Практическая работа 23/24

Сергушов Павел ПМ22-4

1. Создайте индекс на таблице рейсов (flights) по аэропортам прибытия (arrival\_airport). Найдите все рейсы в Москву (с помощью соединения flights и airports), создайте и опишите (объясните) план выполнения запроса. Какое

соединение использовал планировщик? Оцените вычислительную сложность алгоритма соединения (здесь и далее – для данного конкретного случая количеств строк).

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Для каждой строки из первой таблицы (airports\_data ml) выполняется поиск соответствующих строк во второй таблице (flights f) на основе условия соединения.

2. Выполните декартово произведение двух любых таблиц. Какой способ соединения будет выбран планировщиком?

Опишите план запроса. Оцените вычислительную сложность алгоритма соединения.

Изображение выглядит как текст, программное обеспечение, Шрифт, число

Автоматически созданное описание

Изображение выглядит как текст, программное обеспечение, Шрифт, число

Автоматически созданное описание

Изображение выглядит как текст, Шрифт, число, линия

Автоматически созданное описание

3. Напишите запрос, который выводит наименования аэропортов отправления и прибытия для всех рейсов. Создайте и

опишите план запроса (следует соединять flights с airports). Какой способ соединения выбрал планировщик?

Проверьте, хватило ли оперативной памяти для размещения хеш-таблиц. Оцените вычислительную сложность

алгоритма соединения.

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Изображение выглядит как текст, Шрифт, снимок экрана, число

Автоматически созданное описание

Планировщик выбрал для соединения таблиц метод Hash Left Join

4. Напишите запрос, который выводит общее количество занятых мест для всех рейсов. Какой способ соединения

выбрал планировщик? Проверьте, хватило ли оперативной памяти для размещения хеш-таблиц. Оцените

вычислительную сложность алгоритма соединения.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Планировщик выбрал последовательное сканирование (Seq Scan) таблицы tickets для выполнения запроса.

5. Напишите запрос, который для каждого рейса показывает дату бронирования, имя пассажира, номер билета,

стоимость билета (номер рейса тоже нужно показывать). Такой запрос должен соединять 4 таблицы: бронирования

(bookings), билеты (tickets), перелеты (ticket\_flights) и рейсы (flights). Проанализируйте план запроса; объясните, в

каком порядке выполняются операции. Оцените вычислительную сложность алгоритма соединения.

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Сначала выполняется Hash Join между таблицами ticket\_flights. и flights. Это происходит на основе условия соединения ticket\_flights.flight\_id = flights.flight\_id.

Затем выполняется Hash Join между результатом первого шага и таблицей tickets. Условие соединения - ticket\_flights.ticket\_no = ticket.ticket\_no.

Далее происходит Hash Join между результатом второго шага и таблицей bookings. Условие соединения - tickets.book\_ref = bookings.book\_ref.

выполняется Seq Scan на таблице flights, что означает полное сканирование таблицы без использования индексов.

6. Получите план запроса, в котором соединяются tickets, ticket\_flights и boarding\_passes, а результат сортируется по

номеру билета. Объясните порядок выполнения запроса.

Изображение выглядит как текст, снимок экрана, программное обеспечение, число

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Данные из таблиц tickets и tickets\_flights объединяются по условию tickets .ticket\_no = tickets\_flights.ticket\_no с помощью операции Merge Join

Затем данные из предыдущего объединения объединяются с данными из таблицы ticket\_flights по условию tickets.flight\_id = ticket\_flights flight\_id также с помощью операции Merge Join.

7. Получите планы запросов, которые выводят отсортированные по каким-либо столбцам (в одном запросе – по одному

cтолбцу, в другом – по-другому, в третьем – по всем выбранным для сортировки столбцам) строки из flights,

ticket\_flights, airports, tickets. Какие методы сортировки использовались? Объясните результаты.

Изображение выглядит как текст, снимок экрана, программное обеспечение, число

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, программное обеспечение, число

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

**Запрос 3: Вывести отсортированные по всем выбранным для сортировки столбцам строки из аэропортов**

Изображение выглядит как текст, снимок экрана, число, программное обеспечение

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание