

Міністерство освіти і науки України  
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»

Виконав:  
Ст. гр. АІ-204  
Дорожкін Михайло

Перевірив(-ла):  
Блажко О. А.  
Дрозд М.О.

Одеса 2021

Мета роботи: дослідити поведінку процесів-транзакцій в базах даних та засоби керуванням ними через механізм блокування з використанням сучасних систем керування базами даних.

Завдання на 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.

|   |   |   |
|---|---|---|
| 5 | 1 | $T1 = R[D] R[B] W[D] C1$<br>$T2 = W[A] R[B] W[B] C2$<br>$T3 = R[A] W[A] C3$ |
|---|---|---|

### **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.

|   |   |   |   |
|---|---|---|---|
| 5 | 1 | <i>Create table worker<br/>( p_id integer, name char(20),<br/>bd date);</i> | <i>Insert into worker values (1, 'Ivanov',<br/>'01/04/2000');</i> |
|---|---|---|---|

### **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 транзакція 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.

Хід виконання роботи:

#### **Завдання 1 Проектування транзакцій**

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

|   |   |  |
|---|---|--|
| 5 | 1 | T1= R[D] R[B] W[D] C1<br>T2= W[A] R[B] W[B] C2<br>T3= R[A] W[A] C3 |
|---|---|--|

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

Історія квазіпаралельного виконання транзакцій для протоколу 1-го ступеня блокування:

R1[D]; X2[A]; W2[A]; R3[A]; R1[B]; R2[B]; X3[A](Wait); X1[D]; W1[D]; X2[B]; W2[B]; U1; C1; U2; C2; X3[A]; W3[A]; U3; C3;

| №  | T1    | T2    | T3    | Стан |
|----|-------|-------|-------|------|
| 1  | R1[D] |       |       |      |
| 2  |       | X2[A] |       |      |
| 3  |       | W2[A] |       |      |
| 4  |       |       | R3[A] |      |
| 5  | R1[B] |       |       |      |
| 6  |       | R2[B] |       |      |
| 7  |       |       | X3[A] | Wait |
| 8  | X1[D] |       |       |      |
| 9  | W1[D] |       |       |      |
| 10 |       | X2[B] |       |      |
| 11 |       | W2[B] |       |      |
| 12 | U1    |       |       |      |
| 13 | C1    |       |       |      |
| 14 |       | U2    |       |      |
| 15 |       | C2    |       |      |
| 16 |       |       | X3[A] |      |
| 17 |       |       | W3[A] |      |
| 18 |       |       | U3    |      |
| 19 |       |       | C3    |      |

Таблиця блокувань:

| Назва змінної | Перелік встановлених блокувань | Перелік запитів на блок |
|---------------|--------------------------------|-------------------------|
| A             | X2                             | X3                      |

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

Історія квазіпаралельного виконання транзакцій для протоколу 2-го ступеня блокування:

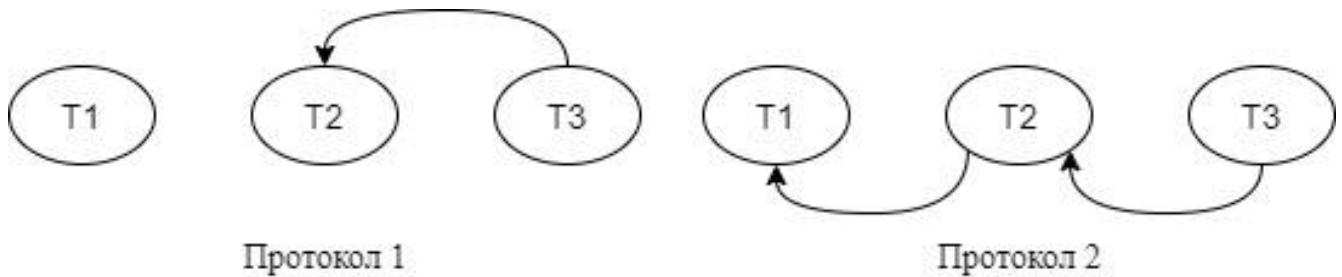
S1[D]; R1[D]; X2[A]; W2[A]; S3[A](Wait); S1[B]; R1[B]; S2[B](Wait); X1[D]; W1[D]; U1; C1; S2[B]; R2[B]; X2[B]; W2[B]; U2; C2; S3[A]; R3[A]; X3[A]; W3[A]; U3; C3;

| №  | T1    | T2    | T3    | Стан |
|----|-------|-------|-------|------|
| 1  | S1[D] |       |       |      |
| 2  | R1[D] |       |       |      |
| 3  |       | X2[A] |       |      |
| 4  |       | W2[A] |       |      |
| 5  |       |       | S3[A] | Wait |
| 6  | S1[B] |       |       |      |
| 7  | R1[B] |       |       |      |
| 8  |       | S2[B] |       | Wait |
| 9  | X1[D] |       |       |      |
| 10 | W1[D] |       |       |      |
| 11 | U1    |       |       |      |
| 12 | C1    |       |       |      |
| 13 |       | S2[B] |       |      |
| 14 |       | R2[B] |       |      |
| 15 |       | X2[B] |       |      |
| 16 |       | W2[B] |       |      |
| 17 |       | U2    |       |      |
| 18 |       | C2    |       |      |
| 19 |       |       | S3[A] |      |
| 20 |       |       | R3[A] |      |
| 21 |       |       | X3[A] |      |
| 22 |       |       | W3[A] |      |
| 23 |       |       | U3    |      |
| 24 |       |       | C3    |      |

Таблиця блокувань:

| Назва змінної | Перелік встановлених блокувань | Перелік запитів на блок |
|---------------|--------------------------------|-------------------------|
| A             | X2                             | S3                      |
| B             | S1                             | S2                      |

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.

1.1 Встановіть з'єднання з вашою базою даних.

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

dorozhkin_mihajlo=> 
```

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

|   |   |   |   |
|---|---|---|---|
| 5 | 1 | <i>Create table worker<br/>( p_id integer, name char(20),<br/>bd date);</i> | <i>Insert into worker values (1, 'Ivanov',<br/>'01/04/2000');</i> |
|---|---|---|---|

```
dorozhkin_mihajlo=> CREATE TABLE worker
dorozhkin_mihajlo-> (p_id integer, name char(20),bd date)
dorozhkin_mihajlo-> ;
CREATE TABLE
dorozhkin_mihajlo=> 
```

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

```
dorozhkin_mihajlo=> INSERT INTO worker VALUES (1, 'Ivanov', '01/04/2000');
INSERT 0 1
dorozhkin_mihajlo=>
```

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

```
dorozhkin_mihajlo=> INSERT INTO worker VALUES (2, 'Petrov', '04/01/2001');
INSERT 0 1
dorozhkin_mihajlo=>
```

1.5 Вміст таблиці

```
dorozhkin_mihajlo=> SELECT * FROM worker;
+-----+-----+
| p_id | name | bd
+-----+-----+
| 1 | Ivanov | 2000-01-04
| 2 | Petrov | 2001-04-01
(2 rows)
```

### 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.

| Транзакція 1                                      | Транзакція 2  |
|---|---|
| START TRANSACTION                                 | START TRANSACTION                                   |
| SELECT * FROM worker WHERE p_id = 1;              | SELECT * FROM worker WHERE p_id = 1;                |
| UPDATE worker SET name = 'Pavlov' WHERE p_id = 1; | UPDATE worker SET bd = '05/05/2002' WHERE p_id = 1; |
| SELECT * FROM worker WHERE p_id = 1;              | SELECT * FROM worker WHERE p_id = 1;                |
| COMMIT;   | COMMIT;   |

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

| Транзакція 1                                      | Транзакція 2  |
|---|---|
| START TRANSACTION                                 | START TRANSACTION                                   |
| SELECT * FROM worker WHERE p_id = 1;              | SELECT * FROM worker WHERE p_id = 1;                |
| LOCK TABLE worker IN EXCLUSIVE MODE;              | LOCK TABLE worker IN EXCLUSIVE MODE;                |
| UPDATE worker SET name = 'Pavlov' WHERE p_id = 1; | UPDATE worker SET bd = '05/05/2002' WHERE p_id = 1; |
| SELECT * FROM worker WHERE p_id = 1;              | SELECT * FROM worker WHERE p_id = 1;                |
| COMMIT;   | COMMIT;   |

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

```
dorozhkin_mihajlo=> START TRANSACTION;
START TRANSACTION
dorozhkin_mihajlo=> SELECT * FROM worker WHERE p_id = 1;
+----+-----+-----+
| p_id | name   | bd      |
+----+-----+-----+
| 1   | Ivanov | 2000-01-04 |
+----+-----+-----+
(1 row)

dorozhkin_mihajlo=> UPDATE worker SET name = 'Pavlov' WHERE p_id = 1;
UPDATE 1
dorozhkin_mihajlo=> UPDATE worker SET name = 'Ivanov' WHERE p_id = 1;
UPDATE 1
dorozhkin_mihajlo=> LOCK TABLE worker IN EXCLUSIVE MODE;
LOCK TABLE
dorozhkin_mihajlo=>
dorozhkin_mihajlo=> UPDATE worker SET name = 'Pavlov' WHERE p_id = 1;
UPDATE 1
dorozhkin_mihajlo=> SELECT * FROM worker WHERE p_id = 1;
+----+-----+-----+
| p_id | name   | bd      |
+----+-----+-----+
| 1   | Pavlov | 2000-01-04 |
+----+-----+-----+
(1 row)

dorozhkin_mihajlo=> COMMIT;
COMMIT
dorozhkin_mihajlo=> 
```

```
dorozhkin_mihajlo@vpsj3leQ:~ %
dorozhkin_mihajlo=> START TRANSACTION;
START TRANSACTION
dorozhkin_mihajlo=> SELECT * FROM worker WHERE p_id = 1;
+----+-----+-----+
| p_id | name   | bd      |
+----+-----+-----+
| 1   | Ivanov | 2000-01-04 |
+----+-----+-----+
(1 row)

dorozhkin_mihajlo=> LOCK TABLE worker IN EXCLUSIVE MODE;
UPDATE worker SET bd = '05/05/2002' WHERE p_id = 1;
SELECT * FROM worker WHERE p_id = 1;
LOCK TABLE
dorozhkin_mihajlo=> UPDATE worker SET bd = '05/05/2002' WHERE p_id = 1;
UPDATE 1
dorozhkin_mihajlo=> SELECT * FROM worker WHERE p_id = 1;
+----+-----+-----+
| p_id | name   | bd      |
+----+-----+-----+
| 1   | Pavlov | 2002-05-05 |
+----+-----+-----+
(1 row)

dorozhkin_mihajlo=> COMMIT;
COMMIT
dorozhkin_mihajlo=> 
```

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

```

dorozhkin_mihajlo@vpsj3leQ:~
1 | Pavlov           | 2000-01-04
(1 row)

dorozhkin_mihajlo=> COMMIT;
COMMIT
dorozhkin_mihajlo=> START TRANSACTION;
START TRANSACTION
dorozhkin_mihajlo=> SELECT * FROM worker WHERE p_id = 1;
 p_id |      name      |      bd
-----+-----+-----+
 1 | Pavlov          | 2002-05-05
(1 row)

dorozhkin_mihajlo=> LOCK TABLE worker IN EXCLUSIVE MODE;
LOCK TABLE
dorozhkin_mihajlo=> UPDATE worker SET name = 'Ivanov' WHERE p_id = 1;
UPDATE 1
dorozhkin_mihajlo=> SELECT * FROM worker WHERE p_id = 1;
 p_id |      name      |      bd
-----+-----+-----+
 1 | Ivanov          | 2002-05-05
(1 row)

dorozhkin_mihajlo=> ROLLBACK;
ROLLBACK
dorozhkin_mihajlo=> 
```

```

dorozhkin_mihajlo@vpsj3leQ:~
dorozhkin_mihajlo=> COMMIT;
COMMIT
dorozhkin_mihajlo=> 
```

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

```

dorozhkin_mihajlo@vpsj3leQ:~
dorozhkin_mihajlo=> START TRANSACTION;
START TRANSACTION
dorozhkin_mihajlo=> SELECT * FROM worker WHERE p_id = 1;
 p_id |      name      |      bd
-----+-----+-----+
 1 | Pavlov          | 2001-05-04
(1 row)

dorozhkin_mihajlo=> LOCK TABLE worker IN SHARE MODE;
LOCK TABLE
dorozhkin_mihajlo=> UPDATE worker SET name = 'Ivanov' WHERE p_id = 1;
UPDATE 1
dorozhkin_mihajlo=> SELECT * FROM worker WHERE p_id = 1;
 p_id |      name      |      bd
-----+-----+-----+
 1 | Ivanov          | 2001-05-04
(1 row)

dorozhkin_mihajlo=> COMMIT;
COMMIT
dorozhkin_mihajlo=> 
```

```

dorozhkin_mihajlo@vpsj3leQ:~
dorozhkin_mihajlo=> START TRANSACTION;
START TRANSACTION
dorozhkin_mihajlo=> SELECT * FROM worker WHERE p_id = 1;
 p_id |      name      |      bd
-----+-----+-----+
 1 | Pavlov          | 2001-05-04
(1 row)

dorozhkin_mihajlo=> LOCK TABLE worker IN SHARE MODE;
LOCK TABLE
dorozhkin_mihajlo=> UPDATE worker SET bd = '05/05/2002' WHERE p_id = 1;
ERROR: deadlock detected
LINE 1: UPDATE worker SET bd = '05/05/2002' WHERE p_id = 1;
^
DETAIL: Process 6952 waits for RowExclusiveLock on relation 16636 of database 1
6528; blocked by process 3862.
Process 3862 waits for RowExclusiveLock on relation 16636 of database 16528; blo
cked by process 6957.
HINT: See server log for query details.
dorozhkin_mihajlo=> 
```

Визначаємо помилку.

**Висновок:** Під час виконання лабораторної роботи ми дослідили поведінку процесів-транзакцій в базах даних та засоби керування ними через механізм блокування з використанням сучасних систем керування базами даних. Найскладнішим для мене було створення історії квазіпаралельного виконання транзакцій.