

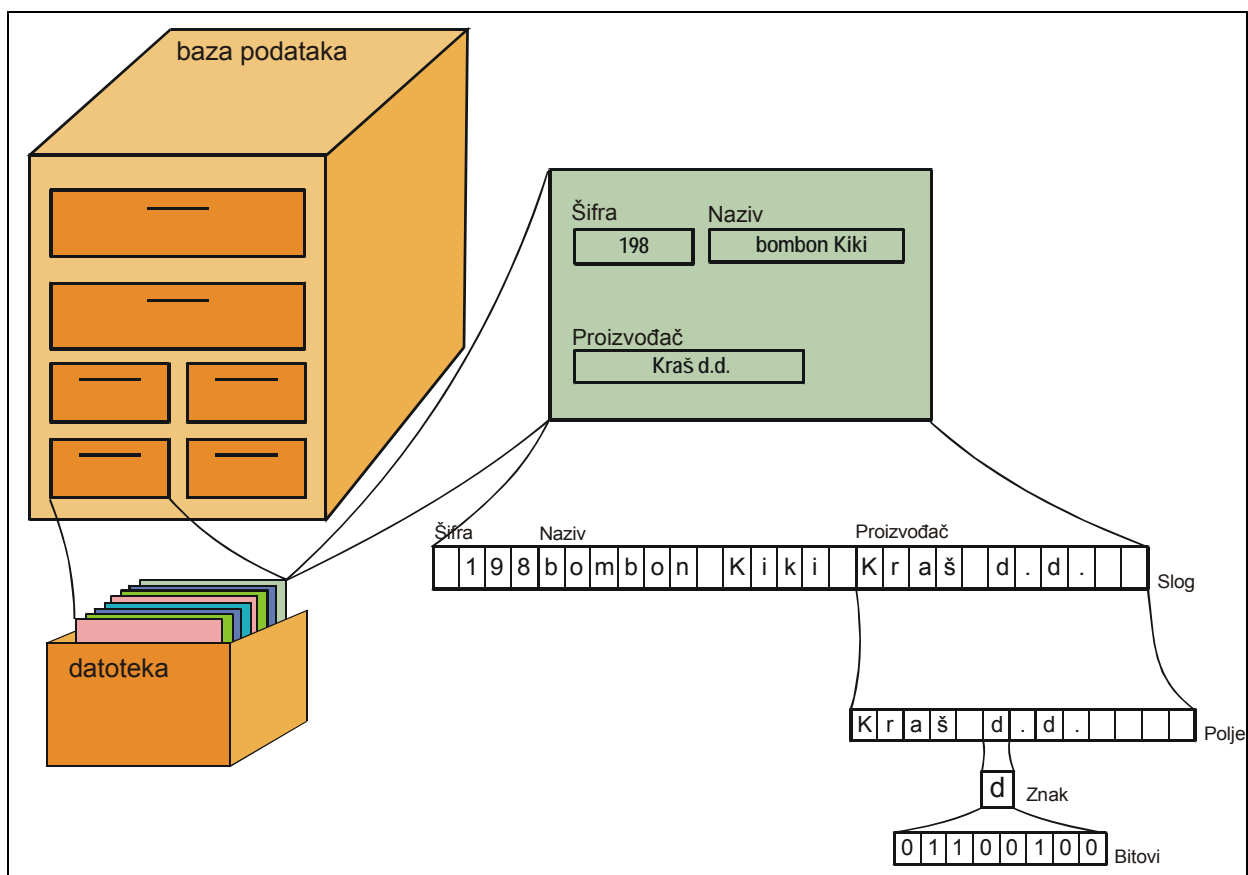
## VI. ORGANIZACIJA I UPRAVLJANJE PODACIMA U RAČUNARU

### 1. ORGANIZACIJA PODATAKA U RAČUNARU

Da bi se podaci mogli efikasno pronalaziti i obrađivati, moraju se u računaru na prikladan način organizirano pohraniti na perifernoj, odnosno sekundarnoj memoriji računara.

Osnovne elemente organizacije podataka prikazuje slika 1.1. Vidi se da je organizacija podataka hijerarhijska. Najviši nivo predstavlja baza podataka, a niži elementi su:

- datoteteka,
- slog,
- polje,
- znak i
- bit, kao dio zapisa znaka u reprezentaciji koju razumije računar.



Sl. 1.1 Hijerarhijska organizacija podataka u bazi podataka

Skup trajno pohranjenih podataka informacionog sistema naziva se *bazom podataka*.

**Baza podataka** (engl. *data base*) je skup povezanih raznovrsnih podataka nekog informacionog sistema, odnosno podataka nekog područja, koje iziskuje prikupljanje odgovarajućih podataka.

Tako baza podataka odjela prodaje nekog poslovnog subjekta sadrži podatke o proizvodima, kupcima i dobavljačima, dakle o različitim objektima koje srećemo u poslovnim sustavima.

## 2. DATOTEKA

### 2.1 Šta je to datoteka?

Datoteka (engl. *file*) je najstariji i najjednostavniji oblik organiziranja i pohranjivanja podataka u računaru. Kod opsežnijih aplikacija, ili kod informacionih sistema koristimo se bazama podataka, u kojima je datoteka osnovni element za pohranjivanje podataka.

Datoteka se sastoji od skupa istovrsnih podataka – *slogova* ili *zapisa*, koji se karakterišu nekim zajedničkim kriterijem i koji su smješteni u memoriji računara. Jedan slog sadrži podatke jednog objekta, na primjer proizvoda, studenta, fakture itd. Stoga se može govoriti o datoteci proizvoda, studenata, faktura itd.

Datoteka se može uporediti s kartotekom, kao na slici 1.1. Ovdje je slog jedna kartica kartoteke.

#### 2.1.1 Osnovni elementi strukture datoteke

##### Polje podatka (Data field)

Individualni elementi opisa nekog događaja se nazivaju *stavke podataka* (data items), ili kratko stavke, a kada je u pitanju digitalna struktura kod registrovanja podataka u memoriji računara, poznati su kao *polja podataka* (data fields).

Svako polje podatka se identifikuje imenom (nazivom), vrstom i veličinom, na način kako je to pokazano slijedećoj tabeli 2.1:

Tabela 2.1 Primjer definisanja polja podataka

Naziv polja	Vrsta ili tip	Veličina polja u znakovima
Šifra materijala	alfa-numeričko (AN)	6
Naziv materijala	alfabetsko (A)	20
Jedinica mjere	numeričko (N)	2
Nabavljena količina	numeričko (N)	10
Cijena	numeričko (N)	6,2 (šest cijelih i dvije decimale)
Dobavljač	numeričko (N)	4

Naziv i veličinu polja određuje korisnik, dok vrsta polja proizilazi iz karaktera podatka koje se “smješta” u određeno polje.

Definisanju elemenata podataka i opisu njihovih polja treba posvetiti posebnu pažnju, jer su to osnovni elementi kojima se određuju strukture datoteka, kao dijelova baza podataka.

Osim toga, definisanjem polja podataka omogućava se i programska kontrola unosa podataka.

### Segment sloga

Više polja podataka, koje karakteriše neki zajednički imenitelj, čini segment sloga. Segment je dakle dio sloga. Tako se, na primjer, slog kupca sastoji od slijedećih segmenata i njihovih polja;

segment 1: adresa kupca, s poljima: država, grad, ulica, broj;

segment 2: uslovi plaćanja, s poljima: rok, način, popust, kamate;

segment 3: stanje računa, s poljima: dužuje, potražuje, saldo.

Segment nije fizička jedinica podataka u organizaciji memorije, ne izdvaja se u posebnu fizičku cjelinu i ne može mu se direktno pristupiti kao cjelini. Segment je dio cjeline koja se naziva slog datoteke.

### Slog podataka (Record)

Kompletan set podataka, koji su vezani za neki objekat, odnosno entitet, kojeg u cjelosti definišu, a koji se pohranjuju i kasnije obrađuju, naziva se slog (record). Na primjer, slog radnika, slog kupca, slog artikla, slog bankarskog čeka, slog osnovnog sredstva, slog proizvoda itd.

Slog se sastoji od određenog broja polja i segmenata, kao skupova određenih polja.

U slijedećoj tabeli je naveden primjer sloga radnika, koji se sastoji od određenih podataka (polja).

**Tabela 2.2 Primjer sloga radnika**

Naziv stavke	Vrsta ili tip	Veličina polja u znakovima
Redni broj u datoteci	numeričko (N)	4
Matični broj	numeričko (N)	13
Ime, ime oca, prezime	alfabetsko (A)	30
Mjesto rođenja	A	12
Datum rođenja (XX/XX/XXXX)	N	8
Mjesto stanovanja	A	12
Ulica, naselje	A	15
Broj	N	3
Bračno stanje	A	1
Broj djece	N	1
Šifra radnog mjesta	N	2
Naziv radnog mjesta	A	10
Ukupna dužina sloga		111

## 2.2 Vrste datoteka s obzirom na njihovu organizaciju

Datoteke se dijele prema metodi adresiranja slogova. Adresiranjem se određuje mjesto, tj. *adresa* sloga u datoteci. Dakle, kriterij podjele datoteka je način fizičkog organiziranja slogova. Osnovne vrste datoteka su:

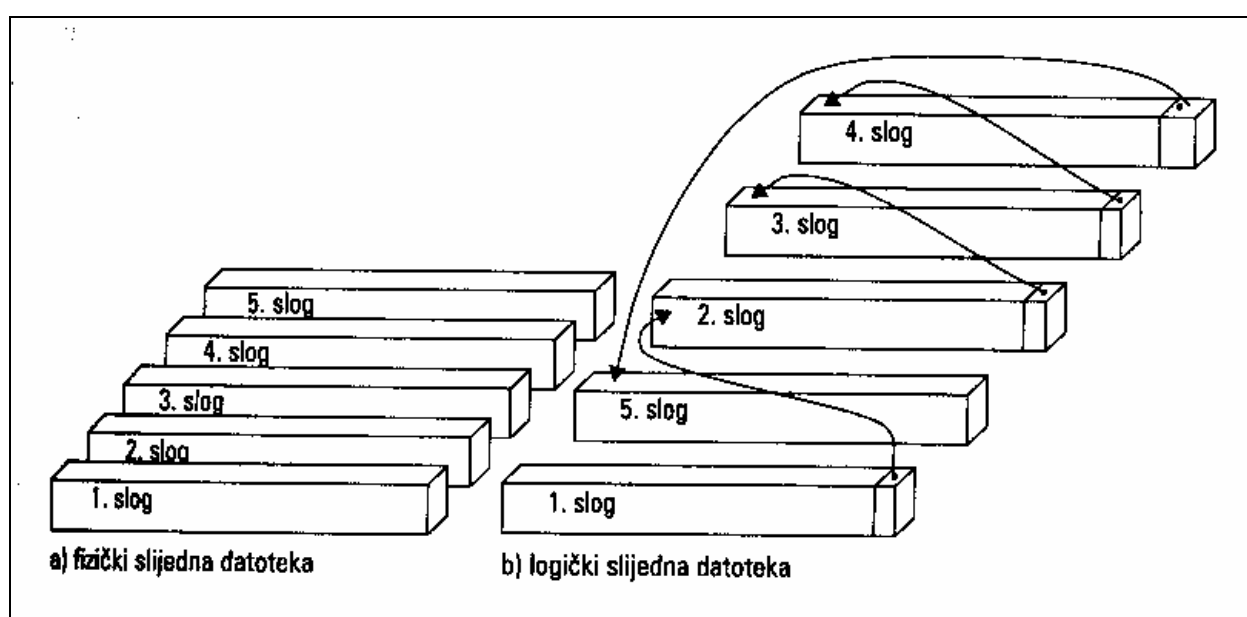
- slijedne, odnosno sekvencijalne datoteke,
- direktne, odnosno relativne datoteke i
- indeksne datoteke.

### 2.2.1 Slijedne ili sekvencijalne datoteke

U slijednim, odnosno sekvencijalnim datotekama slogovi se upisuju i memorišu jedan do drugoga, redoslijedom dolaska. Slijedeći slog se upisuje na adresu neposredno iza prethodnog sloga. Kod obrade se slogovi obrađuju istim redoslijedom

Na slici 2.1a prikazana je slijedna datoteka kod koje su slogovi međusobno *fizički* slijedno povezani. Slogovi su smješteni na susjednim adresama.

Na slici 2.1b prikazana je slijedna datoteka, čiji su slogovi povezani *logički*. Pri tome slogovi ne moraju biti smješteni na susjednim adresama. Povezanost se ostvaruje nizom *pokazivača* (engl. *pointer*), pomoćnih podataka u slogu koji pokazuju adresu slijedećeg sloga u nizu. Niz pokazivača na slici je prikazan strelicama.



Sl. 2.1 Fizički ili logički organizovana slijedna datoteka

### 2.2.2 Direktno ili relativne datoteke

U direktnoj, odnosno relativnoj datoteci (engl. *direct, relative, hash file*) slogovi se pohranjuju na relativnim adresama, počevši od relativne adrese 1 do adrese n. Pri tome se slog s ključem 1 memoriše na relativnoj adresi 1, slog s ključem 2 na adresi 2 itd. Pronalaženje slogova prilikom čitanja datoteke provodi se na isti način, tako da će se na primjer na adresi 25 naći slog s ključem 25 itd. Dakle, postoji čvrst odnos ključa i relativne adrese sloga, pri čemu vrijedi pravilo da je  $adresa = ključ$ .

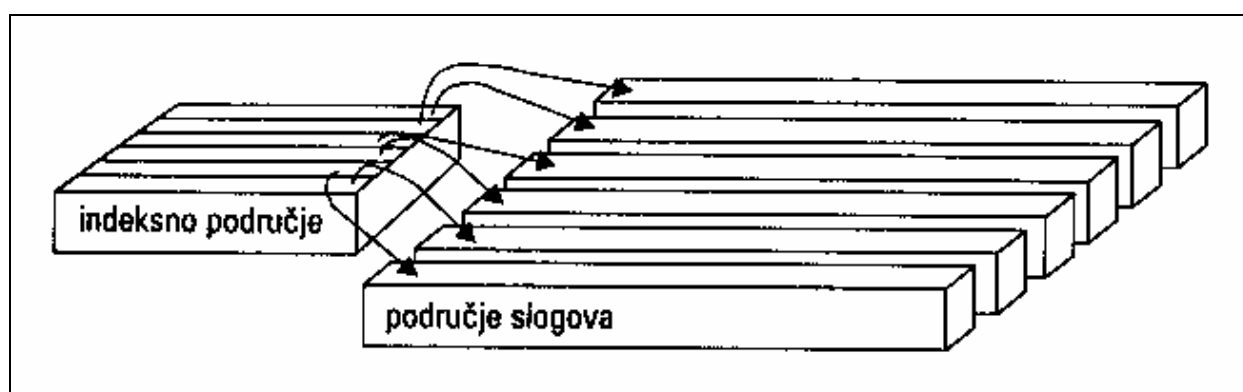
U praksi se rijetko ima situacija da vrijednosti ključeva počinju od 1 i čija je razdioba pravilna. Zbog toga se ključ transformira u adresu pogodnom matematičkom operacijom, tako što je  $adresa = f(ključ)$ .

Najčešće se za transformaciju ključa u adresu (engl. *randomizing function*) koristi operacija dijeljenja s ostatkom (tzv. modularno dijeljenje). Za primjer će se načiniti relativna datoteka za slogove 2200 osoba, tako da svaki slog sadrži podatke jedne osobe. Ako se očekuje da će broj osoba u evidenciji rasti, uzet će se mjesta za na primjer 2500 osoba. To znači da će adrese slogova biti u rasponu od 1 – 2500. Za ključ će se izabrat jedinstveni matični broj građana (JMBG), koji se sastoji od 13 cifara. Vrijednost JMBG će se transformirati u raspon od 1 – 2500, ako se podijeli s kapacitetom datoteke, a to je 2500. Ostatku dijeljenja, koji može biti od 0 – 2499, dodat će se 1, te tako dobiti vrijednosti adrese od 1 – 2500. Za vrijednost JMBG 2508971335285 koju dijelimo s 2500 dobit će se 285 kao ostatak dijeljenja. Prema tome, adresa ovog sloga je 286.

Metodom transformacije ključa u adresu omogućen je direktan pristup svakom slogu, pa se datoteka naziva i *direktnom*. Medij na koji se memoriše datoteka mora imati mogućnost direktnog pristupa, a takav medij je tvrdi disk.

### 2.2.3 Indeksne datoteke

Indeksna datoteka se sastoji od *indeksnog područja* i *područja slogova*, kao što je prikazano na slici 2.2.



Sl. 2.2 Dijelovi indeksne datoteke: indeksno područje i područje slogova

U indeksnom području smješten je *indeks*, sličan indeksu koji srećemo u stručnim knjigama. Uz svaki ključ u indeksu nalazi se adresa njegovog sloga u području podataka.

Pronalaženje slogova uvijek počinje od indeksa, pronalaženjem adrese sloga, a nastavlja se u području slogova pronalaženjem sloga na dobivenoj adresi. Zato se indeksno područje organizuje u obliku datoteke, u kojoj se pronalaženje indeksnih slogova odvija brzo. Za ovo je pogodna direktna datoteka, pa se indeksno područje može organizovati u obliku jedne ovakve datotetke.

Područje slogova obično se organizira kao slijedna datoteka, u kojoj se slogovi nalaze jedan do drugoga, čime se štedi memorijski prostor.

Indeksne datoteke imaju još jednu prednost, koja se ogleda u mogućnosti formiranja više indeksa za više različitih ključeva. Na primjer, ako indeksna datoteka ima slogove o osobama, jednom će biti potrebno brzo pronalaziti osobe po matičnom broju građanina, a drugi put po prezimenu. Problem se rješava formiranjem dvaju samostalnih indeksa, jednog po matičnom broju i drugog po prezimenu.

#### 2.2.4 Indeksne slijedne datoteke

Osim prethodno opisanih osnovnih oblika datoteka, moguće su i njihove različite kombinacije. Među njima je najpoznatija i najčešće korištena indeksna slijedna datoteka. Njene prednosti su što omogućava:

- direktan pristup i pojedinačnu obradu slogova (na primjer čitanje, ili promjene pojedinačnih slogova), jer ima karakteristike indeksne datoteke i
- skupnu obradu svih ili veće skupine slogova, jer ima karakteristike slijedne datoteke.

Kao i indeksna datoteka, i ova datoteka ima dva područja: indeksno područje i područje podataka.

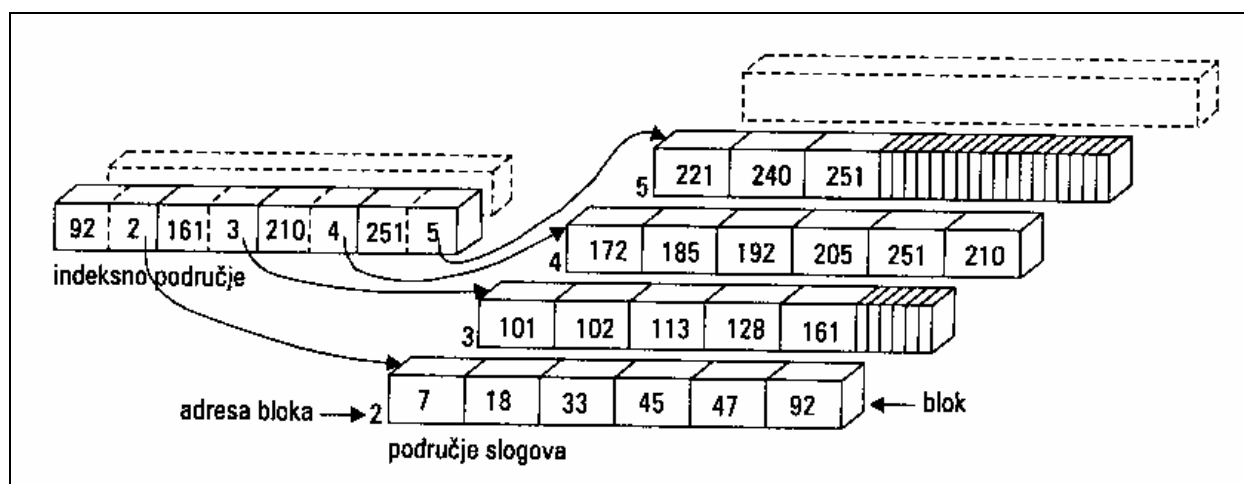
Područje podataka je organizirano kao uređena (sortirana) slijedna datoteka, koja se sastoji od blokova u koje je moguće smjestiti više slogova.

Za svaki se blok bilježi jedna indeksna vrijednost u indeksu, obično vrijednost najvećeg, odnosno posljednjeg sloga u bloku, kao što je prikazano na slici 2.3.

Kod ove datoteke indeks je manji, jer je broj indeksnih vrijednosti manji od broja slogova u datoteci, odnosno jednak je broju blokova.

Kod datoteka kod kojih je broj indeksnih slogova manji od broja slogova u datoteci, indeksno područje mora biti organizirano kao uređena slijedna datoteka i ne može biti organizirana kao direktna datoteka (kao kod indeksne datoteke).

Slogovi na slici 2.3 su označeni vrijednostima svojih ključeva.

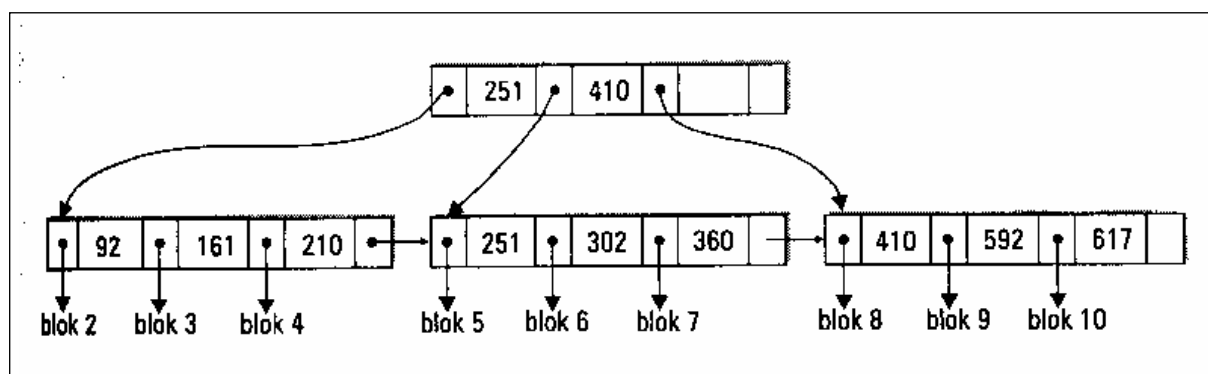


Sl. 2.3 Indeksna slijedna datoteka kao uspješna kombinacija indeksne i slijedne datoteke

Za pojedinačno pronalaženje pojedinih slogova koristi se indeks, od kojeg počinje pretraživanje. Tako prvi indeksni slog pokazuje da se u prvom bloku područja podataka na adresi 1 nalaze svi slogovi do ključa 92, drugi indeksni slog pokazuje da se u drugom bloku područja podataka na adresi 3 nalaze slogovi od ključa većeg od 92 do ključa 161 itd. Traži li se na primjer slog s ključem 205, indeks upućuje na to da se slog s ovim ključem nalazi u trećem bloku na adresi 4. Nakon toga se slijednim pretraživanjem trećeg bloka pronalazi traženi slog.

Za čitanje svih slogova, ili velike skupine slogova, koristi se samo područje podataka, koje se čita kao slijedna datoteka, od prvog sloga u prvom bloku do posljednjeg sloga u posljednjem bloku područja podataka.

Kako je indeksno područje organizirano kao slijedna datoteka, pretraživanje velikih datoteka s puno indeksnih slogova je dugotrajno. U tom se slučaju može organizirati indeks indeksnog područja, tj. indeks s više razina, što ubrzava pronalaženje ključeva u indeksnom području.



Sl. 2.4 Indeks u obliku B-stabla s dvije razine

Traženje po ključu počinje od korijena. Ukoliko je ključ traženog sloga manji od prvog ključa u korijenu, pretraživanje se nastavlja u prvom čvoru na nižoj razini. Ako je ključ traženog sloga jednak ili veći od



vrijednosti prvog ključa i manji od vrijednosti drugog ključa, pretraživanje se nastavlja u drugom čvoru iste razine itd.

### **2.2.5 Usporedba datoteka**

Usporedba svih navedenih tipova pokazuje da ne postoji superiorna i univerzalna vrsta datoteka.