УНИВЕРСИТЕТ ИТМО

Факультет программной инженерии и компьютерной техники

Направление подготовки 09.03.04 Программная инженерия

Дисциплина «Информационный системы и базы данных»

Лабораторная работа №4

Вариант 1527

Студент

Макаров Н. М.

P33111

Преподаватель

Харитонова А. Е.

Санкт-Петербург, 2022 г.

Составить запросы на языке SQL (пункты 1-2).

Для каждого запроса предложить индексы, добавление которых уменьшит время выполнения запроса (указать таблицы/атрибуты, для которых нужно добавить индексы, написать тип индекса; объяснить, почему добавление индекса будет полезным для данного запроса).

Для запросов 1-2 необходимо составить возможные планы выполнения запросов. Планы составляются на основании предположения, что в таблицах отсутствуют индексы. Из составленных планов необходимо выбрать оптимальный и объяснить свой выбор.

Изменятся ли планы при добавлении индекса и как?

Для запросов 1-2 необходимо добавить в отчет вывод команды EXPLAIN ANALYZE [запрос]

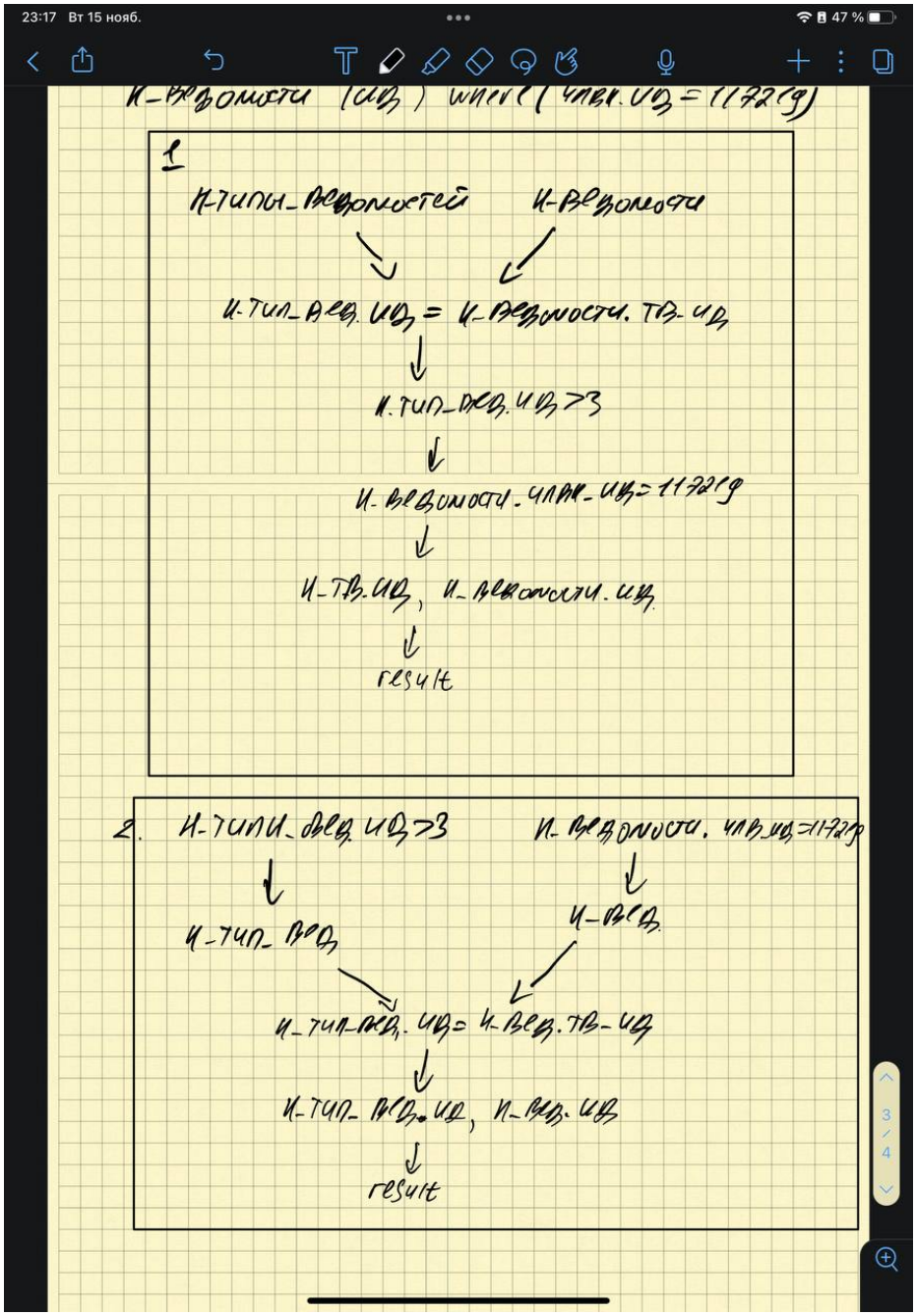
Подробные ответы на все вышеперечисленные вопросы должны присутствовать в отчете (планы выполнения запросов должны быть нарисованы, ответы на вопросы - представлены в текстовом виде).

1. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:
   H_ТИПЫ_ВЕДОМОСТЕЙ, H_ВЕДОМОСТИ.
   Вывести атрибуты: H_ТИПЫ_ВЕДОМОСТЕЙ.ИД, H_ВЕДОМОСТИ.ИД.
   Фильтры (AND):
   a) H_ТИПЫ_ВЕДОМОСТЕЙ.ИД > 3.
   b) H_ВЕДОМОСТИ.ЧЛВК_ИД = 117219.
   Вид соединения: LEFT JOIN.
2. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:
   Таблицы: H_ЛЮДИ, H_ОБУЧЕНИЯ, H_УЧЕНИКИ.
   Вывести атрибуты: H_ЛЮДИ.ОТЧЕСТВО, H_ОБУЧЕНИЯ.ЧЛВК_ИД, H_УЧЕНИКИ.ГРУППА.
   Фильтры: (AND)
   a) H_ЛЮДИ.ИД > 163484.
   b) H_ОБУЧЕНИЯ.НЗК = 999080.
   c) H_УЧЕНИКИ.ИД > 150308.
   Вид соединения: RIGHT JOIN.

Запрос

```
select nv."ИД", "Н_ТИПЫ_ВЕДОМОСТЕЙ"
from "Н_ВЕДОМОСТИ" nv
        left join "Н_ТИПЫ_ВЕДОМОСТЕЙ" on nv."ТВ_ИД" =
"Н_ТИПЫ_ВЕДОМОСТЕЙ"."ИД" and "Н_ТИПЫ_ВЕДОМОСТЕЙ"."ИД" =1
where nv."ЧЛВК_ИД" = 117219;
```

Планы выполнения



Анализ

При первом плане запроса, соединяются две таблицы со всеми записями, а во втором, только подходящие, тем самым размер промежуточных данных будет меньше, значит второй план лучше.

Индексы

```
create index on "Н_ВЕДОМОСТИ" using hash ("ИД");
create index on "Н_ВЕДОМОСТИ" using btree ("ЧЛВК_ИД");
create index on "Н_ТИПЫ_ВЕДОМОСТЕЙ" using btree ("ИД");
```

Выборка происходит с помощью операторов сравнения (‘=’, ‘>’), поэтому структура данных btree наиболее подходящая. Соединяем таблицы с помоьщью ИД, поэтому hash наиболее оптимален.

Explain Analyze

```
Nested Loop Left Join  (cost=0.29..197.79 rows=64 width=662) (actual
time=0.080..0.245 rows=31 loops=1)
"  Join Filter: (nv.""ТВ_ИД"" = ""Н_ТИПЫ_ВЕДОМОСТЕЙ"".""ИД"")"
"  ->  Index Scan using ""ВЕД_ЧЛВК_FK_IFK"" on ""Н_ВЕДОМОСТИ"" nv
(cost=0.29..195.79 rows=64 width=8) (actual time=0.041..0.192 rows=31
loops=1)"
"        Index Cond: (""ЧЛВК_ИД"" = 117219)"
  ->  Materialize  (cost=0.00..1.04 rows=1 width=662) (actual
time=0.001..0.001 rows=1 loops=31)
"        ->  Seq Scan on ""Н_ТИПЫ_ВЕДОМОСТЕЙ""  (cost=0.00..1.04 rows=1
width=662) (actual time=0.030..0.031 rows=1 loops=1)"
"              Filter: (""ИД"" = 1)"
Planning Time: 1.100 ms
Execution Time: 0.321 ms
```

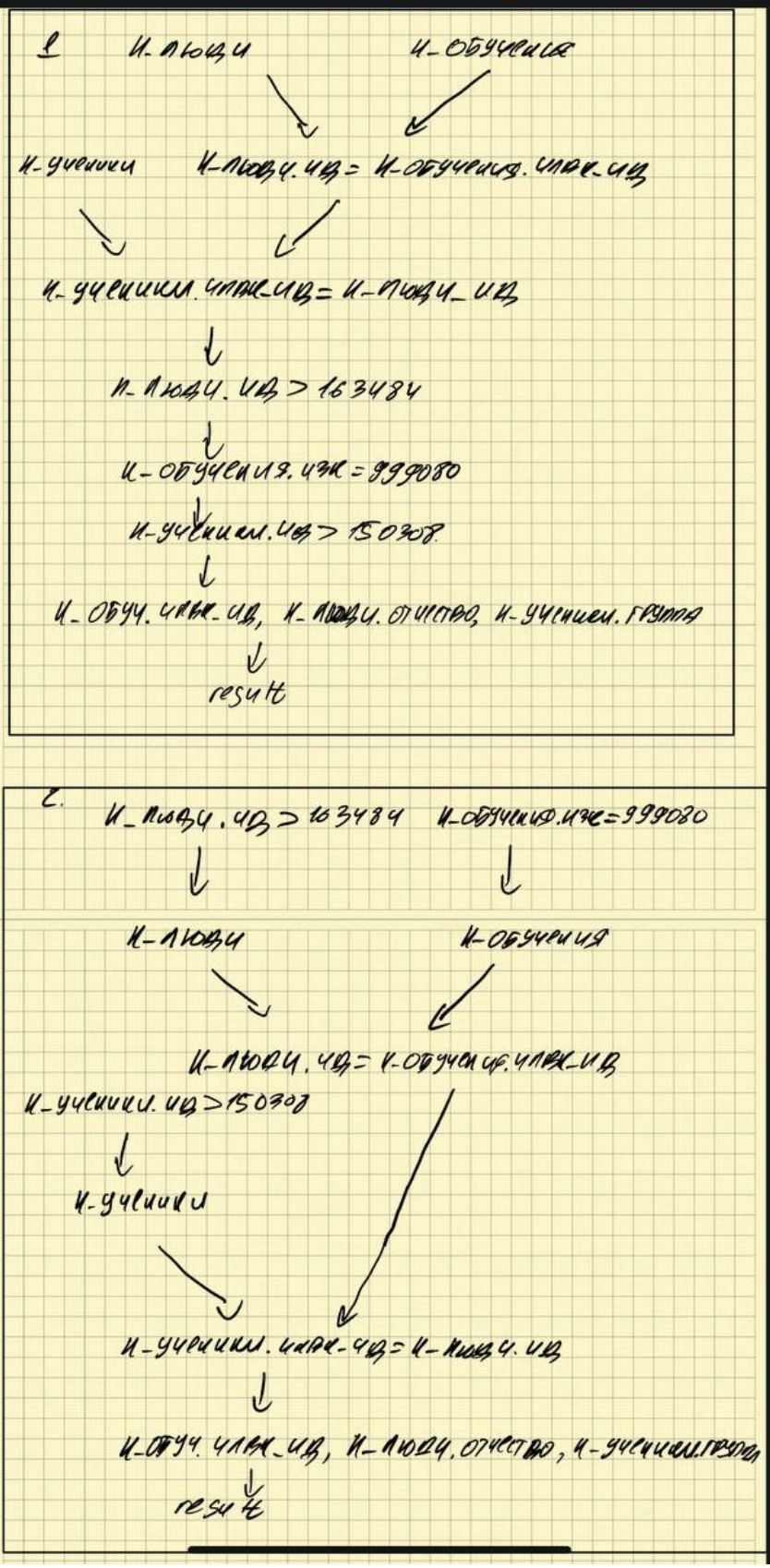
Второй запрос

Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:
Таблицы: Н_ЛЮДИ, Н_ОБУЧЕНИЯ, Н_УЧЕНИКИ.
Вывести атрибуты: Н_ЛЮДИ.ОТЧЕСТВО, Н_ОБУЧЕНИЯ.ЧЛВК_ИД, Н_УЧЕНИКИ.ГРУППА.
Фильтры: (AND)

a) Н_ЛЮДИ.ИД > 163484.
b) Н_ОБУЧЕНИЯ.НЗК = 999080.
c) Н_УЧЕНИКИ.ИД > 150308.
Вид соединения: RIGHT JOIN.

```
select *
from "Н_ЛЮДИ"
        right join "Н_ОБУЧЕНИЯ" no on "ИД" = no."ЧЛВК_ИД"
        right join "Н_УЧЕНИКИ" ny on "Н_ЛЮДИ"."ИД" = ny."ЧЛВК_ИД"
where "Н_ЛЮДИ"."ИД" > 163484
 and no."НЗК" = '111437'
 and ny."ИД" > 150308
```

23:46 Вт 15 нояб. 43 %



Анализ

Второй план является лучше, по той же самой причине.

Индексы

```
create index on "Н_ЛЮДИ" using btree ("ИД");
create index on "Н_ОБУЧЕНИЯ" using btree ("НЗК");
create index on "Н_УЧЕНИКИ" using btree ("ИД");
```

Выборка происходит путем сравнения, поэтому используем сбалансированное дерево.

Explain Analyze

```
Nested Loop  (cost=4.89..31.62 rows=1 width=1035) (actual time=0.005..0.005
rows=0 loops=1)
"  Join Filter: (""Н_ЛЮДИ"".""ИД"" = no.""ЧЛВК_ИД"")"
  ->  Nested Loop  (cost=4.61..31.26 rows=1 width=985) (actual
time=0.004..0.005 rows=0 loops=1)
"        ->  Index Scan using ""ЧЛВК_PK"" on ""Н_ЛЮДИ""  (cost=0.28..8.29
rows=1 width=822) (actual time=0.004..0.004 rows=0 loops=1)"
"              Index Cond: (""ИД"" > 163484)"
"        ->  Bitmap Heap Scan on ""Н_УЧЕНИКИ"" ny  (cost=4.33..22.96
rows=1 width=163) (never executed)"
"              Recheck Cond: (""ЧЛВК_ИД"" = ""Н_ЛЮДИ"".""ИД"")"
"              Filter: (""ИД"" > 150308)"
"              ->  Bitmap Index Scan on ""УЧЕН_ОБУЧ_FK_I""
(cost=0.00..4.32 rows=5 width=0) (never executed)"
"                    Index Cond: (""ЧЛВК_ИД"" = ""Н_ЛЮДИ"".""ИД"")"
"  ->  Index Scan using ""ОБУЧ_ЧЛВК_FK_I"" on ""Н_ОБУЧЕНИЯ"" no
(cost=0.28..0.34 rows=1 width=50) (never executed)"
"        Index Cond: (""ЧЛВК_ИД"" = ny.""ЧЛВК_ИД"")"
"        Filter: ((""НЗК"")::text = '111437'::text)"
Planning Time: 0.581 ms
Execution Time: 0.051 ms
```

Выводы

В ходе выполнения данной лабораторной работы, я познакомился с индексами в PostgreSQL для ускорение обработки данных и какие структуры данных они используют, а именно хэширование и сбалансированное дерево поиска. Строил планы выполнения запроса и выбрал оптимальные.