Struktury baz danych

Projekt 1 – sortowanie z użyciem wielkich buforów

Mateusz Kowalczyk, s188717

1. Wprowadzenie

Projekt miał na celu zaimplementowanie oraz przetestowanie jednego z algorytmów sortowania plików na dysku. Wybrano metodę sortowania z użyciem wielkich buforów.

Po przyjęciu następujących oznaczeń:

N – liczba rekordów w pliku wejściowym,
b – rozmiar bufora (wyrażony w liczbie rekordów), bufor ma rozmiar strony dyskowej,
n– liczba dostępnych buforów,

działanie głównej części programu – sortowania pliku – można w uproszczony sposób opisać następującymi krokami:

1. odczytuj rekordy z pliku wejściowego do wszystkich buforów i, używając sortowania przez kopcowanie, twórz w połączonych buforach serie o rozmiarze nb rekordów, a następnie dystrybuuj je kolejno na $n-1$ pierwszych taśm, aż osiągnięty zostanie koniec pliku wejściowego;
2. używając kopca minimalnego, scalaj pierwsze serie ze wszystkich $n-1$ pierwszych taśm, zapisując wynik na n-tej taśmie, aż osiągnięte zostaną końce wszystkich $n-1$ pierwszych taśm;
3. rozdystrybuuj wszystkie serie z n-tej taśmy kolejno na $n-1$ pierwszych taśm;
4. powtarzaj kroki 2–3, aż w kroku 3. dystrybucja zostanie wykonana na tylko jedną taśmę;
5. przenieś zawartość n-tej taśmy do pliku wejściowego.

Przyjęto następujące wartości liczbowe:

$b = 16$,
$n = 8$.

Ponieważ każda taśma wymaga przydzielenia jej bufora do operacji zapisu lub odczytu z dysku, używanych jest 8 taśm. W tej liczbie nie jest uwzględniony sam plik wejściowy. Na potrzeby odczytu go (w kroku 1.) oraz skopiowania do niego wyniku sortowania z n-tej taśmy (w kroku 5.) przydzielany jest mu pierwszy bufor, gdyż wówczas nie jest on używany przez pozostałe taśmy.

2. Format plików

Rekord posiada następujący format:

- bajty 0–1: klucz (obwód pięciokąta) – 2-bajtowa liczba bez znaku stanowiąca sumę jednobajtowych długości boków,
- bajty 2–6: długości boków – 5 jednobajtowych liczb bez znaku stanowiących długości boków pięciokąta.

Plik wejściowy oraz taśmy używane do sortowania stanowią wyłącznie ciąg rekordów (bez dodatkowych oznaczeń).

3. Sposób prezentacji wyników

Wynik działania programu przedstawiony zostaje na ekranie konsoli. Wyświetlony tekst zawiera:

- zawartość pliku wejściowego przed posortowaniem, każdy rekord w osobnym wierszu w postaci czytelnej dla człowieka (numer rekordu, obwód pięciokąta, długości boków pięciokąta)

```
Records before sorting:
0: [perimeter: 607] sides lengths: 38 39 246 133 151
1: [perimeter: 581] sides lengths: 21 173 29 210 148
2: [perimeter: 617] sides lengths: 221 196 118 25 57
3: [perimeter: 732] sides lengths: 49 241 173 181 88
4: [perimeter: 613] sides lengths: 240 147 151 50 25
5: [perimeter: 719] sides lengths: 43 209 192 253 22
6: [perimeter: 458] sides lengths: 142 78 72 155 11
7: [perimeter: 644] sides lengths: 245 59 73 168 99
8: [perimeter: 710] sides lengths: 93 222 63 223 109
9: [perimeter: 743] sides lengths: 104 180 135 154 170
10: [perimeter: 933] sides lengths: 205 220 247 193 68
```

- kolejność rekordów po każdej fazie sortowania, jeżeli użytkownik wybrał opcję wyświetlania jej

```
Records after a phase:
0: [perimeter: 219] sides lengths: 32 16 90 25 56
1: [perimeter: 240] sides lengths: 65 65 10 62 38
2: [perimeter: 242] sides lengths: 42 6 88 29 77
3: [perimeter: 257] sides lengths: 52 73 24 51 57
4: [perimeter: 264] sides lengths: 9 93 8 65 89
5: [perimeter: 269] sides lengths: 129 41 8 27 64
6: [perimeter: 276] sides lengths: 18 102 72 54 30
7: [perimeter: 276] sides lengths: 64 54 24 18 116
8: [perimeter: 282] sides lengths: 56 1 54 91 80
9: [perimeter: 292] sides lengths: 39 51 9 135 58
10: [perimeter: 302] sides lengths: 23 141 54 12 72
```

- zawartość pliku wejściowego po posortowaniu

```
Records after sorting:
0: [perimeter: 219] sides lengths: 32 16 90 25 56
1: [perimeter: 240] sides lengths: 65 65 10 62 38
2: [perimeter: 241] sides lengths: 82 27 21 29 82
3: [perimeter: 242] sides lengths: 42 6 88 29 77
4: [perimeter: 249] sides lengths: 29 122 4 53 41
5: [perimeter: 257] sides lengths: 52 73 24 51 57
6: [perimeter: 264] sides lengths: 9 93 8 65 89
7: [perimeter: 269] sides lengths: 129 41 8 27 64
8: [perimeter: 273] sides lengths: 78 64 8 7 116
9: [perimeter: 276] sides lengths: 18 102 72 54 30
10: [perimeter: 276] sides lengths: 64 54 24 18 116
```

- liczbę faz sortowania, zapisów na dysk i odczytów z dysku

```
phases number: 4
disk writes number: 1800
disk reads number: 1800
```

4. Eksperyment

Przeprowadzono eksperyment w celu pomiaru wartości liczbowych: liczby faz sortowania, liczby zapisów na dysk i liczby odczytów z dysku. Dla każdej z 5 wartości N przeprowadzono 3 próby. W każdej z nich rekordy wygenerowano losowo, posortowano i odczytano wyniki zaprezentowane przez program. Następnie obliczono wartości średnie, które przedstawiają się następująco:

- liczba faz sortowania: p_i
- liczba operacji dyskowych: d_i

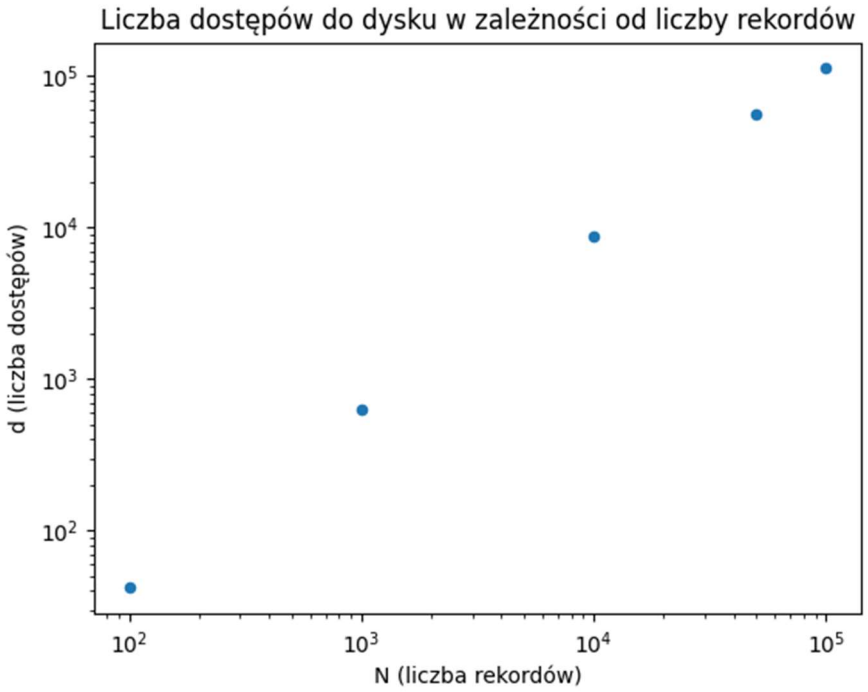
$N_1 = 100$
$p_1 = 1$
$d_1 = 42$

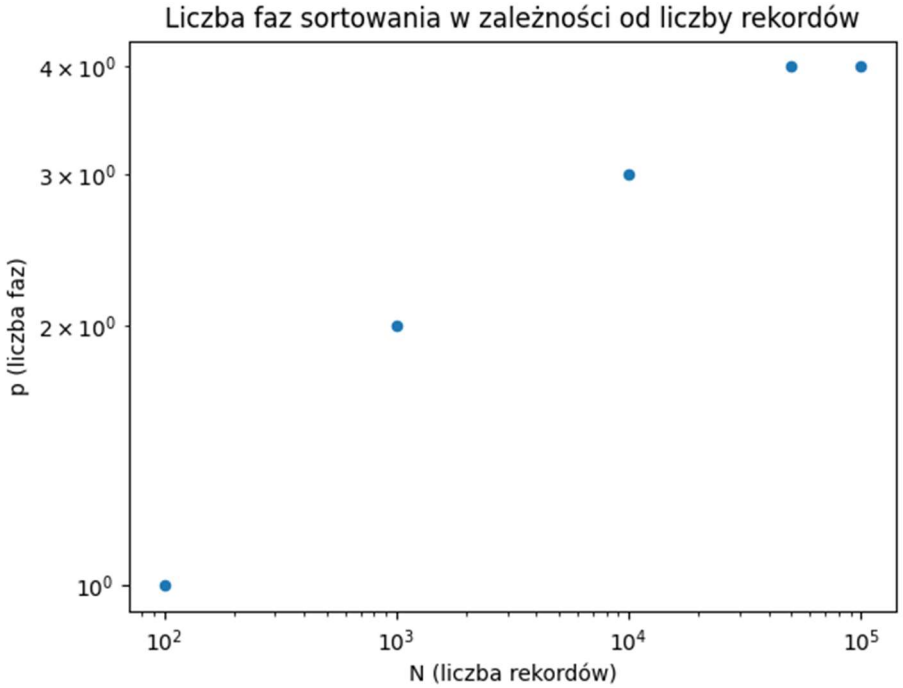
$N_2 = 1\,000$
$p_2 = 2$
$d_2 = 630$

$N_3 = 10\,000$
$p_3 = 3$
$d_3 = 8\,750$

$N_4 = 50\,000$
$p_4 = 4$
$d_4 = 56\,250$

$N_5 = 100\,00$
$p_5 = 4$
$d_5 = 112\,500.$





Wyniki teoretyczne prezentują się następująco:

- liczba faz sortowania: $p_{iteor} = \lceil log_n \frac{N_i}{b} \rceil - 1$,
- liczba operacji dyskowych: $d_{iteor} = 2 \cdot \frac{N_i}{b} log_n \frac{N_i}{b}$.

$N_1 = 100$

$p_{1\,teor} = \left\lceil \log_8 \frac{100}{16} \right\rceil - 1 = 0$

$d_{1\,teor} = 2 \cdot \frac{100}{16} \log_8 \frac{100}{16} \approx 11$

$N_2 = 1\,000$

$p_{2\,teor} = \left\lceil \log_8 \frac{1\,000}{16} \right\rceil - 1 = 1$

$d_{2\,teor} = 2 \cdot \frac{1\,000}{16} \log_8 \frac{1\,000}{16} \approx 249$

$N_3 = 10\,000$

$p_{3\,teor} = \left\lceil \log_8 \frac{10\,000}{16} \right\rceil - 1 = 3$

$d_{3\,teor} = 2 \cdot \frac{10\,000}{16} \log_8 \frac{10\,000}{16} \approx 3\,870$

$N_4 = 50\,000$

$p_{4\,teor} = \left\lceil \log_8 \frac{50\,000}{16} \right\rceil - 1 = 3$

$d_{4\,teor} = 2 \cdot \frac{50\,000}{16} \log_8 \frac{50\,000}{16} \approx 24\,187$

$N_5 = 100\,000$

$p_{5\,teor} = \left\lceil \log_8 \frac{100\,000}{16} \right\rceil - 1 = 4$

$d_{5\,teor} = 2 \cdot \frac{100\,000}{16} \log_8 \frac{100\,000}{16} \approx 52\,540.$

Liczba faz sortowania obliczona nie odbiega wyraźnie od tej otrzymanej w wyniku eksperymentu. W każdym przypadku różnica wynosi co najwyżej 1.

Dla liczby dostępów do dysku obliczono błąd względny:

$\Delta d_i = \frac{d_i - d_{iteor}}{d_{iteor}}$

$\Delta d_1 = \frac{42 - 11}{11} \approx 2{,}82$

$$\Delta d_2 = \frac{630 - 249}{249} \approx 1{,}53$$

$$\Delta d_3 = \frac{8750 - 3\,870}{3870} \approx 1{,}26$$

$$\Delta d_4 = \frac{56\,250 - 24\,187}{24187} \approx 1{,}33$$

$$\Delta d_5 = \frac{112\,500 - 52\,540}{52540} \approx 1{,}14$$

Podsumowanie wyników:

N_i	d_i	d_{iteor}	Δd_i
100	42	11	2,82
1 000	630	249	1,53
10 000	8 750	3 870	1,26
50 000	56 250	24 187	1,33
100 000	112 500	52 540	1,14

Powyższy błąd względny może wynikać z różnic w sposobie scalania długich serii. W projekcie użyto schematu $(n-1)+1$ (scalanie z $(n-1)$ taśm na 1 taśmę), natomiast wzór może opisywać algorytm $(n-1)+(n-1)$ (scalanie z $(n-1)$ taśm na $(n-1)$ taśm) wymagający dwukrotnie mniej operacji dyskowych podczas scalania.