

ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ ПОЛІТЕХНІЧНИЙ УНІВЕРСИТЕТ  
ІНСТИТУТ КОМП'ЮТЕРНИХ СИСТЕМ  
КАФЕДРА «ІНФОРМАЦІЙНИХ СИСТЕМ»

Лабораторна робота №9

з дисципліни

«Операційні системи»

Тема: «Керування процесами-транзакціями в базах даних. Частина 1»

Варіант 5.1

Виконав:

Студент групи АІ-202

Баранюк Д.А.

Перевірили:

Блажко О.А.

Дрозд М.О.

Одеса 2021

### Завдання для виконання:

#### Завдання 1 Проектування транзакцій

Нехай задані три транзакції:

T1= R[D] R[B] W[D] C1

T2= W[A] R[B] W[B] C2

T3= R[A] W[A] C3

1.1 Створіть історії квазіпаралельного виконання транзакцій для протоколу 1-го ступеня блокування з описом таблиці блокування транзакцій.

$H_{T1,T2,T3} = R1[D] X2[A] W2[A] R3[A] R1[B] R2[B] X3[A] - \text{wait } X1[D] W[D] X2[B] W2[B] U3 C3 U1 C1 U2 C2.$

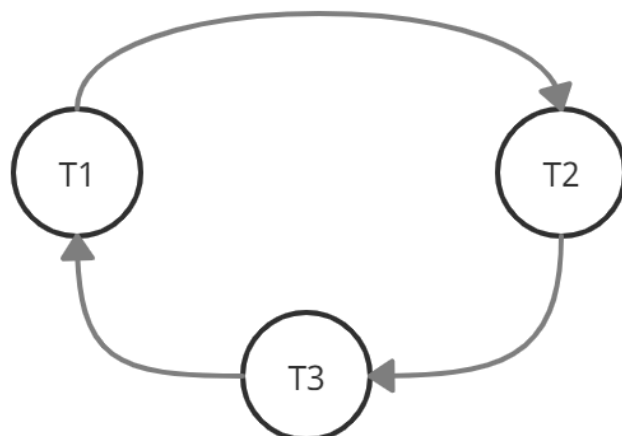
Назва змінної	Перелік встановлених блокувань	Перелік запитів на блокування
A	X2,X3	X3
B	X2	-
D	X1	-

1.2 Повторіть попереднє завдання з використанням протоколу 2-го ступеня блокування.

$H_{T1,T2,T3} = S1[D] R1[D] X2[A] W2[A] S3[A] - \text{wait } S1[B] R1[B] S2[B] R2[B] X3[A] - \text{wait } X1[D] W[D] X2[B] W2[B] U3 C3 U1 C1 U2 C2.$

Назва змінної	Перелік встановлених блокувань	Перелік запитів на блокування
A	X2,X3,S3	X3
B	X2,S1,S2	S3
D	X1,S1	-

1.3 Для створених історій у 1-му та 2-му пунктів завдань визначте наявність тупика транзакції, створивши граф очікування транзакцій.



## Завдання 2 Налаштування бази даних

Нехай існує база даних, назва якої співпадає з іменем вашого користувача в ОС Linux.

Приклад команди створення реляційної таблиці в базі даних наведено в таблиці 1.

Також в таблиці 1 наведено приклад команди внесення одного рядка в реляційну таблицю бази даних.

1.1 Встановіть з'єднання з вашою базою даних.

1.2 У відповідності із варіантом з таблиці 1 створіть реляційну таблицю в базі даних.

1.3 У відповідності із варіантом з таблиці 1 додайте рядок в реляційну таблицю, створену у попередньому пункті завдання.

1.4 Створіть ще одну операцію внесення рядка в таблицю, який буде відрізнятися значеннями всіх змінних від прикладу з варіанту.

```
baranyuk_dmitro@vpsj3IeQ:~  
login as: baranyuk_dmitro  
baranyuk_dmitro@91.219.60.189's password:  
Last login: Tue Apr 27 23:04:50 2021 from 91.222.80.19  
[baranyuk_dmitro@vpsj3IeQ ~]$ psql baranyuk_dmitro  
psql (9.5.25)  
Type "help" for help.  
  
baranyuk_dmitro=> Create table worker (p_id integer, name char(20), bd date);  
CREATE TABLE  
baranyuk_dmitro=> Insert into worker values(1,'Ivanov','01/04/2000');  
INSERT 0 1  
baranyuk_dmitro=> Insert into worker values(2,'Baranyuk','06/16/2003');  
INSERT 0 1  
baranyuk_dmitro=> select * from worker  
baranyuk_dmitro-> ;  
 p_id |      name      |      bd        
-----+-----+-----  
    1 | Ivanov         | 2000-01-04     
    2 | Baranyuk       | 2003-06-16     
(2 rows)  
  
baranyuk dmitro=> █
```

Завдання 3. Керування квазіпаралельним виконанням транзакцій з використанням команд блокування

1 Створіть дві транзакції, кожна з яких повинна включати такі операції:

- операція читання першого рядку таблиці;
- операція зміни однієї із змінних таблиці в першому рядку;

- повторна операція читання першого рядку таблиці;
- операція фіксації всіх змін.

2 При створенні транзакцій включіть відповідні операції блокування для протоколу 1-го ступеня блокування.

3 У двох терміналах виконайте операції транзакцій при їх квазіпаралельному режимі роботи за умови, що одна з транзакцій стартує першою.

4 Повторіть роботу транзакцій, але в першій транзакції замість операції фіксації виконайте операцію відміни.

```
baranyuk_dmitro@vpsj3IeQ:~
baranyuk_dmitro=> Start transaction;
START TRANSACTION
baranyuk_dmitro=> lock table teacher in exclusive mode;
LOCK TABLE
baranyuk_dmitro=> select * from teacher where t_id=1;
  t_id |          name          |          post
-----+-----+-----
      1 | Ivanov                 | docent
(1 row)

baranyuk_dmitro=> update teacher set post = 'professor' where t_id=1;
UPDATE 1
baranyuk_dmitro=> select * from teacher where t_id=1;
  t_id |          name          |          post
-----+-----+-----
      1 | Ivanov                 | professor
(1 row)

baranyuk_dmitro=> COMMIT;
COMMIT
baranyuk_dmitro=> █
```

```
baranyuk_dmitro@vpsj3IeQ:~
login as: baranyuk_dmitro
baranyuk_dmitro@91.219.60.189's password:
Last login: Wed Apr 28 00:15:04 2021 from 91.222.80.19
[baranyuk_dmitro@vpsj3IeQ ~]$ psql
psql (9.5.25)
Type "help" for help.

baranyuk_dmitro=> start transaction;
START TRANSACTION
baranyuk_dmitro=> select * from teacher where t_id=1;
  t_id |          name          |          post
-----+-----+-----
      1 | Ivanov                 | docent
(1 row)

baranyuk_dmitro=> update teacher set post = 'professor' where t_id=1;
UPDATE 1
baranyuk_dmitro=> █
```

```
baranyuk_dmitro@vpsj3leQ:~  
baranyuk_dmitro=> start transaction;  
START TRANSACTION  
baranyuk_dmitro=> lock table teacher in exclusive mode;  
LOCK TABLE  
baranyuk_dmitro=> select * from teacher where t_id=1;  
t_id | name | post  
-----+-----+-----  
1 | Ivanov | professor  
(1 row)  
  
baranyuk_dmitro=> update teacher set post = 'dekan' where t_id=1;  
UPDATE 1  
baranyuk_dmitro=> select * from teacher where t_id=1;  
t_id | name | post  
-----+-----+-----  
1 | Ivanov | dekan  
(1 row)  
  
baranyuk_dmitro=> rollback;  
ROLLBACK  
baranyuk_dmitro=> █
```

```
baranyuk_dmitro@vpsj3leQ:~  
COMMIT  
baranyuk_dmitro=> Start transaction;  
START TRANSACTION  
baranyuk_dmitro=> lock table teacher in share mode;  
LOCK TABLE  
baranyuk_dmitro=> select *from teacher where t_id=1;  
t_id | name | post  
-----+-----+-----  
1 | Ivanov | professor  
(1 row)  
  
baranyuk_dmitro=> COMMIT;  
COMMIT  
baranyuk_dmitro=> start transaction;  
START TRANSACTION  
baranyuk_dmitro=> lock teacher in exclusive mode;  
LOCK TABLE  
baranyuk_dmitro=> █
```

5 Повторіть пункти 3 та 4 але з використанням протоколу 2-го ступеня блокування.

```
baranyuk_dmitro@vpsj3leQ:~  
baranyuk_dmitro=> Start transaction;  
START TRANSACTION  
baranyuk_dmitro=> lock table teacher in share mode;  
LOCK TABLE  
baranyuk_dmitro=> select * from teacher where t_id=1;  
t_id | name | post  
-----+-----+-----  
1 | Ivanov | professor  
(1 row)  
  
baranyuk_dmitro=> update teacher set post = 'prorektor' where t_id=1;  
UPDATE 1  
baranyuk_dmitro=> select * from teacher where t_id=1;  
t_id | name | post  
-----+-----+-----  
1 | Ivanov | prorektor  
(1 row)  
  
baranyuk_dmitro=> Commit;  
COMMIT  
baranyuk_dmitro=> █
```

```
baranyuk_dmitro@vpsj3leQ:~  
START TRANSACTION  
baranyuk_dmitro=> lock table teacher in share mode;  
LOCK TABLE  
baranyuk_dmitro=> select * from teacher where t_id=1;  
t_id | name | post  
-----+-----+-----  
1 | Ivanov | professor  
(1 row)  
  
baranyuk_dmitro=> update teacher set post = 'prorektor' where t_id=1;  
ERROR: deadlock detected  
LINE 1: update teacher set post = 'prorektor' where t_id=1;  
          ^  
  
DETAIL: Process 21145 waits for RowExclusiveLock on relation 16738 of database  
16442; blocked by process 19501.  
Process 19501 waits for RowExclusiveLock on relation 16738 of database 16442; bl  
ocked by process 21145.  
HINT: See server log for query details.  
baranyuk_dmitro=> select * from teacher where t_id=1;  
ERROR: current transaction is aborted, commands ignored until end of transactio  
n block  
baranyuk_dmitro=> commit;  
ROLLBACK  
baranyuk_dmitro=> █
```

```

baranyuk_dmitro@vpsj3leQ:~
baranyuk_dmitro=> select * from techer where t_id=1;
ERROR:  relation "techer" does not exist
LINE 1: select * from techer where t_id=1;
                        ^

baranyuk_dmitro=> select * from teacher where t_id=1;
ERROR:  current transaction is aborted, commands ignored until end of transaction block
baranyuk_dmitro=> commit;
ROLLBACK
baranyuk_dmitro=> start transaction;
START TRANSACTION
baranyuk_dmitro=> lock table teacher in share mode;
LOCK TABLE
baranyuk_dmitro=> select * from teacher where t_id=1;
 t_id |          name          |          post
-----+-----+-----
    1 | Ivanov                  | prorektor
(1 row)

baranyuk_dmitro=> update teacher set post = 'rektor' where t_id=1;

```

```

baranyuk_dmitro@vpsj3leQ:~
ERROR:  deadlock detected
LINE 1: update teacher set post = 'prorektor' where t_id=1;
                        ^

DETAIL:  Process 21145 waits for RowExclusiveLock on relation 16738 of database 16442; blocked by process 19501.
Process 19501 waits for RowExclusiveLock on relation 16738 of database 16442; blocked by process 21145.
HINT:  See server log for query details.
baranyuk_dmitro=> select * from teacher where t_id=1;
ERROR:  current transaction is aborted, commands ignored until end of transaction block
baranyuk_dmitro=> commit;
ROLLBACK
baranyuk_dmitro=> start transaction;
START TRANSACTION
baranyuk_dmitro=> lock table teacher in share mode;
LOCK TABLE
baranyuk_dmitro=> select * from teacher where t_id=1;
 t_id |          name          |          post
-----+-----+-----
    1 | Ivanov                  | prorektor
(1 row)

baranyuk_dmitro=> 

```

```
baranyuk_dmitro@vpsj3leQ:~  
baranyuk_dmitro=> start transaction;  
START TRANSACTION  
baranyuk_dmitro=> lock table teacher in share mode;  
LOCK TABLE  
baranyuk_dmitro=> select * from teacher where t_id=1;  
t_id | name | post  
-----  
1 | Ivanov | prorektor  
(1 row)  
  
baranyuk_dmitro=> update teacher set post = 'rektor' where t_id=1;  
UPDATE 1  
baranyuk_dmitro=> select * from teacher where t_id=1;  
t_id | name | post  
-----  
1 | Ivanov | rektor  
(1 row)  
  
baranyuk_dmitro=> Rollback;  
ROLLBACK  
baranyuk_dmitro=>   
  
baranyuk_dmitro@vpsj3leQ:~  
baranyuk_dmitro=> start transaction;  
START TRANSACTION  
baranyuk_dmitro=> lock table teacher in share mode;  
LOCK TABLE  
baranyuk_dmitro=> select * from teacher where t_id=1;  
t_id | name | post  
-----  
1 | Ivanov | prorektor  
(1 row)  
  
baranyuk_dmitro=> update teacher set post = 'rektor' where t_id=1;  
ERROR: deadlock detected  
LINE 1: update teacher set post = 'rektor' where t_id=1;  
      ^  
  
DETAIL: Process 21145 waits for RowExclusiveLock on relation 16738 of database  
16442; blocked by process 19501.  
Process 19501 waits for RowExclusiveLock on relation 16738 of database 16442; bl  
ocked by process 21145.  
HINT: See server log for query details.  
baranyuk_dmitro=> select * from teacher where t_id=1;  
ERROR: current transaction is aborted, commands ignored until end of transactio  
n block  
baranyuk_dmitro=>   

```

Висновок: під час лабораторної роботи №9 ми отримали практичні навички з керування процесами-транзакціями в базах даних OS Linux. Найскладнішим завдання було завдання 1.