

# Sprawozdanie

## Implementacja pliku sekwencyjno-indeksowego

Bazyli Polednia, ETI 2020

### 1. Wstęp teoretyczny

Systemy baz danych wykorzystują wiele struktur plików w celu przechowywania rekordów. Jednym z najprostszych są pliki sekwencyjne – do rekordów dodany jest klucz (zazwyczaj w postaci liczby naturalnej), na podstawie którego szeregowane są kolejne wpisy. Rozwiązanie to gwarantuje prostotę przeglądania kolejnych rekordów (w pliku występują one już w formie posortowanej według kluczy), natomiast problem stanowią operacje dodawania i usuwania rekordów - wymagałoby to przesunięcia wszystkich rekordów z kluczami większymi niż dodawany/usuwany. W przypadku dużej ilości rekordów równie kłopotliwe jest aktualizowanie rekordu – gdyby znajdował się on na końcu pliku, wymagałoby to odczytania wszystkich stron dyskowych w bazie.

Z powyższych powodów częściej używa się struktury sekwencyjno-indeksowej. Polega ona na użyciu dodatkowego pliku, w którym przechowuje się klucze pierwszych rekordów znajdujących się na kolejnych stronach dyskowych. Plik ten jest stosunkowo mniejszy niż plik z rekordami, co pozwala na przetrzymywanie go w pamięci operacyjnej. Dzięki temu chcąc uzyskać dostęp do rekordu (np. w celu dodania, usunięcia lub aktualizacji), potrzebne jest wykonanie tylko jednego odczytu strony dyskowej (chyba że rekord znajduje się w przestrzeni dodatkowej, o czym później). Dodatkowo, pliki sekwencyjno-indeksowe używają dodatkowej przestrzeni, nazywanej dalej overflow, w którym przechowuje się rekordy, które nie mieszczą się w głównej części pliku. Aby uniknąć nadmiernego powiększania się rozmiaru overflow, regularnie trzeba wywoływać operację reorganizacji. Zwiększana jest wtedy ilość stron dyskowych w głównej części pliku, a rekordy znajdujące się w overflow są do nich przepisywane wraz z rekordami znajdującymi się wcześniej w głównym obszarze, zerując tym samym rozmiar overflow.

### 2. Opis implementacji

Klasą reprezentującą plik sekwencyjno-indeksowy jest SeqIndFile. Udostępnia ona następujące metody pozwalające operować na pliku: dodaj rekord, odczytaj rekord, usuń rekord, modyfikuj rekord, reorganizuj plik, wyświetl wszystkie rekordy. SeqIndFile to klasa wysokopoziomowa, opakowująca funkcjonalności klas Database oraz IndexFile. Database odpowiada za operacje na plikach, w których znajdują się rekordy (główny obszar oraz overflow), natomiast IndexFile za obsługę pliku przechowującego indeksy (klucze pierwszych rekordów stron oraz numery stron). Fizyczne pliki zapisywane są w formacie .dat. Rekordy są w nich przechowywane w postaci bajtowej.

Rekordami przechowywanymi w pliku są studenci – reprezentowani przez numer indeksu oraz oceny z 3 kolejnych kolokwii. Plik dodatkowo rozszerza je o klucz (w postaci liczby naturalnej), flagę sygnalizującą czy rekord został usunięty (fizyczne usunięcie następuje dopiero po reorganizacji) oraz wskaźnik na następny rekord w overflow. Jeżeli następny rekord

pod względem klucza znajduje się w obszarze overflow, wskaźnik występuje w formie numer\_strony:offset\_rekordu, gdzie numer\_strony oznacza numer strony w overflow w której znajduje się rekord, a offset\_rekordu oznacza numer rekordu na stronie. Jeżeli rekord jest końcowym z listy tzn. następny rekord w kolejności znajduje się w głównym obszarze, pointer ustawiany jest na "x:x". Na początku pliku zawsze znajduje się rekord z najmniejszą możliwą wartością klucza, aby możliwe było dodanie rekordu z dowolnym kluczem – nie jest on jednak wypisywany, dopóki faktycznie nie zostanie dodany rekord o takim kluczu.

Indeksy rekordów przechowywane są w pamięci operacyjnej, co pozwala na redukcję liczby odczytów i zapisów stron dyskowych w klasie IndexFile. Indeksy są zapisywane do pliku jedynie na koniec działania programu, aby możliwa była weryfikacja poprawności działania programu.

Reorganizacja wywoływana jest po przekroczeniu ustawionego progu ilości stron w overflow. Zaimplementowana ona została zgodnie z pseudokodem zaprezentowanym na wykładzie.

Wszystkie wartości konfiguracyjne programu definiowane są z poziomu pliku konfiguracyjnego zapisanego w formacie JSON. Można w nim ustawić następujące parametry:

- Blocking factor - średnia ilość rekordów znajdujących się na stronie dyskowej
- Alfa – współczynnik określający ilość rekordów znajdujących się na stronie dyskowej po reorganizacji pliku
- Rozmiar rekordu
- Maksymalny klucz
- Początkowa liczba stron – liczba pustych stron stworzonych przy inicjalizacji pliku
- Symbol paddingu – symbol uzupełniający niepełne strony
- Maksymalna liczba stron w overflow – liczba stron w overflow przy których zostanie wykonana reorganizacja
- Liczba generowanych losowych rekordów
- Wypisuj wiadomości debugowe
- Wypisuj dodatkowe informacje o rekordach
- Wypisuj ilość operacji dyskowych
- Wypisuj zawartość pliku po każdej operacji

### 3. Specyfikacja formatu pliku testowego

Plik obsługuje dwie możliwości przyjmowania rekordów - generowanie losowe oraz wczytywanie danych z pliku testowego. Wybierając drugą opcję, plik musi zawierać operacje w oddzielnych liniach w jednym z następujących formatów:

**Dodanie rekordu:** A klucz numer\_indeksu ocena1 ocena2 ocena3

**Aktualizacja rekordu:** U klucz numer\_indeksu ocena1 ocena2 ocena3

**Usunięcie rekordu:** D klucz

Program wykonuje następnie operacje w kolejności występowania ich w pliku testowym.

### 4. Sposób prezentacji wyników działania programu

Po dodaniu wszystkich rekordów do pliku (zarówno w przypadku generowania rekordów losowo jak i odczytu z pliku) program wyświetla wszystkie rekordy w pliku posortowane według klucza. Rekordy prezentowane są w postaci rekordów przetrzymywanych w bazie danych tj. łącznie z kluczem, wskaźnikiem na następny rekord oraz flagą usunięcia. W przypadku ustawienia dodatkowych parametrów w pliku konfiguracyjnym, możliwe jest wypisywanie większej ilości informacji np. strony oraz offsetu każdego rekordu lub wyświetlania zawartości pliku po każdej pojedynczej operacji.

## 5. Przeprowadzenie eksperymentu

Na tak przygotowanym pliku został przeprowadzony eksperyment, w celu obserwacji wpływu parametrów implementacyjnych na optymalność działania. Eksperyment brał pod uwagę dwa parametry: współczynnik alfa oraz maksymalną ilość stron w obszarze overflow. Jako wartości oceniające optymalność programu wybrane zostały liczba operacji dyskowych (odczytu + zapisu) oraz czas działania w sekundach. Pomiary zostały wykonane dla 4 konfiguracji:

- Dodanie 10000 rekordów
- Dodanie 1000 rekordów
- Dodanie + aktualizacja 1000 rekordów
- Dodanie + usunięcie 1000 rekordów

Parametry interpretacyjne były iloczynem kartezjańskim następujących wartości:

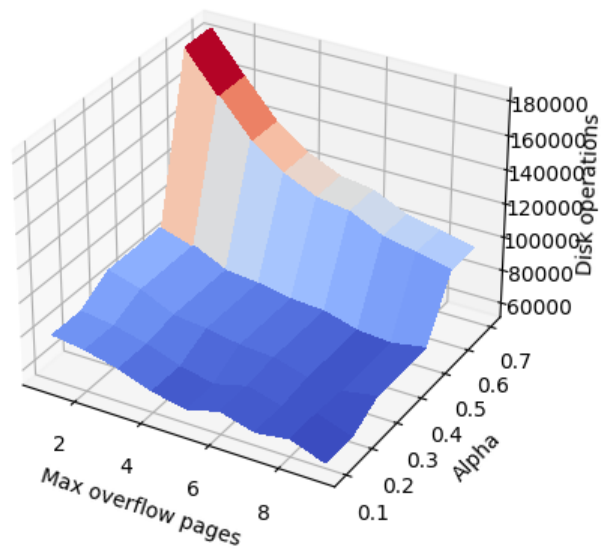
- Współczynnik alfa: 0.1, 0.2, ..., 0.7
- Maksymalna liczba stron w overflow: 1, 2, ..., 9

Współczynnik alfa nie przekracza wartości 0.7 ze względu na ustawiony współczynnik blokowania równy 4. Gdyby alfa wynosiła powyżej 0.75, reorganizacja w praktyce nie zmieniałaby faktycznego rozmieszczenia rekordów w pliku - wszystkie strony w obszarze głównym byłyby od razu zapełnione. Powoduje to znaczący wzrost działania programu, dlatego wartości alfy zostały ograniczone jedynie do mniejszych wartości. Rekordy były wczytywane z wcześniej przygotowanych plików testowych zawierających odpowiednie komendy. Wyniki eksperymentu zaprezentowane są poniżej.

### Dodanie 10000 rekordów

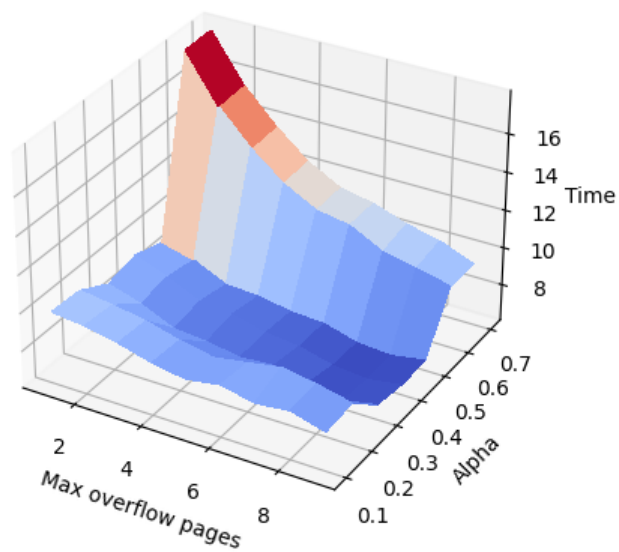
Operacje dyskowe:

		Maksymalna liczba stron w <u>overflow</u>								
Alfa		1	2	3	4	5	6	7	8	9
	0,1	78875	76016	71154	64051	59893	65038	59229	61820	54538
	0,2	78741	75869	70905	63787	59587	64660	58817	61324	53959
	0,3	89566	83621	74224	72147	68514	67104	61370	59799	63301
	0,4	89559	83614	74215	72138	68504	67093	61358	59787	63287
	0,5	89555	83609	74210	72132	68498	67086	61351	59779	63279
	0,6	184714	161578	139719	125444	116698	113777	104434	100590	96471
	0,7	184712	161576	139717	125441	116696	113774	104430	100587	96467



Czas działania:

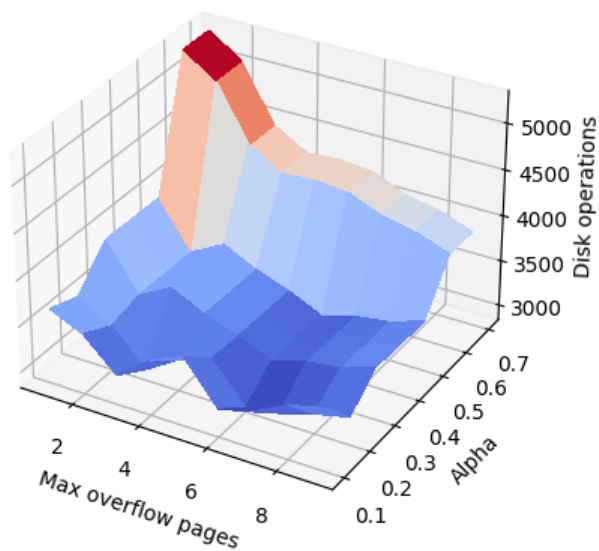
		Maksymalna liczba stron w overflow								
Alfa		1	2	3	4	5	6	7	8	9
	0,1	9,887	9,705	9,347	8,864	8,564	8,847	8,457	8,635	8,122
	0,2	9,785	9,534	9,233	8,758	8,405	8,691	8,987	8,687	8,279
	0,3	8,912	8,367	7,573	7,405	7,116	7,238	6,860	6,501	6,858
	0,4	9,256	8,616	7,812	7,663	7,183	6,973	6,510	6,339	6,595
	0,5	8,788	8,318	7,487	7,379	7,062	6,956	6,446	6,319	6,621
	0,6	17,954	15,470	13,704	12,297	11,362	11,065	10,147	9,781	9,359
	0,7	17,981	15,553	13,793	12,346	11,411	11,067	10,464	10,061	9,365



## Dodanie 1000 rekordów

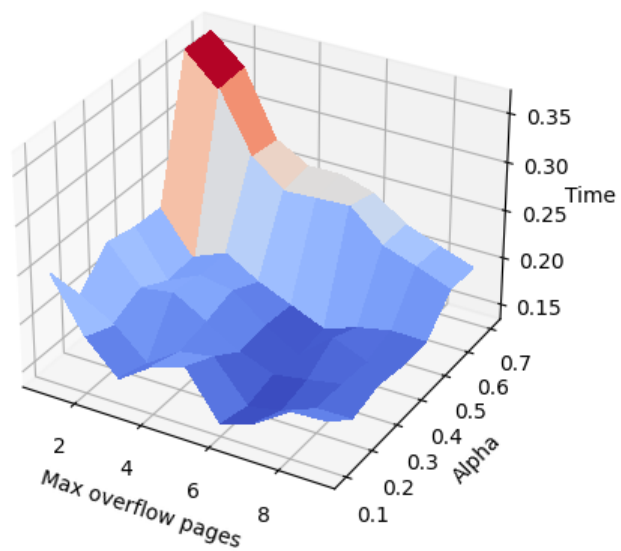
Operacje dyskowe:

		Maksymalna liczba stron w overflow								
Alfa		1	2	3	4	5	6	7	8	9
	0,1	3655	3494	3157	3432	3590	3141	3233	3426	3515
	0,2	3536	3320	2949	3174	3299	2883	2938	3052	3137
	0,3	3909	3402	3594	3417	3285	3119	3229	3280	3404
	0,4	3903	3395	3585	3408	3275	3108	3217	3268	3390
	0,5	3899	3390	3580	3402	3269	3101	3210	3260	3382
	0,6	5298	5057	4449	4235	4259	4226	4077	4002	3869
	0,7	5297	5055	4447	4232	4257	4223	4073	3999	3865



Czas działania:

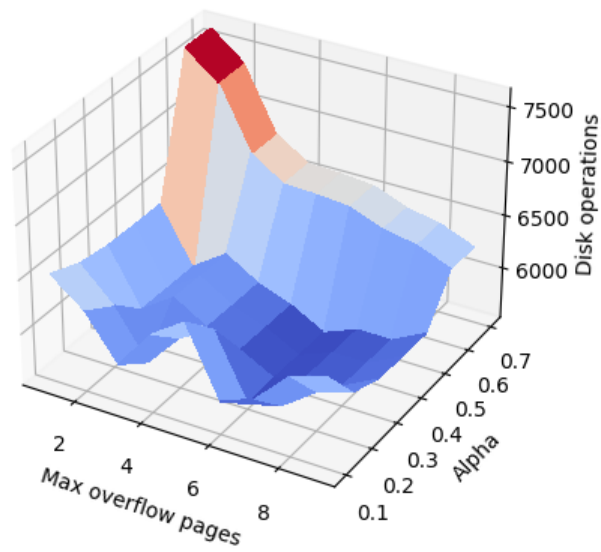
		Maksymalna liczba stron w overflow								
Alfa		1	2	3	4	5	6	7	8	9
	0,1	0,249	0,194	0,161	0,185	0,195	0,147	0,160	0,183	0,185
	0,2	0,201	0,181	0,150	0,174	0,179	0,139	0,145	0,159	0,160
	0,3	0,233	0,195	0,200	0,180	0,159	0,146	0,152	0,158	0,168
	0,4	0,226	0,179	0,195	0,189	0,162	0,141	0,171	0,156	0,163
	0,5	0,224	0,181	0,195	0,174	0,159	0,144	0,157	0,164	0,164
	0,6	0,369	0,337	0,276	0,248	0,250	0,252	0,220	0,208	0,194
	0,7	0,368	0,339	0,272	0,253	0,253	0,243	0,219	0,206	0,194



## Dodanie + aktualizacja 1000 rekordów

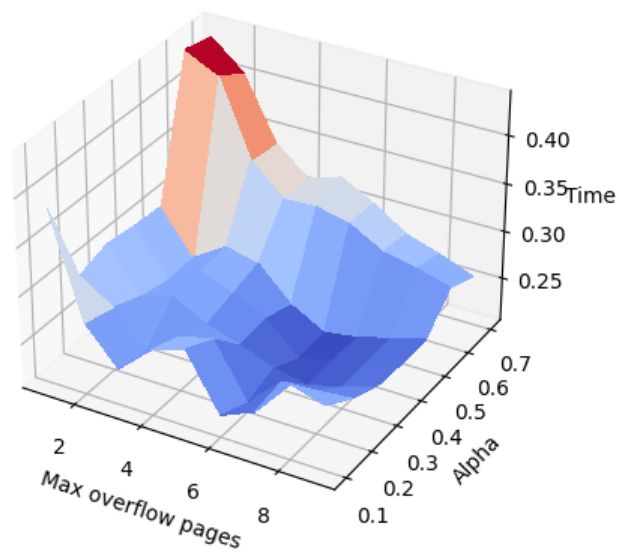
Operacje dyskowe:

		Maksymalna liczba stron w overflow								
Alfa		1	2	3	4	5	6	7	8	9
	0,1	6527	6286	5875	6254	6462	5832	5957	6266	6364
	0,2	6408	6112	5667	5996	6171	5574	5662	5892	5986
	0,3	6363	5849	6058	5869	5753	5584	5712	5709	5873
	0,4	6357	5842	6049	5860	5743	5573	5700	5697	5859
	0,5	6353	5837	6044	5854	5737	5566	5693	5689	5851
	0,6	7628	7384	6798	6596	6584	6553	6432	6371	6244
	0,7	7627	7382	6796	6593	6582	6550	6428	6368	6240



Czas działania:

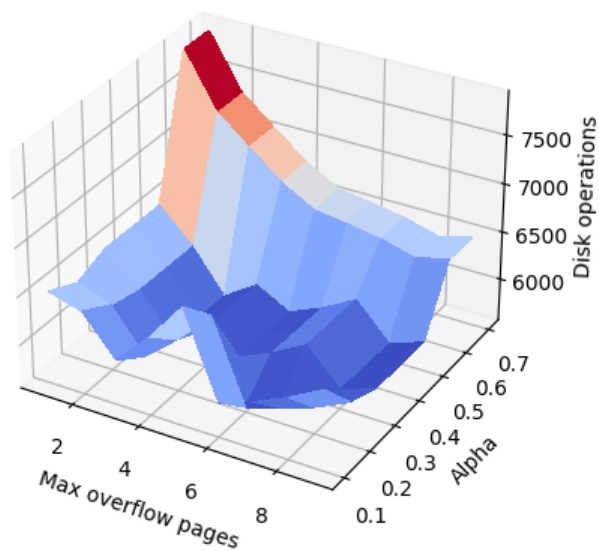
		Maksymalna liczba stron w <u>overflow</u>								
		1	2	3	4	5	6	7	8	9
Alfa	0,1	0,387	0,280	0,243	0,274	0,285	0,228	0,243	0,283	0,272
	0,2	0,293	0,271	0,236	0,262	0,270	0,214	0,248	0,250	0,253
	0,3	0,304	0,258	0,268	0,249	0,238	0,224	0,218	0,227	0,240
	0,4	0,300	0,251	0,271	0,258	0,241	0,210	0,220	0,228	0,238
	0,5	0,297	0,252	0,273	0,266	0,232	0,213	0,218	0,228	0,238
	0,6	0,441	0,424	0,343	0,312	0,313	0,305	0,285	0,278	0,268
	0,7	0,434	0,405	0,340	0,311	0,322	0,304	0,281	0,270	0,256



## Dodanie + usunięcie 1000 rekordów

Operacje dyskowe:

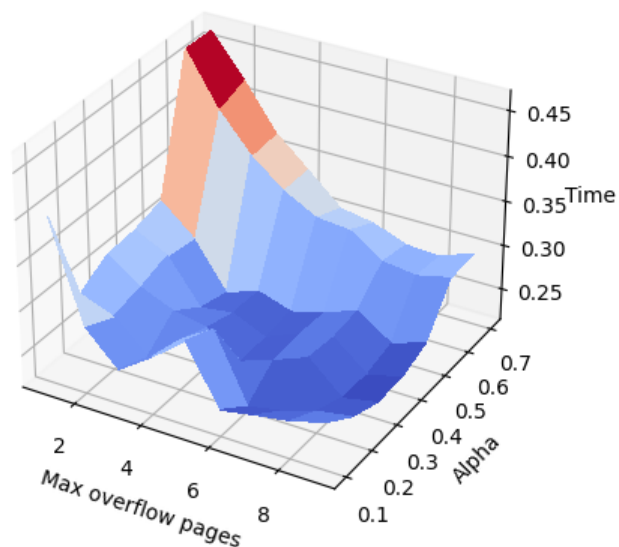
		Maksymalna liczba stron w overflow								
Alfa		1	2	3	4	5	6	7	8	9
	0,1	6524	6423	6025	6371	6784	5924	6043	6199	6390
	0,2	6374	6245	5813	6102	6452	5609	5721	5838	6006
	0,3	6498	6175	5728	5953	5769	5923	6113	5755	5884
	0,4	6492	6168	5719	5944	5759	5912	6101	5743	5870
	0,5	6488	6163	5714	5938	5753	5905	6094	5735	5862
	0,6	7903	7351	7070	6796	6603	6556	6495	6403	6489
	0,7	7902	7349	7068	6793	6601	6553	6491	6400	6485



Czas działania:

		Maksymalna liczba stron w overflow								
Alfa		1	2	3	4	5	6	7	8	9
	0,1	0,400	0,291	0,254	0,280	0,309	0,245	0,251	0,257	0,268
	0,2	0,289	0,274	0,239	0,271	0,295	0,220	0,227	0,239	0,245
	0,3	0,308	0,281	0,265	0,258	0,231	0,244	0,259	0,226	0,233
	0,4	0,310	0,292	0,243	0,248	0,231	0,247	0,255	0,224	0,233
	0,5	0,319	0,287	0,234	0,247	0,235	0,245	0,259	0,233	0,245
	0,6	0,467	0,410	0,368	0,338	0,318	0,314	0,289	0,284	0,293
	0,7	0,466	0,421	0,370	0,338	0,304	0,298	0,283	0,281	0,295





Wyniki eksperymentu pokazują, że parametry implementacyjne pliku sekwencyjno-indeksowego jednakowo wpływają na liczbę potrzebnych operacji dyskowych oraz czas działania, niezależnie od rodzaju wykonywanych operacji (dodawanie, aktualizowanie, usuwanie). Czas działania programu jest wprost proporcjonalny do ilości potrzebnych operacji dyskowych. Jak można zauważyć, przy stałej ilości maksymalnej liczby stron w overflow, liczba potrzebnych operacji dyskowych maleje wraz ze wzrostem współczynnika alfa, aby potem zacząć ponownie rosnąć. W zależności od rodzajów wykonywanych operacji, wartość współczynnika alfa dla którego zaczyna następować wzrost, zmienia się. Przy projektowaniu struktury baz danych trzeba także brać pod uwagę czas działania operacji, który drastycznie wzrasta dla dużego współczynnika alfa, zwłaszcza w połączeniu z niewielką liczbą dostępnych stron w overflow. Dlatego z wyników eksperymentu wywnioskować można, że najlepsze działanie pliku osiągnąć można przy wyborze współczynnika alfa ok. 0.3-0.5 w połączeniu z dość dużą liczbą stron w overflow. Gwarantuje to relatywnie niską liczbę potrzebnych operacji dyskowych przy jednoczesnym krótkim czasie działania.