Indeksowo-sekwencyjna organizacja pliku

1. Wstęp

Celem projektu było zaprojektowanie i zaimplementowanie indeksowej organizacji pliku. W projekcie zrealizowano indeksowo-sekwencyjną organizację pliku. Zadanie zostało wykonane w języku C++. Zostały zaimplementowane operacje: wstawiania, aktualizacji, odczytu, usuwania rekordu, wyświetlania zawartości pliku z danymi i indeksu, przeglądania całej zawartości pliku zgodnie z kolejnością wartości klucza, reorganizacji automatycznej i na żądanie, czytania poleceń z pliku testowego. Program działa w sposób interaktywny. Po pobraniu i wykonaniu komendy można przedstawić jej wynik i podać kolejną komendę. Jest możliwość wczytania danych testowych z pliku i zakończenia działania programu, lub opcja wczytania danych testowych i dalszej interakcji.

2. Opis rekordu

Rekordy w pliku składają się z dziesięciu parametrów: $key,\ a_0,\ a_1,\ a_2,\ a_3,\ a_4,\ x,\ deleted,\ pointerToPage,\ pointerToPosition$. Wartość klucza jest obliczana ze wzoru $key = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4$. Wszystkie wymienione pola rekordu są typu całkowitego int. W rezultacie rekord jest stałej wielkości 40. bajtów. Pola $pointerToPage,\ pointerToPosition$to wskaźniki na numer strony i pozycji na stronie w Obszarze Nadmiarowym.

3. Opis indeksu

Pojedynczy indeks w pliku indeksowym składa się z dwóch parametrów $key,\ page$. Wszystkie pola są typu całkowitego int. W rezultacie pojedynczy indeks jest stałej wielkości 8. bajtów. Pola $key,\ page$ to odpowiednio klucz pierwszego rekordu na stronie i numer strony.

4. Opis zastosowanej metody

W projekcie przyjęto, że na początku programu rezerwowane jest w pliku głównym po jednej stronie na Obszar Główny i Obszar Nadmiarowy. Strona ma rozmiar 160 bajtów. Strony są zapełniane pustymi wartościami, które przy wykonywanych operacjach będą zamieniane na odpowiednie wartości. Pierwszym rekordem na pierwszej stronie Obszaru Głównego zawsze będzie sztuczny rekord o kluczu równym -1 ustawiony jako rekord usunięty. Ma to na celu umożliwienie wstawianie mniejszych rekordów, niż dodane wcześniej.Przyjęto stały współczynnik blokowania dla stron w pliku głównym i w indeksie. Wynoszą one odpowiednio BLOCK_RATE = 4 i INDEX_BLOCK_RATE = 20 (współczynnik blokowania dla indeksu jest pięć razy większy od współczynnika blokowania pliku głównego, co wynika z rozmiarów pojedynczego rekordu i wpisu w indeksie). Gdy rekordy nie mieszczą się w Obszarze Głównym dodawane są do Obszaru Nadmiarowego. Gdy Obszar Nadmiarowy zostanie zapełniony i nie mam miejsca na dodanie rekordu następuje automatyczna reorganizacja (jest możliwość zmiany warunków reorganizacji. Został ustalony stały współczynnik BETA = 0.2. Jeżeli stosunek liczby rekordów w Obszarze Nadmiarowym do liczby rekordów w Obszarze Głównym przekroczy dany współczynnik lub gdy Obszar Nadmiarowy jest pełen następuje automatyczna reorganizacja. Sposób ten jednak utrudnia sprawdzenie poprawności wstawiania rekordów do Obszaru Nadmiarowego). W czasie reorganizacji tworzona jest kopia pliku głównego i tworzony jest nowy plik o większej liczbie stron. Strony

nowego pliku są zapełniane do połowy (zgodnie ze współczynnikiem α), aby po reorganizacji nowe rekordy mogły być dodawane do Obszaru Głównego. Nowa liczba stron w Obszarze Głównym jest ustalana na podstawie wzoru:

$$S_{N\,new} = \lceil (N + V)/(bf \cdot \alpha) \rceil$$

$gdzie$:
$N - liczba\ rekordów\ w\ Obszarze\ Głównym$
$V - liczba\ rekordów\ w\ Obszarze\ Nadmiarowym$
$bf - współczynnik\ blokowania\ w\ pliku\ głównym\ (BLOCKRATE = 4)$
$\alpha - współczynnik\ wykorzystania\ strony\ w\ Obszarze\ Głównym\ po\ reorganizacji\ (ALFA = 0.5)$

a liczba stron Obszaru Nadmiarowego jest ustalana na podstawie wzoru:

$$S_{V\,new} = \lceil S_{N\,new} \cdot VNRATIO \rceil$$

$gdzie$:
$S_{N\,new} - nowa\ liczba\ stron\ w\ Obszarze\ Głównym$
$VNRATIO - stały\ współczynnik\ równy\ 0.2$

Po reorganizacji tworzony jest indeks, który stanowi klucze pierwszych rekordów na stronie i numery tych stron. Liczba stron indeksu zależy od liczby stron w Obszarze Głównym i jest określana wzorem:

$$SI_{N\,new} = \lceil S_{N\,new} / bi \rceil$$

$gdzie$:
$S_{N\,new} - liczba\ stron\ w\ Obszarze\ Głównym$
$bi - współczynnik\ blokowania\ w\ indeksie\ (INDEXBLOCKRATE = 20)$

Indeks pozwala na szybszy dostęp asocjacyjny. Aby wstawić lub odszukać konkretny rekord, w indeksie wyszukiwany jest numer strony, na której może się znajdować (klucz rekordu musi być większy od pierwszego klucza na stronie, ale mniejszy od pierwszego klucza na kolejnej stronie) i do bufora w pamięci operacyjnej wczytywana jest tylko jedna strona. Jeżeli strona jest w buforze, nie jest ponownie czytana.

Operacja usuwania rekordu polega na zaznaczeniu flagi *deleted.* Faktyczne usuwanie dokonuje się podczas reorganizacji. Do zaznaczonego do usunięcia rekordu można dowiązać wskaźnik na następny rekord w Obszarze Nadmiarowym. Taki rekord nie będzie wyświetlony po operacji przeglądania całej zawartości pliku zgodnie z kolejnością wartości klucza (*s*).

Aktualizować można rekordy, które nie są zaznaczone do usunięcia. Aktualizacja bez zmiany klucza może wystąpić, gdy nowo podany rekord ma klucz równy wartości klucza rekordu w pliku. Wtedy aktualizacji podlegają parametry a_0, a_1, a_2, a_3, a_4, x. Aktualizacja ze zmianą klucza polega na zaznaczeniu do usunięcia rekordu o podanym kluczu i wstawieniu nowego rekordu.

5. Specyfikacja formatu plików dyskowych i pliku testowego

Pliki dyskowe są realizowane w formie plików binarnych z rozszerzeniem .bin. Przyjęto, że plik testowy może mieć format tekstowy z rozszerzeniem .txt. Plik testowy stanowi

sekwencja operacji, które umożliwia program. Każda operacja zapisana jest w nowym wierszu. Pojedyncza operacja ma format: SymbolOperacji Parametry. Sekwencja operacji musi być zakończona symbole *e* oznaczającym zakończenie wczytywania.

Nazwa operacji	Symbol Operacji	Parametry
dodaj rekord	*a*	$a_0\ a_1\ a_2\ a_3\ a_4\ x$
aktualizuj rekord	*u*	$kluczRekorduDoAktualizacjia_0\ a_1\ a_2\ a_3\ a_4\ x$
odczytaj rekord	*r*	$kluczRekordu$
usuń rekord	*d*	$kluczRekordu$
wyświetl zawartość pliku głównego	*f*	
wyświetl zawartość pliku indeksowego	*i*	
przeglądaj zawartość pliku zgodnie z kolejnością wartości klucza	*s*	
reorganizuj	*o*	
wczytaj dane testowe	*t*	
koniec wczytywania	*e*	

6. Sposób prezentacji wyników działania programu

W przypadku prezentacji wyników działania programu wykonującego operacje z pliku testowego na ekranie wyświetlane są opisy poszczególnych operacji. W przypadku wprowadzania operacji z klawiatury istnieje możliwość wyświetlania wyników programu na bieżąco po każdej wykonanej operacji.

Wyświetlanie zawartości plików z danymi daje możliwość sprawdzenia , czy rekord znajduje się w części głównej, czy nadmiarowej, dokładnie w którym miejscu, ile jest pustych miejsc na poszczególnych stronach pliku, czy też zawartości poszczególnych łańcuchów przepełnień. Oznaczenie w formie *numerStrony.numerPozycji* informuje na jakiej stronie i na jakiej pozycji znajduje się rekord. Wskaźniki w tej samej postaci oznaczają na jakiej stronie i na jakiej pozycji w Obszarze Nadmiarowym znajduje się kolejny rekord. Przykład poniżej prezentuje opisany schemat:

PRIMARY AREA:
0.0 key: -1 pointer: NULL deleted: 1
0.1 key: 10 a0: 10 a1: 0 a2: 0 a3: 0 a4: 0 x: 0 deleted: 0 pointer: NULL
0.2 NULL
0.3 NULL
1.0 key: 20 a0: 20 a1: 0 a2: 0 a3: 0 a4: 0 x: 0 deleted: 0 pointer: NULL
1.1 key: 30 a0: 30 a1: 0 a2: 0 a3: 0 a4: 0 x: 0 deleted: 0 pointer: NULL

1.2 NULL
1.3 NULL
2.0 key: 45 a0: 45 a1: 0 a2: 0 a3: 0 a4: 0 x: 0 deleted: 0 pointer: NULL
2.1 key: 50 a0: 50 a1: 0 a2: 0 a3: 0 a4: 0 x: 0 deleted: 0 pointer: NULL
2.2 NULL
2.3 NULL
3.0 key: 55 a0: 55 a1: 0 a2: 0 a3: 0 a4: 0 x: 0 deleted: 0 pointer: NULL
3.1 key: 57 a0: 57 a1: 0 a2: 0 a3: 0 a4: 0 x: 0 deleted: 0 pointer: NULL
3.2 key: 59 a0: 59 a1: 0 a2: 0 a3: 0 a4: 0 x: 0 deleted: 0 pointer: NULL
3.3 key: 60 a0: 60 a1: 0 a2: 0 a3: 0 a4: 0 x: 0 deleted: 0 pointer: 0.2
OVERFLOW AREA:
0.0 key: 70 a0: 70 a1: 0 a2: 0 a3: 0 a4: 0 x: 0 deleted: 0 pointer: NULL
0.1 key: 65 a0: 65 a1: 0 a2: 0 a3: 0 a4: 0 x: 0 deleted: 0 pointer: 0.3
0.2 key: 63 a0: 63 a1: 0 a2: 0 a3: 0 a4: 0 x: 0 deleted: 0 pointer: 0.1
0.3 key: 66 a0: 66 a1: 0 a2: 0 a3: 0 a4: 0 x: 0 deleted: 0 pointer: 0.0

Po każdej przeprowadzonej operacji wyświetlana jest liczba zapisów i odczytów stron dyskowych, które wymaga dana opcja.

7. Eksperyment

Celem eksperymentu było ustalenie wpływu parametrów implementacyjnych i liczby przechowywanych w pliku rekordów na złożoność poszczególnych operacji w pliku. Zostanie pokazana średnia liczba odczytów i zapisów stron dyskowych na jedną operację, zajętość pamięci przez plik i indeks w funkcji liczby rekordów. W analizie problemu będą wzięte pod uwagę następujące parametry implementacyjne:

- α – $współczynnik\ wykorzystania\ strony\ w\ Obszarze\ Głównym\ po\ reorganizacji$
- β – $współczynnik\ warunku\ automatycznej\ reorganizacji$

Do eksperymentów wygenerowano losowo różną liczbę Nrekordów i wykorzystano różne współczynniki α i β. $N = \{10,\ 50,\ 100,\ 500\}$ (w *N* nie zawiera się rekord sztuczny o kluczu -1), $\alpha = \{0.2,\ 0.5,\ 0.8\}$, $\beta = \{0.2,\ 0.5,\ 0.8\}$. Stworzono również generator operacji i parametrów. Ze względu na dużą liczbę rekordów współczynnik blokowania w pliku głównym został ustawiony na 10 (*BLOCK_RATE* = 10, *INDEX_BLOCK_RATE* = 50).

Wyniki zostaną omówione i zostanie rozważona optymalna wartość badanych parametrów.

W tabeli pokazano średnią liczbę operacji dyskowych w zależności od liczby wygenerowanych rekordów, współczynnika α i β.

N	ALFA	BETA	add	delete	update	readRecord	reorganise	sequence	index
10	0,2	0,2	3	2	4	2	43	6	1
10	0,2	0,5	3	2	4	2	43	6	1
10	0,2	0,8	3	2	4	2	43	6	1
10	0,5	0,2	3	2	4	2	23	3	1
10	0,5	0,5	3	2	4	2	23	3	1
10	0,5	0,8	3	2	4	2	23	3	1
10	0,8	0,2	3	2	4	2	17	2	1

10	0,8	0,5	3	2	4	2	17	2	1
10	0,8	0,8	3	2	4	2	17	2	1
50	0,2	0,2	3	2	4	2	172	26	1
50	0,2	0,5	3	2	4	2	173	26	1
50	0,2	0,8	3	2	4	2	175	26	1
50	0,5	0,2	3	2	4	2	73	11	1
50	0,5	0,5	4	2	4	2	75	11	1
50	0,5	0,8	4	2	4	2	78	11	1
50	0,8	0,2	3	2	4	2	46	7	1
50	0,8	0,5	4	2	4	2	50	7	1
50	0,8	0,8	4	2	4	2	51	7	1
100	0,2	0,2	3	2	4	2	336	51	2
100	0,2	0,5	3	2	4	2	349	51	2
100	0,2	0,8	3	2	4	2	349	51	2
100	0,5	0,2	3	2	4	2	134	21	1
100	0,5	0,5	4	2	4	2	140	21	1
100	0,5	0,8	4	2	4	2	141	21	1
100	0,8	0,2	3	2	4	2	80	13	1
100	0,8	0,5	4	2	4	2	86	13	1
100	0,8	0,8	5	2	4	2	87	13	1
500	0,2	0,2	3	2	6	2	1682	250	5
500	0,2	0,5	4	2	5	2	1719	250	5
500	0,2	0,8	4	2	5	2	1732	250	5
500	0,5	0,2	3	2	4	2	643	100	2
500	0,5	0,5	4	2	4	2	663	100	2
500	0,5	0,8	5	2	5	2	666	100	2
500	0,8	0,2	4	2	4	2	372	63	2
500	0,8	0,5	5	2	4	2	390	63	2
500	0,8	0,8	5	2	4	2	398	63	2

Dodanie rekordu wymaga średnio do 3. do 5. operacji dyskowych. Na tą liczbę ma wpływ operacja szukania odpowiedniego numeru strony w indeksie. Różnice mogą wynikać ze specyfikacji istniejących już rekordów a także z konieczności dodawania ich do Obszaru Głównego lub Nadmiarowego. Minimalna liczba operacji to 3 (odszukanie strony indeksu, odczytanie konkretnej strony i zapisanie na stronę).Parametry implementacyjne α i β nie mają wpływu na operację dodawania.
Operacja zaznaczenia do usunięcia wymaga średnio 2 operacje, niezależnie od N, α i β.
Aktualizacja rekordu wymaga średnio od 4. do 6. operacji. Nie widać zależności z N, α i β. Średnia ta jest większa , gdyż przy zmianie klucza wymaga zarówno usunięcia i dodania.
Odczyt pojedynczego rekordu wymaga średnio 2. operacji w każdym przypadku.
Należy zauważyć, że wartości były generowane losowo, dlatego rekord do usunięcia, aktualizacji, odczytu mógł nie istnieć. Na większą średnią tych operacji może mieć wpływ konieczność przeszukiwania Obszaru Głównego i Nadmiarowego.

Parametry implementacyjne α i β mają znaczący wpływ na reorganizację. Zmiana α powoduje większą różnicę w średniej liczbie operacji niż zmiana β. Im większa α, tym mniejsza liczba operacji. Powodem jest mniejsza liczba stron, ich większe zapełnienie i mniejsza wolna przestrzeń w Obszarze Głównym.Mimo, że zwiększająca się β powoduje rzadszą reorganizację, liczba operacji rośnie w większości przypadków. Wraz ze wzrostem Nliczba średnia liczba operacji znacznie rośnie.

Średnia liczba operacji przeznaczona na przeglądanie zawartości pliku zgodnie z kolejnością wartości klucza maleje wraz ze wzrostem α, ponieważ wraz ze wzrostem tego parametru jest mniej stron do odczytu. β nie ma wpływu na średnią dla tej operacji.

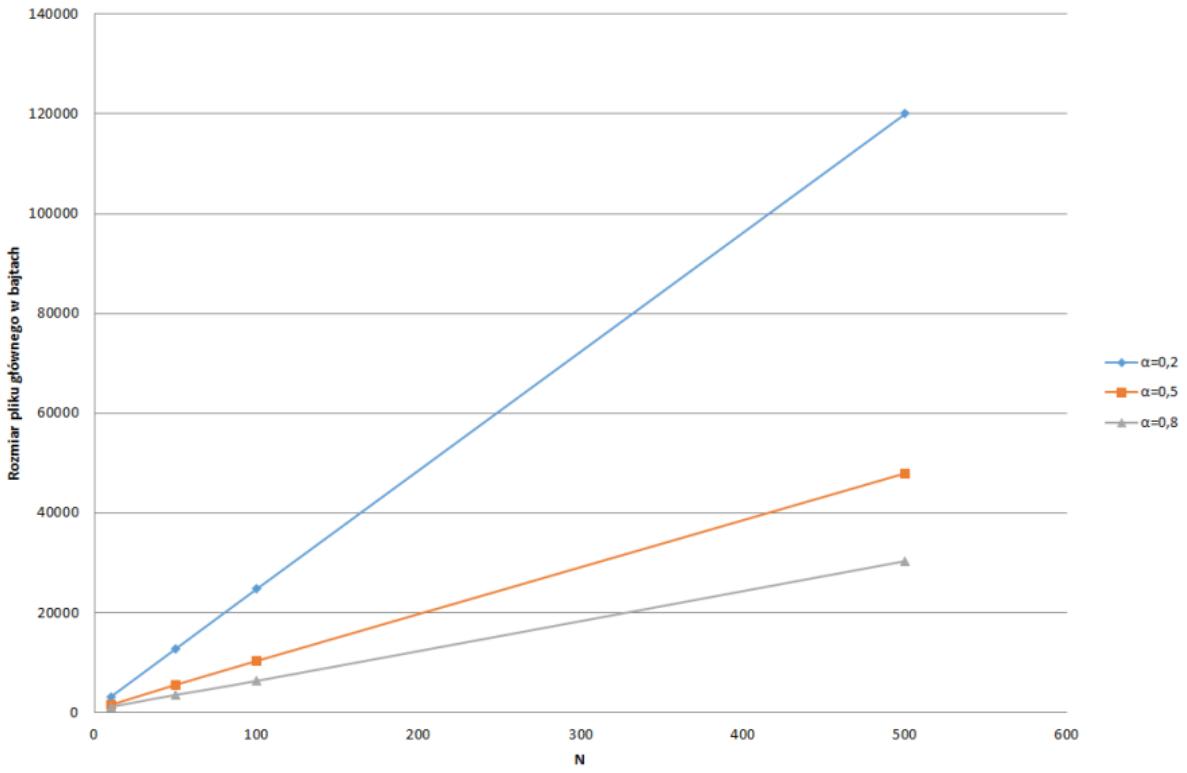
Odczytanie indeksu w badanych przypadkach wymaga średnio od 1. do 5. operacji. Wartości te są uzależnione od N, gdyż wraz ze wzrostem Nrośnie liczba stron w pliku głównym a tym samym rośnie liczba stron indeksu. (Jedna strona indeksu ma pojemność 5 razy większą niż pojemność strony w pliku głównym).. Dla większych, niezbadanych Nwartość ta byłaby większa.

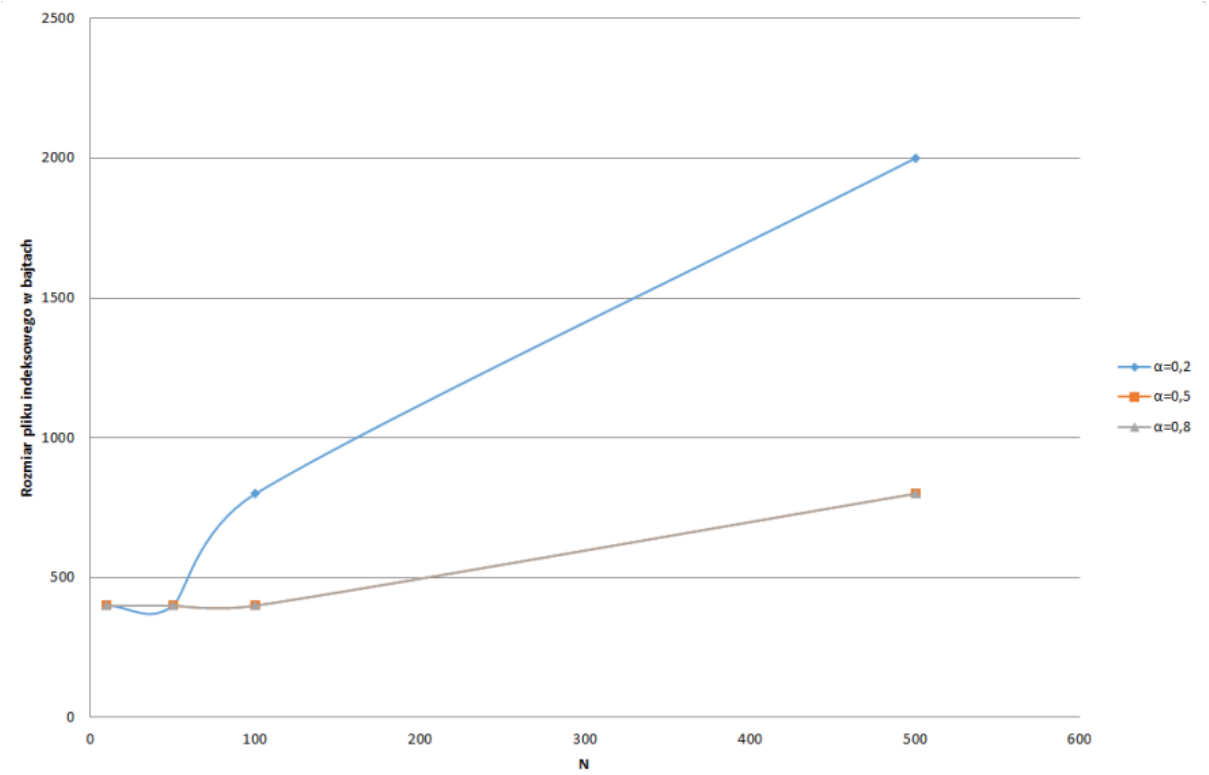
Ustalenie najlepszych parametrów implementacyjnych można dokonać w oparciu o dane dla N większych niż 10 . Dla każdego $N = \{50, 100, 500\}$zaznaczono minimalną wartość w każdej kolumnie. Na tej podstawie wybrano parametry $\alpha = 0.8$ i $\beta = 0.2$, dla których eksperyment dał najkorzystniejszą liczbę operacji.

Tabela przedstawia zajętość pamięci liczonej w bajtach przez plik główny i indeksowy, które są wynikiem przeprowadzonych eksperymentów

N	ALFA	BETA	rozmiar pliku głównego w bajtach	rozmiar pliku indeksowego w bajtach
10	0,2	0,2	3200	400
10	0,2	0,5	3200	400
10	0,2	0,8	3200	400
10	0,5	0,2	1600	400
10	0,5	0,5	1600	400
10	0,5	0,8	1600	400
10	0,8	0,2	1200	400
10	0,8	0,5	1200	400
10	0,8	0,8	1200	400
50	0,2	0,2	12800	400
50	0,2	0,5	12800	400
50	0,2	0,8	12800	400
50	0,5	0,2	5600	400
50	0,5	0,5	5600	400
50	0,5	0,8	5600	400
50	0,8	0,2	3600	400
50	0,8	0,5	3600	400

50	0,8	0,8	3600	400
100	0,2	0,2	24800	800
100	0,2	0,5	24800	800
100	0,2	0,8	24800	800
100	0,5	0,2	10400	400
100	0,5	0,5	10400	400
100	0,5	0,8	10400	400
100	0,8	0,2	6400	400
100	0,8	0,5	6400	400
100	0,8	0,8	6400	400
500	0,2	0,2	120000	2000
500	0,2	0,5	120000	2000
500	0,2	0,8	120000	2000
500	0,5	0,2	48000	800
500	0,5	0,5	48000	800
500	0,5	0,8	48000	800
500	0,8	0,2	30400	800
500	0,8	0,5	30400	800
500	0,8	0,8	30400	800





Pliki o większej liczbie rekordów zajmują więcej bajtów. Większy parametr α pozwala na większe zapełnienie stron, dlatego jest mniej wolnych przestrzeni i plik zajmuje mniej bajtów. Zmiana parametru β nie ma wpływu na zajętość pamięci. Wraz z większym N rośnie liczba stron w Obszarze Głównym, a tym samym rośnie liczba stron indeksu, lecz zapełnia się ona wolniej niż strona pliku głównego. Rozmiar pliku głównego rośnie liniowo wraz z zwiększającą się liczbą rekordów