

OSNOVNE KARAKTERISTIKE INDEKSNE ORGANIZACIJE

U radu su s više jednostavnih primjera ilustrirane karakteristike i mogućnosti indeksno organiziranih datoteka. Prikazan je ilustrativni primjer pomoću kojeg se pokazuje prednost koju indeksna organizacija ima pred sekvencijalnom u slučaju prekoredne obrade, pri čemu je upotrijebljen najjednostavniji oblik indeksne organizacije podataka u kojem indeks predstavlja slijedni niz, odnosno skup uređenih parova sortiranih prema ključu, a osnovni skup logičkih slogova može biti sortiran. Preuzimanje traženog logičkog sloga izvedeno je korištenjem tehnike slijednog pretraživanja indeksa. Matematičkim putem dokazano je da vrijeme potrebno za preuzimanje sloga iz osnovnog skupa stoji u odnosu 0,255 : 10 sec. u korist indeksne organizacije. U daljnjem tekstu rad je orijentiran domeni korištenja indeksno organiziranih datoteka u fazama kreiranja, ažuriranja te dodavanja i isključivanja jednog ili skupine slogova iz osnovnog skupa i indeksa. Također je obrađena kategorija logički slijedne obrade.

1.

Kod indeksnih organizacija relacija, kojom je definiran odnos između ključa i adrese logičkog sloga na memoriji, ima oblik skupa uređenih parova. Skup uređenih parova, ili tzv. indeks, pohranjen je na memoriji manje ili više neovisno o osnovnom skupu. Indeks možemo shvatiti kao pomoćnu datoteku koja se koristi da bi se ubrzalo prekoredno preuzimanje logičkih slogova iz osnovnog skupa. Umjesto da se pretražuje osnovni skup, kao u slučaju sekvencijalno organizirane datoteke, do adrese na kojoj se nalazi traženi slog dolazimo pretraživanjem indeksa (Tkalec, 1979.).

Sekvencijalna organizacija omogućava i pogoduje uglavnom slijednoj obradi podataka. To je uvjetovano karakteristikama osnovnog skupa logičkih slogova koji su međusobno povezani samo fizičkim vezama. Da bi se s memorije preuzeo jedan određeni logički slog, potrebno je izvršiti pretraživanje datoteke.

Indeksno organizirana datoteka omogućava, pored slijedne, i prekorednu obradu logičkih slogova. U najjednostavnijem slučaju, za svaki logički slog u osnovnom skupu, u indeksu postoji uređeni par: KLJUČ, ADRESA (u daljnjem tekstu: K, A). Pretraživanjem indeksa moguće je ustanoviti adresu na kojoj se nalazi logički slog sa zadanim ključem.

2.

Slijedećim primjerom pokazat ćemo prednost koju indeksna organizacija ima pred sekvencijalnom u slučaju prekoredne obrade.

Neka je skup od 1000 logičkih slogova organiziran u prvom slučaju kao sekvencijalna, a u drugom kao indeksna datoteka, s tim da je redoslijed logičkih slogova u oba slučaja identičan. Neka u indeksu indeksno organizirane datoteke za svaki logički slog u osnovnom skupu postoji uređeni par K,A i neka se u oba slučaja koristi metoda slijednog pretraživanja. Konačno, neka su zapisi u indeksu blokirani (faktor blokiranja = 300). Ako pretpostavimo da je slog koji tražimo zapisan kao 400. slog u datoteci, nije teško izračunati vrijeme potrebno da bi se i jednom i drugom slučaju pronašao i preuzeo traženi slog iz datoteke.

U slučaju sekvencijalno organizirane datoteke bit će zbog slijednog pretraživanja potrebno iz datoteke preuzeti i ispitati sadržaj svih slogova koji prethode traženom slogu. Tek nakon toga doći ćemo do traženog sloga. Vrijeme potrebno za ispitivanje zadržaja možemo zanemariti. Vrijeme potrebno za preuzimanje jednog sloga iz datoteke jednako je vremenu potrebnom za jedan okret diska. Pretpostavimo da je to vrijeme 25_{ms} . Ukupno vrijeme potrebno za pronalaženje i preuzimanje traženog sloga s diska bit će jednako

$$T_s = am$$

gdje je »a« vrijeme okreta diska, a »n« redni broj traženog sloga.

U našem primjeru to će biti

$$T_s = 25 \times 400 = 10\,000_{ms}$$

odnosno

$$10 \text{ sec.}$$

Ukoliko datoteka ima indeksnu organizaciju, do adrese na kojoj se traženi slog nalazi u osnovnom skupu dolazimo slijednim pretraživanjem indeksa. Za to će biti potrebno iz indeksa preuzeti i u memoriji pretražiti prva dva bloka, ako pretpostavimo da je osnovni skup sortiran, odnosno da je uređeni par K,A koji odgovara 400. slogu u osnovnom skupu, 400. uređeni par u indeksu. I ovdje možemo zanemariti vrijeme potrebno za pretraživanje blokova.

Ukupno vrijeme potrebno za pronalaženje i preuzimanje određenog logičkog sloga iz indeksne datoteke sastoji se iz vremena potrebnog za obradu indeksa (T_i) i vremena potrebnog za preuzimanje sloga iz osnovnog skupa (T_o):

$$T_u = T_i + T_o$$

Vrijeme potrebno za preuzimanje jednog bloka iz indeksa jednako je u principu vremenu potrebnom za preuzimanje jednog sloga iz sekvencijalno organizirane datoteke. Međutim, budući da je broj blokova koje treba preuzeti malen, ne smijemo zanemariti vrijeme potrebno da bi se uređaj za pisanje i čitanje diska postavio na početak indeksa (75_{ms}):

$$T_i = t_p + am$$

$$T_i = 75 + 25 \times 2 = 125_{ms}$$

Ukupno vrijeme iznosi:

$$T_u = T_i + T_o$$

$$T_u = 125 + 100 = 225_{ms}$$

odnosno

$$0,255 \text{ sec.}$$

Dakle, u slučaju sekvencijalne organizacije ukupno vrijeme potrebno za preuzimanje traženog sloga iznosi 10 sec, dok u skučaju korištenja indeksne organizacije to isto vrijeme iznosi samo 0,255 sec.

U prikazanom primjeru upotrijebljen je najjednostavniji oblik indeksne organizacije podataka u kojem indeks predstavlja slijedni niz, odnosno skup uređenih parova sortiranih prema ključu, a osnovni skup logičkih slogova može biti sortiran (prema: Tkalac, 1979). Preuzimanje traženog logičkog sloga izvedeno je korištenjem tehnike slijednog pretraživanja indeksa. Slijedno pretraživanje pripada kategoriji jednostavnijih i relativno sporih tehnika pretraživanja. Ukoliko bi se umjesto navedenih koristili savršeniji oblici indeksne organizacije (npr. hijerarhijska struktura indeksa kombinirana s osnovnim skupom logičkih slogova koji je sortiran i podijeljen na podliste) i složenije tehnike pretraživanja (kao što je pretraživanje sa zadanim korakom ispitivanja ili tehnika binarnog pretraživanja), vrijeme potrebno za preuzimanje traženog logičkog sloga bilo bi još kraće.

3.

Indeksno organizirana datoteka sastoji se od osnovnog skupa logičkih slogova i pripadajućeg indeksa. U najjednostavnijem slučaju osnovni skup predstavlja niz logičkih slogova smještenih na memoriji, a indeks je skup uređenih parova (K,A) sortiranih prema ključu. Svakom logičkom slogu osnovnog skupa pripada po jedan uređeni par u indeksu. Elementi uređenog para jesu već ranije spomenuti ključ i adresa logičkog sloga (prema: Tkalac, 1979).

Postupak kreiranja ovako organizirane indeksne datoteke ne razlikuje se bitno od kreiranja sekvencijalno organizirane datoteke. Logički slogovi osnovnog skupa preuzimaju se redom, onako kako su obuhvaćeni u ulaznoj datoteci, i upisuju se na memoriju. Formiranje indeksa izvršava se paralelno s upisivanjem logičkih slogova osnovnog skupa. Rezultat postupka kreiranja indeksno organizirane datoteke je osnovni skup logičkih slogova koji može, ali ne mora, biti sortiran i indeks sortiran prema ključu.

Zbog jednostavnosti postupka, u primjeru kreiranja indeksno organizirane datoteke, logički slogovi bit će predstavljeni samo relevantnim podatkom, tj. ključem sloga, a indeks će se sastojati od skupa uređenih parova K,A. Osnovni skup neće biti sortiran, što znači da slogovi ne moraju biti upisani na memoriji na ona mjesta koja im prema ključu pripadaju.

U prvom koraku kreiranja uz osnovni skup logičkih slogova pojavljuje se i indeks koji još nije sortiran:

OSNOVNI SKUP		INDEKS	
R. Br	Ključ	Adresa	Ključ
1.	2468	01	2468
2.	3512	02	3512
3.	1084	03	1084
4.	0097	04	0097
5.	2680	05	2680

U drugom koraku kreiranja indeksno organizirane datoteke dolazi do sortiranja indeksa prema ključu, dok osnovni skup logičkih slogova ostaje nepromijenjen:

OSNOVNI SKUP		INDEKS	
R. Br	Ključ	Adresa	Ključ
1.	2468	04	0097
2.	3512	03	1084
3.	1084	01	2468
4.	0097	05	2680
5.	2680	02	3512

Pod slijednom obradom ove datoteke podrazumijevalo bi se da se slogovi preuzimaju i obrađuju iz osnovnog skupa onim redom kojim su zapisani na memoriji. Postojanje indeksa omogućava, pored vrlo uspješne prekorodne obrade, i tzv. logički slijedna obradu indeksno organizirane datoteke. Ovaj pojam podrazumijeva da se slogovi iz datoteke preuzimaju i obrađuju redom prema kojem su njihovi uređeni parovi upisani u indeks. Kako se kad je u pitanju indeksna datoteka samo izuzetno koristi slijedna obrada (kod preispitivanja datoteke, npr.) u praksi se zbog kratkoće koristi izraz »slijedna obrada« s tim da se pod tim izrazom u stvari podrazumijeva logički slijedna obrada.

4.

Kod indeksno organiziranih datoteka u kojima osnovni skup može, ali ne mora biti sortiran novi slogovi se upisuju na kraju niza. Proširenje osnovnog skupa logičkih slogova vrlo je jednostavno, njegovo održavanje također, a preuzimanje logičkih slogova uvjetovano je postojanjem indeksa sortiranog prema ključu. Indeks se također proširuje odgovarajućim brojem uređenih parova koji nose informacije o ključevima i adresama logičkih slogova u osnovnom skupu.

Pretpostavimo da postojećem osnovnom skupu logičkih slogova želimo dodati slogove čiji su ključevi 1840, 0026 i 2615. U slučaju sortiranog osnovnog skupa novi slogovi morali bi se upisati na one adrese koje im prema redoslijedu ključeva pripadaju. To su adrese 03, 01 i 04. Budući da se na tim adresama nalaze već otprije upisani slogovi čiji su ključevi 1084, 2468 i 0097, u slučaju sortiranog osnovnog skupa, postupku dodavanja novih slogova morao bi prethoditi postupak reorganizacije osnovnog skupa, ili bi slogove trebalo povezati logičkim vezama. Sve bi to otežalo obradu i kompliciralo postupak održavanja datoteke.

Za slučaj osnovnog skupa logičkih slogova, koji nije sortirán, novi slogovi se upisuju na kraju niza. Istovremeno se izvršava i proširenje indeksa potrebnim brojem uređenih parova (K,A):

OSNOVNI SKUP		INDEKS	
R. Br	Ključ	Adresa	Ključ
1.	2468	04	0097
2.	3512	03	1084
3.	1084	01	2468
4.	0097	05	2680
5.	2680	02	3512
6.	1840	06	1840
7.	0026	07	0026
8.	2615	08	2615

Kako bi datoteka bila pripremljena za obradu, potrebno je indeks ponovo sortirati prema ključu. Pri tome redosljed logičkih slogova u osnovnom skupu ostaje nepromijenjen:

OSNOVNI SKUP		INDEKS	
R. Br	Ključ	Adresa	Ključ
1.	2468	07	0026
2.	3512	04	0097
3.	1084	03	1084
4.	0097	06	1840
5.	2680	01	2468
6.	1840	08	2615
7.	0026	05	2680
8.	2615	02	3512

Ilustracije radi slijednom obradom osnovnog skupa logički slog s pripadajućim ključem 0026 bio bi obrađen kao sedmi po redu. Korištenjem indeksne organizacije i slijednim pretraživanjem indeksa do adrese istog sloga dolazi se već u prvom koraku.

5.

Postupak održavanja indeksno organizirane datoteke nije složen. Pored dodavanja slogova ili promjene njihovog sadržaja (pri čemu se mogu mijenjati svi elementi, osim ključa sloga) zahtjevi obrade podataka obuhvaćaju i postupak isključivanja logičkih slogova iz osnovnog skupa. Postoje momenti u kojima neki od postojećih logičkih slogova iz bilo kojih razloga postaju nepotrebni za daljnju obradu. Samim tim prisustvo takvih slogova u osnovnom skupu postaje bespredmetno.

Uobičajenim postupkom suvišni (nepotrebni) slogovi brisali bi se iz osnovnog skupa, a paralelno s tim i iz indeksa. Međutim, to bi neizostavno uvjetovalo reorganizaciju i osnovnog skupa i indeksa. Reorganizacija datoteke uključuje i ponovni postupak kreiranja. Nije optimalno provoditi reorganizaciju datoteke ukoliko se pojavi samo jedan ili manja skupina

suvišnih, nepotrebnih slogova. Stoga se ovakvi slogovi posebno označuju i ne uzimaju u obzir prilikom obrade, a reorganizacija se izvršava onda kada broj nepotrebnih slogova postane dovoljno velik da bi opravdao takav postupak.

Nepotrebne (suvišne) slogove moguće je označiti nevažećima u osnovnom skupu, u indeksu ili na oba mjesta.

Ukoliko se slogovi označe kao nevažeći samo u osnovnom skupu logičkih slogova, tada se do tog saznanja dolazi tek preuzimanjem slogova iz osnovnog skupa. Preuzimanju slogova prethodi pretraživanje indeksa koje će se izvršiti u svakom slučaju, što produžava vrijeme izvođenja obrade.

U drugom slučaju slogovi se mogu označiti nevažećima samo u indeksu. Sada dolazi do uštede u vremenu jer se već samim korištenjem indeksa može ustanoviti koji su to slogovi, pa neće doći do njihovog preuzimanja iz osnovnog skupa. Ovakav postupak ima, međutim, jednu manu: u osnovnom skupu logičkih slogova ne postoje oznake za nevažeće slogove, pa bi se moglo dogoditi da slijedna obrada osnovnog skupa koja isključuje korištenje indeksa uzme u obzir i one slogove koji se ne bi smjeli obrađivati.

Najbolje je nevažeće slogove označiti i u indeksu i u osnovnom skupu. Slog koji je označen kao nevažeći (bilo u indeksu, bilo u osnovnom skupu) može se i dalje koristiti (npr. ukoliko se naknadno pokaže da je greškom označen kao nevažeći, tada se to može ispraviti). Međutim, korištenje »nevažećeg« sloga moguće je samo u osnovnom skupu, ali ne i u indeksu. Naime, nevažeći slog ne prestaje fizički postojati u osnovnom skupu, pa se slijednom obradom može (ako je to potrebno) i dalje koristiti. Nasuprot tome označavanje sloga nevažećim u indeksu u biti predstavlja »izbacivanje« uređenog para (K,A) iz indeksa. Iz toga se, međutim, nikako ne bi smio izvući pogrešan zaključak. U pravilu, nevažeće slogove označavamo i u osnovnom skupu logičkih slogova i u indeksu. Pri tome se prilikom izvođenja obrade pokazuju osnovne prednosti ovakvog postupka:

- ako je slog označen kao nevažeći samo u osnovnom skupu, tada je neohodno pretraživanjem indeksa preuzeti traženi slog, a njegov status »nevažećeg« mora se programski kontrolirati;
- ukoliko se slog označi nevažećim na oba mjesta (u osnovnom skupu i u indeksu), uređeni par K,A iz indeksa nestaje, nije potrebna programska kontrola u osnovnom skupu (jer do preuzimanja sloga ne može doći), a fizičko prisustvo nevažećeg sloga u osnovnom skupu omogućuje slijednu obradu, ukoliko se to pokaže potrebnim.

Pretpostavimo da se u već promatranom primejru indeksno organizirane datoteke pojavljuju:

- novi slogovi, čiji su ključevi 0123 i 1357, te
- nevažeći slogovi, čiji su ključevi 0026, 1084 i 2615.

Novi slogovi će se upisati na kraju niza uz istovremeno proširenje indeksa za odgovarajući broj uređenih parova. Nevažeći slogovi će dobiti oznaku »N« u odgovarajućem broju logičkih slogova osnovnog skupa i (u prvom koraku) radi lakšeg razumijevanja, ta će se oznaka pojaviti i u indeksu:

OSNOVNI SKUP			INDEKS		
R. Br	Ključ	Oznaka	Adresa	Ključ	Oznaka
1.	2468		07	0026	N
2.	3512		04	0097	
3.	1084	N	03	1084	N
4.	0097		06	1840	
5.	2680		01	2468	
6.	1840		08	2615	N
7.	0026	N	05	2680	
8.	2615	N	02	3512	
9.	0123		09	0123	
10.	1357		10	1357	

U drugom koraku nakon sortiranja indeksa prema ključu uz istovremeno »izbacivanje« uređenih parova onih slogova, koji su označeni nevažećim (pri čemu osnovni skup logičkih slogova ostaje nepromijenjen), dobiva se ovakav raspored:

OSNOVNI SKUP			INDEKS	
R. Br	Ključ	Oznaka	Adresa	Ključ
1.	2468			
2.	3512		04	0097
3.	1084	N	09	0123
4.	0097			
5.	2680		10	1357
6.	1840		06	1840
7.	0026	N	01	2468
8.	2615	N		
9.	0123		05	2680
10.	1357		02	3512

Napomena: u indeksu su samo zbog preglednosti ostavljene praznine na mjestima ređenih parova onih slogova koji su označeni nevažećima.

Prije izvođenja postupka dodavanja novih slogova i označavanja nekog od postojećih slogova nevažećim indeksno organizirana datoteka imala je ukupno 5, a zatim 8 slogova. Sada datoteka ima ukupno 10 slogova u osnovnom skupu. Međutim, kako su tri sloga proglašena i označena nevažećim, datoteka i pored 5 novih slogova ima samo 7 korisnih (što se lijepo može vidjeti u indeksu). Ti će se slogovi uzimati u obzir prilikom obrade, bilo da se radi o logički slijednoj ili o prekorednoj obradi podataka.

6.

Rezimirajući sveukupni tekst ovog rada, koji se odnosi na osnovne karakteristike indeksne organizacije, potrebno je napomenuti da je izlaganje bilo izvedeno na primjeru osnovnog i jednog od najjednostavnijih oblika te organizacije. Takav pristup omogućio je upoznavanje (prema: Tkalac, 1979) karakteristika osnovnog skupa, karakteristika indeksa kao skupa uređenih parova i karakteristika odnosa u kojem osnovni skup i indeks međusobno stoje. U tom kontekstu rad će biti dostupan i razumljiv širem krugu korisnika, što i jest njegov osnovni cilj.

L I T E R A T U R A

1. Flores, I. (1966), *Computer Programming*, Prentice-Hall Inc, Englewood Clifts. N.J.
2. Lipljin, N. (1980), *Programski primjeri kao mogućnost metodološkog pristupa izučavanju viših programskih jezika*, magistarski rad, Sveučilište u Zagrebu, Zagreb.
3. Manna, Z. / Waldinger, R. (1977), *Studies in Automatic Programming Logic*, Elsevier North-Holland.
4. Schmidt, V. (1972), *Visokoškolska didaktika*, Pedagoškoknjiževni Zbor, Zagreb.
5. Stanojević, M. (1977), *O profilima kadrova informatike*, članak, Zbornik radova VI jugoslovenskog savetovanja o informacionim sistemima, 406—418, Beograd.
6. Tkalac, S. (1979), *Struktura i organizacija podataka*, udžbenik, Fakultet organizacije i informatike Varaždin.

Primljeno: 1983-06-02

Lipljin N. Die Hauptmerkmale einer Index-Organisation

Z U S A M M E N F A S S U N G

In dieser Arbeit erklärt man die Hauptmerkmale der Grundmenge logischer Sätze und des dazugehörenden Indexes in der Form einer geordneten Menge, sowie der Beziehungen der Grundmenge zum Index. Man gibt auch einen illustrativen Beispiel einer Verarbeitung, die auf Vergleichung einer sequentiellen und Index-Datenorganisation gegründet ist. Dieser Beispiel weist auf die Vorteile der Nutzung einer Index-Organisation unter dem Aspekt der gesamten Such- und Übernahmezeit eines logischen Satzes aus dem Speicher.

Hier präsentiert man das Verfahren bei der Dateierstellung, die Eingabe neuer Sätze in die bestehende Datei und die Instandhaltung einer Index-Datei. Diese Darlegung enthält auch Beispiele, die auf eine einfache Weise die theoretischen Behauptungen verdeutlichen.

(Prijevod: Vesna Šimunić-Vučković)