# **Zadanie 1**

**A)**

1. **Przed usunięciem klucza głównego:**
   * **Czas wykonania:** 50 ms
   * **Koszty operacji I/O:** Niskie
   * **Koszty CPU:** Niskie
   * **Ilość wczytanych wierszy:** Minimalna, dzięki indeksowi klucza głównego **(13)**
2. **Po usunięciu klucza głównego (tabela jako sterta):**
   * **Czas wykonania:** Dłuższy niż wcześniej
   * **Koszty operacji I/O:** Wyższe, brak optymalizacji indeksowej **(600 razy większy)**
   * **Koszty CPU:** Wyższe, brak optymalizacji indeksowej **(1330 razy większy)**
   * **Ilość wczytanych wierszy:** W przypadku sterty wszystkie wiersze muszą być przeszukane, co prowadzi do większej liczby wczytanych wierszy. **(121317)**

Różnice wynikają z braku indeksu klucza głównego po usunięciu go. Indeksy są kluczowe dla efektywnego przeszukiwania danych, a ich brak prowadzi do zwiększenia kosztów operacji I/O i CPU.

**B)**

W przypadku zmiany słowa AND na OR w warunkach zapytania, może dojść do zmiany planu wykonania. Zazwyczaj zapytania z OR są mniej efektywne niż te z AND, ponieważ OR wymaga sprawdzenia obu warunków niezależnie od tego, czy pierwszy warunek jest spełniony.

Koszty operacji I/O i CPU mogą wzrosnąć w przypadku OR, ponieważ wymaga to dodatkowych operacji logicznych.

Zmiana słowa AND na OR może wpłynąć na efektywność zapytania, ale dokładna analiza będzie zależała od specyfiki danych i indeksów w danej tabeli.

# **Zadanie 2**

1. **Przed stworzeniem indeksu:**
   * Plan wykonania: Skan ciągły tabeli (Table Scan)
   * Czas wykonania operacji: Wysoki
   * Koszty operacji I/O: Wysokie
   * Koszty CPU: Wysokie
2. **Po stworzeniu indeksu:**
   * Plan wykonania: Wyszukiwanie z wykorzystaniem indeksu (Index Seek)
   * Czas wykonania operacji: Niski
   * Koszty operacji I/O: Niskie
   * Koszty CPU: Niskie

Stworzenie indeksu na kolumnie używanej w klauzuli WHERE pozwala na efektywne wyszukiwanie, co przekłada się na znaczną poprawę wydajności zapytania.