Indeksowo-sekwencyjna organizacja pliku
sprawozdanie

Michał Tarnacki s188627

14 grudnia 2023

1 Wstęp

Celem projektu było zaimplementowanie algorytmu symulującego indeksowo-sekwencyjną organizację pliku. Metoda ta posiada wiele zalet względem organizacji sekwencyjnej. Ponieważ wykorzystywany jest indeks wskazujący na pierwsze rekordy poszczególnych stron dyskowych pliku z danymi, łatwo można odnaleźć tą część, w której leży szukany rekord, czy też do której powinien zostać wstawiony. Dokładniejszy opis zaimplementowanego algorytmu znajduje się w kolejnej sekcji.

2 Algorytm

2.1 Opis

2.1.1 Ogólne

Do zaimplementowania algorytmu został wykorzystany język Java. Domyślnie ustawione są następujące parametry:

- współczynnik wypełnienia strony po reorganizacji:

$$\alpha = 0.5$$

- współczynnik rozmiaru obszaru nadmiarowego w stosunku do obszaru głównego:

$$\frac{V}{N} = 0.2$$

- liczba rekordów na blok w obszarze głównym (bf) oraz w pliku indeksowym (bi):

$$bf = bi = 4$$

2.1.2 Rekord oraz indeks

Struktura rekordu prezentuje się następująco:

- klucz [int]
- objętość [double]
- promień [char]
- wysokość [char]
- indeks obszaru nadmiarowego [int]

Indeks natomiast:

- klucz [int]
- strona [int]

Zgodnie z językiem Java rekord zajmuje więc 20 B a indeks 8 B. Dla obu struktur najstarszy bit klucza zarezerwowany jest na flagę usunięcia a wartość IntMAX do zaznaczenia że dany rekord/indeks jest pusty (jest traktowany przez algorytm tak jakby nie istniał, jednak wypełnia niezajęte miejsca w blokach)

2.2 Szczegóły implementacyjne

- Każda klasa obsługująca plik [IndexFile, PrimaryArea, OverflowFile] dziedziczy po klasie bazowej File metody do operacji blokowych na plikach, w tym operacje odczytu oraz zapisu sekwencyjnego (zapisywany/odczytywany jest kolejny blok pliku) oraz operacje o dostępie swobodnym (odczytywany/zapisywany jest blok o numerze n).
- Ze względu na odczyt swobodny, możliwe jest odczytanie k-tego rekordu z pliku (numer bloku oraz przesunięcie obliczane na podstawie k oraz pojemności bloku).
- Każdy plik zapisuje ostatnio odczytany blok a więc odczyty nie są wykonywane ponownie po wywołaniu funkcji odczytu tego samego bloku.
- Plik zawierający indeksy odczytywany oraz zapisywany jest sekwencyjnie oraz, ze względu na stosunkowo niewielki rozmiar, w całości wczytywany jest do pamięci podręcznej. Dzięki temu po wczytaniu, przy dowolnej operacji, nie jest wykonywana żadna operacja dyskowa związana z tym plikiem.
- Indeksy w pamięci podręcznej szukane są wyszukiwaniem binarnym.
- W plikach znajdują się tylko indeksy/rekordy. Pliki nie zawierają żadnych dodatkowych metadanych.
- Reorganizacja wywoływana jest automatycznie, gdy plik indeksowy się wypełni lub na żądanie.
- Jeśli wstawiany rekord nie mieści się w obszarze głównym, wstawiany jest na koniec obszaru nadmiarowego a wskaźniki na rekordy ustawiane są tak aby tworzyły jednokierunkową, posortowaną rosnąco listę.
- Ze względu na odczyt/zapis blokowy stosunek V/N dotyczy liczby bloków a nie pojedynczych rekordów (w innym przypadku bloki zawierałyby miejsce nie do wykorzystania).

3 Korzystanie z programu

Do obsługi programu wykorzystać można następujące polecenia:

- open-file [filename] - tworzy (jeśli nie istniej) oraz otwiera plik [filename]
- print-files - wypisuje zawartość wszystkich plików
- print-all - wypisuje wszystkie rekordy w kolejności rosnącej
- insert [key] [radius] [height] - dodaje nowy rekord o podanych właściwościach
- remove-record [key] - usuwa rekord o podanym kluczu
- update-record [key] [newKey] [newRadius] [newHeight] - aktualizuje rekord o podanym kluczu
- reorganize - reorganizuje plik
- set-a - ustawia współczynnik α
- set-vn - ustawia współczynnik V/N
- toggle-print - włącza/wyłącza drukowanie liczby operacji dyskowych
- exit - kończy działanie programu

4 Eksperyment

W celu przeprowadzenia eksperymentu zostały dodane następujące opcje menu:

- insert-random [count] - dodaje [count] nowych losowych rekordów
- remove-random [count] - usuwa [count] istniejących rekordów rekordów
- read-record-random [count] - odczytuje [count] istniejących rekordów rekordów
- update-record-random-same-key [count] - aktualizuje [count] istniejących rekordów rekordów nie zmieniając klucza
- update-record-random-new-key [count] - aktualizuje [count] istniejących rekordów zmieniając klucz
- toggle-print-short - wypisuje sumę operacji dyskowych każdego polecenia
- remove-files - usuwa pliki testowe oraz wypisuje stosunek rozmiaru wszystkich wykorzystywanych plików do sytuacji gdy rekordy ułożone byłyby w jednym pliku sekwencyjnie

oraz przygotowane pliki tekstowe: test, test2 oraz test3 zawierające polecenia programu. Fragment pliku testowego:

```
open-file test set-a 0.50 set-vn 0.75 toggle-print-short insert-random 10000
remove-random 250 read-record-random 250 update-record-random-same-key 250
update-record-random-new-key 250 reorganize remove-files toggle-print-short

open-file test set-a 0.50 set-vn 1 toggle-print-short insert-random 10000
remove-random 250 read-record-random 250 update-record-random-same-key 250
update-record-random-new-key 250 reorganize remove-files toggle-print-short
```

W każdym pliku testowym testowane jest 16 przypadków ustawienia współczynnika α oraz V/N

$$(\alpha,\ \mathrm{V/N} \in \{0.25,\ 0.5,\ 0.75,\ 1\}).$$

a każda operacja związana z plikiem (wstawianie, usuwanie, edycja rekordu) wykonywana jest 250 razy (oprócz raz wykonywanej reorganizacji). Pliki testowe natomiast, różnią się jedynie ilością wstawianych rekordów (testowane 1000, 5000, 10000).

4.1 Test 1

Wyniki testu 1. prezentują się następująco:

		Liczba operacji dyskowych w zależności od parametrów α i V/N oraz od polecenia							
		Dodaj	Usuń	Odczytaj	Zmień rekord z tym samym kluczem	Zmień rekord z innym kluczem	Reorganizuj	Suma	Zajętość pamięci względem pliku sekwencyjnego
α	V/N	1000	250	250	250	250			
0.25	0.25	5505	561	302	567	1786	1945	10666	5,41
0.25	0.50	5840	610	337	613	2075	2128	11603	6,41
0.25	0.75	10180	500	250	500	1102	2507	15039	7,41
0.25	1.0	6551	798	469	754	3123	2571	14266	8,41
0.50	0.25	6574	530	293	554	1661	1245	10857	2,71
0.50	0.50	6120	666	416	654	3716	1069	12641	3,21
0.50	0.75	6808	628	360	617	2278	1506	12197	3,71
0.50	1.0	9111	516	265	519	1472	1509	13392	4,21
0.75	0.25	6825	530	283	535	2744	650	11567	1,81
0.75	0.50	6345	665	410	646	2584	871	11521	2,15
0.75	0.75	6921	588	334	583	2078	1182	11686	2,48
0.75	1.0	7229	640	388	628	2469	1321	12675	2,81
1.0	0.25	7839	535	294	550	2377	628	12223	1,35
1.0	0.50	7268	570	342	591	2944	605	12320	1,6
1.0	0.75	7743	541	311	563	2004	991	12153	1,85
1.0	1.0	8957	502	251	502	1620	929	12761	2,11

		Średnia liczba operacji dyskowych w zależności od parametrów α i V/N przypadająca na każde polecenie					
		Dodaj	Usuń	Odczytaj	Zmień rekord z tym samym kluczem	Zmień rekord z innym kluczem	
α	V/N	1000	250	250	250	250	Suma
0.25	0.25	5,505	2,244	1,208	2,268	7,144	18,369
0.25	0.50	5,84	2,44	1,348	2,452	8,3	20,38
0.25	0.75	10,18	2	1	2	4,408	19,588
0.25	1.0	6,551	3,192	1,876	3,016	12,492	27,127
0.50	0.25	6,574	2,12	1,172	2,216	6,644	18,726
0.50	0.50	6,12	2,664	1,664	2,616	14,864	27,928
0.50	0.75	6,808	2,512	1,44	2,468	9,112	22,34
0.50	1.0	9,111	2,064	1,06	2,076	5,888	20,199
0.75	0.25	6,825	2,12	1,132	2,14	10,976	23,193
0.75	0.50	6,345	2,66	1,64	2,584	10,336	23,565
0.75	0.75	6,921	2,352	1,336	2,332	8,312	21,253
0.75	1.0	7,229	2,56	1,552	2,512	9,876	23,729
1.0	0.25	7,839	2,14	1,176	2,2	9,508	22,863
1.0	0.50	7,268	2,28	1,368	2,364	11,776	25,056
1.0	0.75	7,743	2,164	1,244	2,252	8,016	21,419
1.0	1.0	8,957	2,008	1,004	2,008	6,48	20,457

Możemy zauważyć iż najmniej operacji zostało wykonanych dla $\alpha = 0.25$ oraz $V/N = 0.25$. Wynika to najpewniej z rzadko przeprowadzanej reorganizacji, bo choć jej koszt jest jednym z wyższych to koszt operacji dodawania jest najniższy (a więc rzadko trzeba reorganizować). Przez takie ustawienie parametrów dużo miejsca zajmują jednak puste przestrzenie. Zatem, choć nie jest to zajętość ponad 8-krotnie większa niż dla pliku sekwencyjnego, daleko jest jej do wartości minimalnej jaką udało się osiągnąć czyli 1.35.

4.2 Test 2

Wyniki testu 2. prezentują się następująco:

		Liczba operacji dyskowych w zależności od parametrów α i V/N oraz od polecenia							
		Dodaj	Usuń	Odczytaj	Zmień rekord z tym samym kluczem	Zmień rekord z innym kluczem	Reorganizuj	Suma	Zajętość pamięci względem pliku sekwencyjnego
α	V/N	5000	250	250	250	250			
0.25	0.25	40193	500	250	500	1018	12090	54551	5,4
0.25	0.50	29238	653	401	633	2132	11687	44744	6,4
0.25	0.75	35544	992	644	945	3771	13264	55160	7,4
0.25	1.0	33895	673	435	664	2288	14099	52054	8,4
0.50	0.25	35946	514	265	516	1304	6216	44761	2,7
0.50	0.50	32290	628	399	627	2080	7186	43210	3,2
0.50	0.75	45601	504	255	505	1216	7381	55462	3,7
0.50	1.0	39838	601	360	595	1829	8226	51449	4,2
0.75	0.25	33306	573	348	578	7060	4048	45913	1,8
0.75	0.50	33337	601	364	596	2018	5441	42357	2,13
0.75	0.75	35292	603	359	605	2032	5791	44682	2,47
0.75	1.0	39267	566	341	582	1769	5994	48519	2,8
1.0	0.25	39395	550	315	561	6215	2984	50020	1,35
1.0	0.50	37298	569	341	588	1882	4385	45063	1,6
1.0	0.75	41260	529	276	531	1605	4214	48415	1,85
1.0	1.0	40824	564	336	582	1861	4945	49112	2,1

		Średnia liczba operacji dyskowych w zależności od parametrów α i V/N przypadająca na każde polecenie					
		Dodaj	Usuń	Odczytaj	Zmień rekord z tym samym kluczem	Zmień rekord z innym kluczem	
α	V/N	5000	250	250	250	250	Suma
0.25	0.25	8,0386	2	1	2	4,072	17,1106
0.25	0.50	5,8476	2,612	1,604	2,532	8,528	21,1236
0.25	0.75	7,1088	3,968	2,576	3,78	15,084	32,5168
0.25	1.0	6,779	2,692	1,74	2,656	9,152	23,019
0.50	0.25	7,1892	2,056	1,06	2,064	5,216	17,5852
0.50	0.50	6,458	2,512	1,596	2,508	8,32	21,394
0.50	0.75	9,1202	2,016	1,02	2,02	4,864	19,0402
0.50	1.0	7,9676	2,404	1,44	2,38	7,316	21,5076
0.75	0.25	6,6612	2,292	1,392	2,312	28,24	40,8972
0.75	0.50	6,6674	2,404	1,456	2,384	8,072	20,9834
0.75	0.75	7,0584	2,412	1,436	2,42	8,128	21,4544
0.75	1.0	7,8534	2,264	1,364	2,328	7,076	20,8854
1.0	0.25	7,879	2,2	1,26	2,244	24,86	38,443
1.0	0.50	7,4596	2,276	1,364	2,352	7,528	20,9796
1.0	0.75	8,252	2,116	1,104	2,124	6,42	20,016
1.0	1.0	8,1648	2,256	1,344	2,328	7,444	21,5368

Tym razem najmniej operacji zostało wykonanych dla $\alpha = 0.75$ oraz $\mathrm{V/N} = 0.5$. Wynik zdaje się być nieco odmienny od zaobserwowanego w poprzednim teście. W badanym przypadku koszt reorganizacji jest jednym z niższych. Widać także że koszt dodawania jest stosunkowo niski. Udało się więc osiągnąć pewne optimum - reorganizacja wykonywana jest niezbyt często oraz w miarę niskim kosztem. Ponownie najlepiej wykorzystywana jest pamięć dla parametrów $\alpha = 1$ oraz $\mathrm{V/N} = 0.25$. Pomijając reorganizację, ponownie najlepszy okazał się pierwszy przypadek.

4.3 Test 3

Wyniki testu 3. prezentują się następująco:

		Liczba operacji dyskowych w zależności od parametrów α i V/N oraz od polecenia							
		Dodaj	Usuń	Odczytaj	Zmień rekord z tym samym kluczem	Zmień rekord z innym kluczem			Zajętość pamięci względem pliku sekwencyjnego
α	V/N	10000	250	250	250	250	Reorganizuj	Suma	
0.25	0.25	54948	582	303	571	1594	21240	79238	5,4
0.25	0.50	81713	506	254	509	1128	25386	109496	6,4
0.25	0.75	71847	557	290	555	1484	26227	100960	7,4
0.25	1.0	81380	1037	868	1134	4043	29456	117918	8,4
0.50	0.25	67876	543	282	526	1467	12575	83269	2,7
0.50	0.50	67857	594	326	576	1754	14024	85131	3,2
0.50	0.75	68682	681	399	674	2294	15821	88551	3,7
0.50	1.0	77216	839	582	863	3214	18038	100752	4,2
0.75	0.25	66831	577	312	565	1821	9512	79618	1,8
0.75	0.50	65718	650	380	624	2275	11095	80742	2,13
0.75	0.75	72656	590	324	577	1858	11232	87237	2,47
0.75	1.0	89219	516	263	509	1368	11062	102937	2,8
1.0	0.25	78367	566	305	544	10724	6119	96625	1,35
1.0	0.50	77749	543	281	525	1678	7893	88669	1,6
1.0	0.75	77714	570	312	554	1882	9151	90183	1,85
1.0	1.0	81423	572	317	558	1888	9782	94540	2,1

		Średnia liczba operacji dyskowych w zależności od parametrów α i V/N przypadająca na każde polecenie					
		Dodaj	Usuń	Odczytaj	Zmień rekord z tym samym kluczem	Zmień rekord z innym kluczem	
α	V/N	10000	250	250	250	250	Suma
0.25	0.25	5,4948	2,328	1,212	2,284	6,376	17,6948
0.25	0.50	8,1713	2,024	1,016	2,036	4,512	17,7593
0.25	0.75	7,1847	2,228	1,16	2,22	5,936	18,7287
0.25	1.0	8,138	4,148	3,472	4,536	16,172	36,466
0.50	0.25	6,7876	2,172	1,128	2,104	5,868	18,0596
0.50	0.50	6,7857	2,376	1,304	2,304	7,016	19,7857
0.50	0.75	6,8682	2,724	1,596	2,696	9,176	23,0602
0.50	1.0	7,7216	3,356	2,328	3,452	12,856	29,7136
0.75	0.25	6,6831	2,308	1,248	2,26	7,284	19,7831
0.75	0.50	6,5718	2,6	1,52	2,496	9,1	22,2878
0.75	0.75	7,2656	2,36	1,296	2,308	7,432	20,6616
0.75	1.0	8,9219	2,064	1,052	2,036	5,472	19,5459
1.0	0.25	7,8367	2,264	1,22	2,176	42,896	56,3927
1.0	0.50	7,7749	2,172	1,124	2,1	6,712	19,8829
1.0	0.75	7,7714	2,28	1,248	2,216	7,528	21,0434
1.0	1.0	8,1423	2,288	1,268	2,232	7,552	21,4823

W tym przypadku wyniki są bardzo zbliżone do testu 1. Ponownie $\alpha = 0.25$ oraz V/N $= 0.25$ okazały się najlepszymi parametrami jeśli chodzi o koszt wszystkich operacji, a $\alpha = 1$ oraz V/N $= 0.25$ jeśli chodzi o zajętość pamięci.

4.4 Dodatkowe obserwacje

- Dla $\alpha = 1$ oraz V/N $= 0.25$ w każdym teście najoptymalniej wykorzystujemy pamięć oraz koszt reorganizacji jest jednym z najniższych. Jednak reorganizacje są częste przez co koszt operacji dodawania jest wysoki.
- Mimo wysokiego kosztu reorganizacji $\alpha = 0.25$ oraz V/N $= 0.25$ prawie w każdym teście posiadał najmniejszy koszt wszystkich operacji. Daleko mu do minimalnego wykorzystania pamięci jednak to $\alpha = 0.25$ oraz V/N $= 1$ zawsze najgorzej wykorzystywał pamięć

4.5 Wnioski

Nie istnieje jednoznacznie najlepszy zestaw parametrów. Gdy chcemy aby operacje koszt operacji dyskowych był najniższy, należy zadbać aby najbardziej kosztowna operacja czyli reorganizacja wykonywana była najrzadziej. Wiąże się to jednak z większą zajętością pamięci. Gdy chcemy zajmować mało pamięci, częściej będziemy wykonywać reorganizacje. Moim zdaniem pewnym kompromisem są parametry $\alpha = 0.5$ oraz V/N $= 0.25$. Łączą one stosunkowo niski koszt wszystkich operacji oraz niską zajętość pamięci.