

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

Лабораторна робота №9

З дисципліни

«Операційні системи»

Тема

«Керування процесами-транзакціями в базах даних»

Виконав:

Студент групи АІ-203

Вояковський Д. П.

Перевірили:

Дрозд М.О. Блажко О.А.

Одеса 2021

Завдання для виконання

Завдання 1 Проектування транзакцій

Нехай задані три транзакції, приклади яких представлено в таблиці 13. У відповідності з вашим варіантом виконайте наступні теоретичні завдання.

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

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

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

№ команди	№ учасника	Приклади транзакцій
1	2	T1= W[A] R[B] W[B] C1 T2= R[D] R[B] W[D] C2 T3= R[A] W[A] C3

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.

№ команди	№ учасника	Операція створення реляційної таблиці в базі даних	Операції внесення даних в реляційну таблицю в базі даних
1	2	Create table student (s_id integer, name char(20), kurs integer);	Insert into student values (1, 'Petrov', 4);

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.

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

1.1 $H_{T1,T2,T3}=X1[A], W1[A], R2[D], R3[A], R1[B], R2[B], X3[A]$ -Wait, $X1[B], W1[B], X2[D], W2[D], U1, C1, X3[A], W3[A], U2, C2, U3, C3$

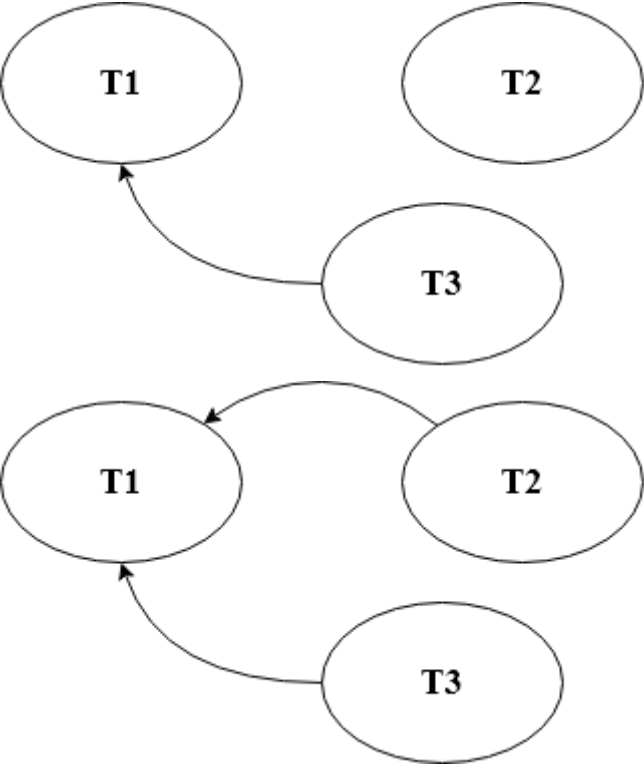
N	Операції T1	Операції T2	Операції T3	
1	X1[A]			
2	W1[A]			
3		R2[D]		
4			R3[A]	
5	R1[B]			
6		R2[B]		
7			X3[A]	Wait
8	X1[B]			
9	W1[B]			
10		X2[D]		
11		W2[D]		
12	U1			
13	C1			
14			X3[A]	
15			W3[A]	
16		U2		
17		C2		
18			U3	
19			C3	

1.2 $H_{T1,T2,T3}=X1[A], W1[A], S2[D], R2[D], S3[A], R3[A], S1[B], R1[B], S2[B]$ -Wait, $X3[A]$ -Wait, $X1[B], W1[B], X2[D], W2[D], U1, C1, S2[B], R2[B], X3[A], W3[A], U2, C2, U3, C3$

N	Операції T1	Операції T2	Операції T3	
1	X1[A]			
2	W1[A]			
3		S2[D]		
4		R2[D]		
5			S3[A]	
6			R3[A]	
7	S1[B]			
8	R1[B]			
9		S2[B]		Wait
10			X3[A]	Wait
11	X1[B]			
12	W1[B]			
13		X2[D]		
14		W2[D]		
15	U1			
16	C1			

17		S2[B]	
18		R2[B]	
19			X3[A]
20			W3[A]
21		U2	
22		C2	
23			U3
24			C3

1.3



2.1-2.4

```
[voyakovskij_dmitro@vpsj3IeQ ~]$ psql voyakovskij_dmitro
psql (9.5.25)
Type "help" for help.
```

```
voyakovskij_dmitro=> create table student (s_id integer, name char(20), kurs integer);
CREATE TABLE
voyakovskij_dmitro=> insert into student values (1, 'Petrov', 4);
INSERT 0 1
voyakovskij_dmitro=> insert into student values (3, 'Ivanov', 2);
INSERT 0 1
```

3

```
voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> select * from student where s_id = 1;
 s_id |      name      | kurs
-----+-----+-----
    1 | Petrov         |    4
(1 row)
```

```
voyakovskij_dmitro=> update student set s_id = 2 where kurs = 4;
UPDATE 1
voyakovskij_dmitro=> select * from student where name = 'Petrov';
 s_id |      name      | kurs
-----+-----+-----
    2 | Petrov         |    4
(1 row)
```

```
voyakovskij_dmitro=> commit;
COMMIT
```

```
voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> select * from student where name = 'Petrov';
 s_id |      name      | kurs
-----+-----+-----
    2 | Petrov         |    4
(1 row)
```

```
voyakovskij_dmitro=> update student set kurs = 5 where name = 'Petrov';
UPDATE 1
voyakovskij_dmitro=> select * from student where name = 'Petrov';
 s_id |      name      | kurs
-----+-----+-----
    2 | Petrov         |    5
(1 row)
```

```
voyakovskij_dmitro=> commit;
COMMIT
```

```

voyakovskij_dmitro=> commit;
COMMIT
voyakovskij_dmitro=> select * from student where name = 'Petrov';
  s_id |          name          | kurs
-----+-----+-----
    7 | Petrov                  |    5
voyakovskij_dmitro=> update student set s_id = 9 where kurs = 5;

UPDATE 1
voyakovskij_dmitro=>
voyakovskij_dmitro=>
voyakovskij_dmitro=>
voyakovskij_dmitro=> select * from student where name = 'Petrov';
  s_id |          name          | kurs
-----+-----+-----
    9 | Petrov                  |    5

voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> update student set s_id = 8 where kurs = 5;
UPDATE 1
voyakovskij_dmitro=> rollback;
ROLLBACK
voyakovskij_dmitro=> select * from student where name = 'Petrov';
  s_id |          name          | kurs
-----+-----+-----
    6 | Petrov                  |    5

voyakovskij_dmitro=> start transaction;
START TRANSACTION
voyakovskij_dmitro=> update student set s_id = 6 where kurs = 5;
UPDATE 1

```

Висновки: під час виконання лабораторної роботи усі завдання були помірної складності.