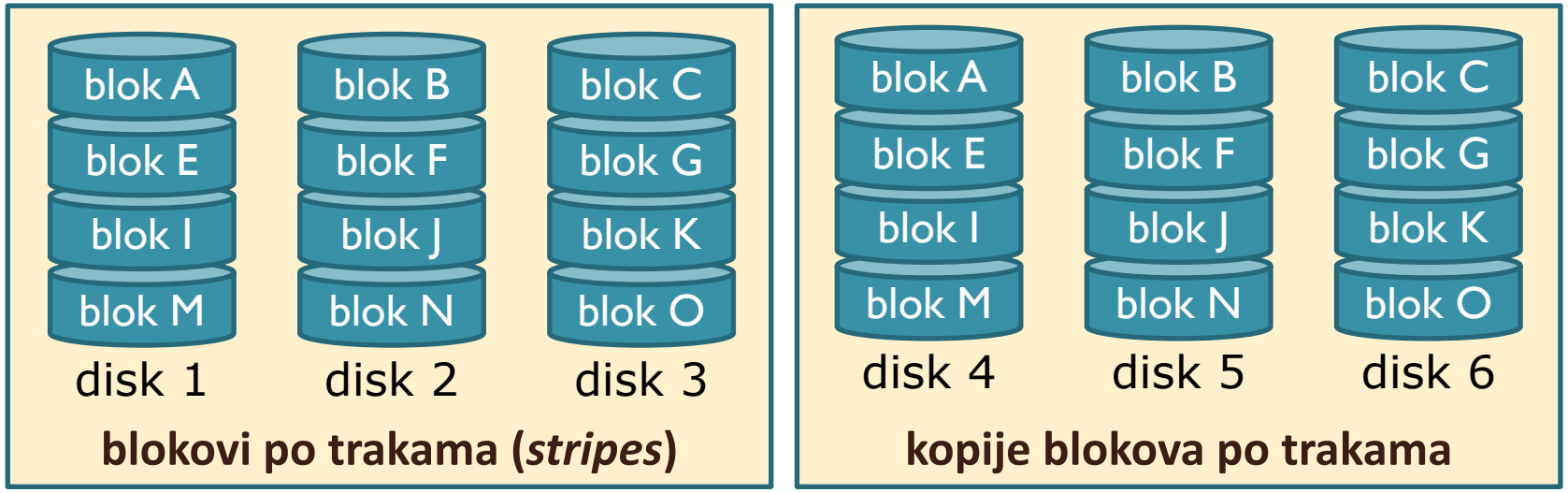
RAID nivoi

- **RAID 6**
- RAID 6 (*P+Q redundancy scheme*) predstavlja jedinu RAID kombinaciju koja može razrešiti problem u slučaju otkaza više diskova
- radi kao RAID 5, ali umesto parnosti, čuva dodatne redundantne ECC kodove kako bi se zaštitio od otkaza više od jednog diska
- rezultat ovog **dvostrukog pariteta** je svakako povećanje tolerancije, tako da RAID 6 može toleriše kvar bilo koja dva diska
- RAID 6 je malo lošiji od RAID 5 zbog izračunavanja i zapisivanja više paritetnih informacija, dok je kod čitanja podataka malo bolja situacija zbog čitanja podataka sa jednog dodatnog diska

9.5 RAID strukture - realizacija stabilnih sistema

RAID nivoi

- **RAID (0+1) i RAID (1+0)**
- kvalitetne kombinacije tehnike 0 i 1
  - RAID 0 obezbeđuje visoke performance
  - RAID 1 obezbeđuje visoku pouzdanost
  - kombinacija ostaje i dalje skupa jer udvostručava broj diskova
- kombinacija 0+1:
  - skup diskova deli podatke
  - potom se sve *stripe* jedinice u celini kopiraju u svoje ogledalo



9.5 RAID strukture - realizacija stabilnih sistema

RAID nivoi

- **RAID (0+1) i RAID (1+0)** (*nastavak*)
- kombinacija 1+0:
  - svaki disk ima svoje ogledalo
  - podaci se dele između ogledala