Implementacja i analiza indeksowo-sekwencyjnej organizacji plików

Wojciech Trapkowski

8 grudnia 2024

1 Wprowadzenie

Celem projektu jest implementacja i analiza indeksowo-sekwencyjnej organizacji plików (ISAM - Indexed Sequential Access Method), która łączy zalety dostępu sekwencyjnego i bezpośredniego do danych. Metoda ta została opracowana przez IBM w latach 60-tych i do dziś stanowi podstawę wielu systemów bazodanowych.

Podstawowe założenia tej organizacji plików obejmują:

- Przechowywanie danych w uporządkowanej sekwencji rekordów, pogrupowanych w strony o stałym rozmiarze
- Utrzymywanie oddzielnego pliku indeksowego, zawierającego klucze i wskaźniki do odpowiadających im stron w pliku głównym
- Wykorzystanie obszaru przepełnień do obsługi nowych rekordów, których nie można umieścić w pierwotnie przydzielonych stronach
- Okresową reorganizację pliku w celu optymalizacji jego struktury

Organizacja ISAM zapewnia wydajne operacje wyszukiwania dzięki indeksom, zachowując jednocześnie możliwość sekwencyjnego przetwarzania danych. Jest szczególnie efektywna w systemach, gdzie stosunek operacji odczytu do zapisu jest wysoki, a dane są względnie statyczne.

2 Struktury plików

- Struktura pliku indeksowego zawiera strony indeksowe, gdzie każdy wpis składa się z klucza początkowego oraz wskaźnika do odpowiedniej strony w pliku głównym.
- Struktura pliku głównego składa się z nagłówka zawierającego liczbę stron oraz wskaźnik do obszaru przepełnień (strażnika), a następnie sekwencji stron zawierających rekordy.
- Obszar przepełnień służy do przechowywania rekordów, które nie mogą być umieszczone w pierwotnie przydzielonych stronach. Każdy rekord w obszarze głównym może wskazywać na dodatkowe rekordy w obszarze przepełnień.
- Organizacja rekordów w stronach opiera się na strukturze Page, która zawiera stałą liczbę wpisów. Każdy wpis zawiera klucz, wartość (PESEL), wskaźnik do obszaru przepełnień oraz flagę usunięcia.

3 Szczegóły implementacyjne

3.1 Buforowanie w pamięci operacyjnej

- Mechanizm buforowania zaimplementowany jest w klasie PageBuffer, która wykorzystuje inteligentne wskaźniki do zarządzania stronami w pamięci.
- Wielkość bufora jest określona przez stałą, która definiuje maksymalną liczbę stron przechowywanych jednocześnie w pamięci.
- Strategia zastępowania stron opiera się na liczbie referencji do strony - usuwane są strony z pojedynczą referencją. Przed usunięciem strony z bufora, jej zawartość jest zapisywana na dysk.
- System śledzi liczbę operacji odczytu i zapisu poprzez liczniki.

3.2 Parametry implementacyjne

- Rozmiar strony (PAGE_SIZE) jest stałą określającą liczbę rekordów w pojedynczej stronie.

- Współczynnik wypełnienia (α) określa maksymalną liczbę rekordów w stronie po reorganizacji.
- Współczynnik obszaru przepełnień (β) definiuje stosunek rozmiaru obszaru przepełnień do obszaru głównego
- Reorganizacja jest wykonywana gdy liczba rekordów w obszarze przepełnień przekroczy ustalony próg (γ).

4 Format pliku testowego

4.1 Struktura rekordu

W implementacji rekord jest reprezentowany jako pojedyncza liczba całkowita typu uint64_t, przechowująca numer PESEL.

5 Prezentacja wyników

5.1 Interfejs użytkownika

Program oferuje interaktywny interfejs wiersza poleceń oraz możliwość wykonywania komend z pliku. Dostępne są następujące tryby pracy:

- Tryb interaktywny - oznaczony znakiem zachęty ¿
- Tryb wsadowy - wykonywanie komend z pliku

5.2 Dostępne komendy

Program obsługuje następujące polecenia:

- `insert <klucz> <wartość>` - wstawia nowy rekord
- `update <klucz> <wartość>` - aktualizuje istniejący rekord
- `search <klucz>` - wyszukuje rekord o podanym kluczu
- `remove <klucz>` - usuwa rekord o podanym kluczu
- `print` - wyświetla zawartość całej bazy danych

- `print_stats` - wyświetla statystyki (liczba operacji I/O)
- `generate <liczba_kluczy>` - generuje zadaną liczbę losowych rekordów
- `reorganise` - wymusza reorganizację struktury
- `flush` - wymusza zapis buforowanych danych na dysk
- `help` - wyświetla listę dostępnych komend
- `exit`/`quit` - kończy działanie programu

5.3 Format wyświetlania

- Wyniki wyszukiwania są wyświetlane w formacie: wartość rekordu lub komunikat "Not found"dla nieznalezionych kluczy
- Błędy operacji są sygnalizowane odpowiednimi komunikatami
- Statystyki pokazują liczbę operacji odczytu i zapisu wykonanych na dysku

6 Eksperymenty

6.1 Metodologia

- Przeprowadzone testy obejmowały analizę wydajności programu poprzez żonglowanie wszystkimi kluczowymi parametrami:
  - Rozmiar strony (współczynnik blokowania pliku)
  - Rozmiar bufora przechowującego strony w pamięci operacyjnej
  - Liczba rekordów w bazie danych
  - Współczynnik wypełnienia strony (α)
  - Współczynnik rozmiaru obszaru przepełnień (β)
  - Próg reorganizacji (γ)
- Mierzone metryki obejmowały:

  - Liczba operacji dyskowych przy:
    * Usuwaniu rekordów
    * Aktualizacji rekordów
    * Wyszukiwaniu rekordów
    * Wstawianiu nowych rekordów
    * Reorganizacji struktury
  - Zużycie pamięci przez pliki bazy danych

- Metodyka pomiarów:
  - Każda operacja była wykonywana na świeżo zainicjalizowanej bazie danych
  - Liczniki operacji I/O były zerowane przed każdą operacją
  - Pomiary były agregowane dla serii identycznych operacji w celu uzyskania średnich wartości
  - Testy przeprowadzano dla różnych kombinacji parametrów, aby zbadać ich wzajemny wpływ

6.2 Wyniki

Tabela 1 przedstawia wyniki przeprowadzonych testów wydajnościowych. Poszczególne kolumny reprezentują:

- Kolumny operacji dyskowych (I/O):
  - A - Liczba operacji dyskowych przy usuwaniu rekordów
  - B - Liczba operacji dyskowych przy aktualizacji istniejących rekordów
  - C - Liczba operacji dyskowych przy wyszukiwaniu rekordów
  - D - Liczba operacji dyskowych przy wstawianiu nowych rekordów
  - E - Liczba operacji dyskowych podczas reorganizacji struktury

- Parametry pamięciowe:
  - F - Całkowita pamięć zajmowana przez pliki bazy danych (KB)
  - G - Współczynnik blokowania pliku (rozmiar strony)

  - H - Rozmiar bufora przechowującego strony w pamięci operacyjnej

- Parametry konfiguracyjne:
  - I - Liczba rekordów w bazie danych
  - J - Współczynnik wypełnienia strony (α)
  - K - Współczynnik rozmiaru obszaru przepełnień (β)
  - L - Próg reorganizacji (γ)

A	B	C	D	E	F	G	H	I	J	K	L
0	0	0	2	998	82	4	4	1,000	0.5	0.13	0.5
0	0	0	2	492	72	8	4	1,000	0.5	0.13	0.5
0	0	4	2	244	66	16	4	1,000	0.5	0.13	0.5
0	0	0	0	1,476	659	16	512	10,000	0.5	0.13	0.5
0	0	2	0	644	440	16	512	10,000	0.75	0.13	0.5
0	0	2	0	644	440	16	512	10,000	0.75	0.13	0.5
0	0	0	230	1,104	842	16	512	10,000	0.75	0.5	0.5
0	0	0	0	484	674	16	512	10,000	0.75	0.5	1
0	0	0	2	228	333	16	512	10,000	1	0.5	1
0	0	2	230	228	549	16	512	10,000	1	1	1

7 Wnioski

- Wpływ rozmiaru strony:
  - Większy rozmiar strony znacząco redukuje liczbę operacji I/O
  - Jednak zbyt duży rozmiar strony może prowadzić do nieefektywnego wykorzystania pamięci
  - Optymalny rozmiar strony zależy od charakterystyki danych i częstotliwości operacji
- Wpływ rozmiaru bufora:
  - Większy bufor stron znacząco zmniejsza liczbę fizycznych operacji I/O
  - Szczególnie efektywny przy operacjach wyszukiwania i aktualizacji
  - Należy znaleźć kompromis między wydajnością a zużyciem pamięci operacyjnej

- Wpływ współczynnika α (wypełnienie strony):
  - Wartości w przedziale 0.5 - 0.75 okazały się optymalne
  - Niższe wartości α skutkują większym rozmiarem plików
  - Wyższe wartości α zwiększają częstotliwość reorganizacji i liczbę operacji przy wstawianiu
- Wpływ współczynnika β (rozmiar obszaru przepełnień):
  - Silnie wpływa na całkowite zużycie pamięci
  - Mniejsze wartości β są zalecane dla optymalizacji przestrzeni
  - Zbyt niski β może prowadzić do częstszych reorganizacji
- Wpływ współczynnika γ (próg reorganizacji):
  - Niższe wartości γ prowadzą do częstszych reorganizacji, ale utrzymują strukturę w lepszej kondycji
  - Wyższe wartości γ zmniejszają częstotliwość reorganizacji, ale mogą prowadzić do degradacji wydajności wyszukiwania
  - Optymalny γ zależy od proporcji operacji odczytu do zapisu
- Ograniczenia metody:
  - Koszt reorganizacji może być znaczący dla dużych zbiorów danych
  - Wymaga odpowiedniego dostrojenia parametrów do konkretnego przypadku użycia