Project – 1

Rigorous Two-Phase Locking protocol with Wound-Wait method

By Kiran Mai Puli, Student. ID: 1001661668

In this project, a program that simulates the behavior of the Rigorous Two-Phase Locking protocol for concurrency control is implemented. For dealing with deadlock, the Wound-Wait method is used.

Programming Language: The programming language will be used for implementing this project is Python3

Data Structures:

The following data structures are used for implementing tables and other operations in the project

HashTable: The Transaction table and the Lock table are implemented using this data structure.

- **Transaction Table**: A dataframe is created with the following columns to be used as a transaction table
  1) TID (Transaction ID). This is used as the primary key for the transaction table
  2) TimeStamp (Transaction Timestamp)
  3) State (state of the transaction) – the possible states of the transaction are active, blocked, committed and aborted
  4) blockedBy (the transaction blocking our current transaction under discussion)
  5) blockedOperations (the set of operations blocked due to the current transaction being in a Blocked state). As we might be having more than one blocked operations for a transaction, we will be using queues for implementing this column in the table. The data structure queue in order to maintain the "First in – First Out order" while executing these operations.

- **Lock Table**: A dataframe is created with the following columns to be used as a lock table
  1) DataItem (The data item which will be locked by the input operation of the transaction). This is used as the primary key for Lock table
  2) LockMode – Two possible modes are R (read) and W (write)
  3) TIDList – List of transactions that lock the data item. When the LockMode is R, more than one transaction can lock the same data item. The Queue will be used for maintaining the set of transactions
  4) blockedTIDS – List of transactions waiting for the data item to be unlocked. Again a queue will be used for this list

Queue: For maintaining the blocked operations in the transaction table and, transactions list and blocked transactions list in Lock table, queues are used. As there should be an order maintained for implementing the above operations, queues are used.

__Major operations involved in this project:__

1) begin: when the first letter of the input is 'b', then egin function is called. In that function, we insert a new transaction into the transaction table with state as "Active". Timestamp is updated using timeStampCounter
2) Read: when the first letter of the input is 'r', we first check the state of the transaction. If the transaction is 'Active', then readLock function is called where if
   1) If the data item does not exist in the lock table, insert the data item into the lock table with lockMode as 'R' and append the TID to the TIDList
   2) If the data item exits in the lock table:
      - **a.** If LockMode is 'R', append the TID to the TIDList of the lock table
      - **b.** If LockMode is 'W' and if the TIDList[0] == Ti, then we downgrade the lockMode to read
      - **c.** Else implement Wound-Wait deadlock mechanism
        deadlock:
        get the TIDList for the current data item
        if TS(TIDList[i]) > TS(TID) then, abort the transaction
        else, block the current transaction and add the input operation to the blocked operations list in the transaction table and add the TID to the blockedTIDS of the lock table
3) Write: when the first letter of the input is 'w', we first check the state of the transaction. If the transaction is 'Active', then writeLock function is called where if
   1) If the data item does not exist in the lock table, insert the data item into the lock table with lockMode as 'W' and append the TID to the TIDList
   2) If the data item exits in the lock table:
      - a) If LockMode is 'R' and if the TIDList[0] == Ti, and len(TIDList) == 1, then we upgrade the lockMode to write
      - **d.** Else implement Wound-Wait deadlock mechanism
        deadlock:
        get the TIDList for the current data item
        if TS(TIDList[i]) > TS(TID) then, abort the transaction
        else, block the current transaction and add the input operation to the blocked operations list in the transaction table and add the TID to the blockedTIDS of the lock table

4. End: when the first letter of the input is 'e', we first check the state of the transaction. If the state s 'Active', we change the state to 'Committed' and unlock all the data items locked by the current transaction. If the state is 'Blocked', then add the input operation to the blockedOperations. If the state is 'Aborted', then ignore the input operation.

Pseudocode:

The program starts from the main method. In the main method, a loop is used to read the input file one line at a time. inputOperations method is called for each line of the input file. In this method, some string manipulation functions are performed to get the first letter of the input operations and the transaction number

The following cases are possible for different first letters of the input operation

```
1)  if the letter is 'b' then
            timeStampCounter is incremented by 1
            'begin' function is called
                    A new record is inserted into the transaction table with state as 'Active' and
                    timeStampCounter value is assigned to timestamp and display it
    End if

2)  if the letter is 'r' then
       'read' function is called
            If the state of the transaction is "Active" then
                'readLock' function is called
                    if this data item does not exist in the lock table then
                        data item is added to the lock table with lockMode as "R"
`                   else if data item exists and its lockMode is "R" then
                        Add the current transaction into TIDList of Lock table
                    else if date item exists and its locked by the same transaction with write mode
                    then
                        downgrade the lock, i.e., change the lockMode to "R"
                    else if data item exists and its lockMode is "W" then
                       "deadLock" function is called
```

```
                        Get all the TIDs for that particular data item.
                        For all transactions with timestamps greater than timestamp of the
                        current transaction, call “abort” function
                          Transactions in TIDList are aborted. Remove the transactions
                          from TIDList and unlock any data items which are locked by
                          the aborted transactions
                          If the aborted transactions are blocking any transactions then
                            Unblock the transactions and start executing those bloc-
                            ed operations
                        For all transactions with timestamps lesser than timestamp of the current
                        transaction,
                          The current transaction is blocked. Update the state as “Blocked” and
                          add the current operation to the blockedOperations in the tras-
                          action table. Add the transaction to blockedTIDS in lock table
                        end if
                    end if
                end if
            else if the state of the transaction is “Blocked” then
                the current input operation is added to “blockedOperations” of the transaction table
            else if the state is “Aborted” then
                this operation is ignored
            end if

3)  if the letter is ‘w’ then
      ‘write’ function is called
            If the state of the transaction is “Active” then
              ‘writeLock’ function is called
                    if this data item does not exist in the lock table then
                      data item is added to the lock table with lockMode as “W”
                    else if date item exists and its locked by the same transaction with read mode
                    then
                      upgrade the lock, i.e., change the mode to “W”
                    else if data item exists spawned by other transactions
                    "deadLock" function is called
                        For all transactions with timestamps greater than timestamp of the
                        current transaction, call "abort" function
                            Transactions in TIDList are aborted. Remove the transactions
                            from TIDList and unlock any data items which are locked by
                            the aborted transactions
                            If the aborted transactions are blocking any transactions then
                                Unblock the transactions and start executing those bloc-
                                ed operations
                        For all transactions with timestamps lesser than timestamp of the current
                        transaction,
                            The transaction is blocked. Update the state as "Blocked" and
                            add the current operation to the blockedOperations in the tras-
                            action table. Add the transaction to blockedTIDS in lock table
                        end if
                end if
            end if
    else if the state of the transaction is "Blocked" then
            the current input operation is added to "blockedOperations" of the transaction table
    else if the state is "Aborted" then
            this operation is ignored
    end if

4)  if the letter is 'e' then
            if transaction state is 'Active' then
                change the state to 'committed' and unlock the data items that are locked by the
                transaction. If the transaction is blocking any other transactions, unblock those
                transactions and start executing those operations
            else if transaction state is 'Blocked' then
                add this input operation to the list of blocked operations in the transaction table
            else if transaction state is 'Aborted' then
                ignore this input operation
```

Input1:

```
b1;
r1 (Y);
w1 (Y);
r1 (Z);
b3;
r3 (X);
w3 (X);
w1 (Z);
e1;
r3 (Y);
b2;
r2 (Z);
w2 (Z);
w3 (Y);
e3;
r2 (X);
w2 (X);
e2;
```

output1:

```
******************BEGIN******************
operation : b1;
Begin transaction : T1



            Transaction Table
   timeStamp   state blockedBy blockedOperations
TID
T1         1  Active       []              []
```

```
                 Lock Table
Empty DataFrame
Columns: [lockMode, TIDList, blockedTIDS]
Index: []
******************END******************


******************BEGIN******************
operation : r1 (Y);
Item Y read locked by T1


                  Transaction Table
    timeStamp   state blockedBy blockedOperations
TID
T1          1  Active        []                 []


                 Lock Table
         lockMode TIDList blockedTIDS
dataItem
Y                R    [T1]          []
******************END******************


******************BEGIN******************
operation : w1 (Y);
lock mode is upgraded for the data item Y
```

```
                  Transaction Table
    timeStamp   state blockedBy blockedOperations
TID
T1          1  Active       []                []



                 Lock Table
         lockMode TIDList blockedTIDS
dataItem
Y                W    [T1]         []
*******************END*******************



*******************BEGIN*******************
operation : r1 (Z);
Item Z read locked by T1



                  Transaction Table
    timeStamp   state blockedBy blockedOperations
TID
T1          1  Active       []                []



                 Lock Table
         lockMode TIDList blockedTIDS
dataItem
Y                W    [T1]         []
```

```
Z                R    [T1]          []
******************END******************



******************BEGIN******************
operation : b3;
Begin transaction : T3



                  Transaction Table
    timeStamp   state blockedBy blockedOperations
TID
T1          1  Active       []               []
T3          2  Active       []               []



                 Lock Table
         lockMode TIDList blockedTIDS
dataItem
Y                W    [T1]          []
Z                R    [T1]          []
******************END******************



******************BEGIN******************
operation : r3 (X);
Item X read locked by T3



                  Transaction Table
    timeStamp   state blockedBy blockedOperations
TID
T1          1  Active       []              []
T3          2  Active       []              []



                 Lock Table
         lockMode TIDList blockedTIDS
dataItem
Y                W    [T1]         []
Z               R    [T1]         []
X                R    [T3]         []
*******************END*******************



*******************BEGIN*******************
operation : w3 (X);
lock mode is upgraded for the data item X



                  Transaction Table
    timeStamp   state blockedBy blockedOperations
TID
T1          1  Active       []              []
T3          2  Active       []              []



                 Lock Table
         lockMode TIDList blockedTIDS
dataItem
```

```
Y           W    [T1]       []
Z           R    [T1]       []
X           W    [T3]       []
*******************END*******************



*******************BEGIN*******************
operation : w1 (Z);
lock mode is upgraded for the data item Z



              Transaction Table
   timeStamp   state blockedBy blockedOperations
TID
T1        1  Active      []           []
T3        2  Active      []           []



             Lock Table
       lockMode TIDList blockedTIDS
dataItem
Y           W    [T1]       []
Z           W    [T1]       []
X           W    [T3]       []
*******************END*******************



*******************BEGIN*******************
operation : e1;
transaction T1 is committed
```

```
operation e1



                  Transaction Table
    timeStamp       state blockedBy blockedOperations
TID
T1           1  Committed        []               []
T3           2     Active        []               []



                 Lock Table
         lockMode TIDList blockedTIDS
dataItem
X                W    [T3]         []
******************END******************



******************BEGIN******************
operation : r3 (Y);
Item Y read locked by T3



                  Transaction Table
    timeStamp       state blockedBy blockedOperations
TID
T1           1  Committed        []               []
T3           2     Active        []               []
```

```
                 Lock Table
         lockMode TIDList blockedTIDS
dataItem
X                W    [T3]         []
Y                R    [T3]         []
******************END******************



******************BEGIN******************
operation : b2;
Begin transaction : T2



                  Transaction Table
    timeStamp      state blockedBy blockedOperations
TID
T1          1  Committed       []               []
T3          2     Active       []               []
T2          3     Active       []               []



                 Lock Table
         lockMode TIDList blockedTIDS
dataItem
X                W    [T3]         []
Y                R    [T3]         []
******************END******************
```

```
******************BEGIN******************
operation : r2 (Z);
Item Z read locked by T2



              Transaction Table
   timeStamp      state blockedBy blockedOperations
TID
T1         1  Committed       []              []
T3         2     Active       []              []
T2         3     Active       []              []



             Lock Table
       lockMode TIDList blockedTIDS
dataItem
X              W    [T3]         []
Y              R    [T3]         []
Z              R    [T2]         []
******************END******************



******************BEGIN******************
operation : w2 (Z);
lock mode is upgraded for the data item Z



              Transaction Table
   timeStamp      state blockedBy blockedOperations
TID
```

```
T1          1  Committed       []              []

T3          2     Active       []              []

T2          3     Active       []              []



               Lock Table

        lockMode TIDList blockedTIDS

dataItem

X               W    [T3]         []

Y               R    [T3]         []

Z               W    [T2]         []

******************END******************



******************BEGIN******************

operation : w3 (Y);

lock mode is upgraded for the data item Y



                Transaction Table

   timeStamp      state blockedBy blockedOperations

TID

T1          1  Committed       []              []

T3          2     Active       []              []

T2          3     Active       []              []



               Lock Table

        lockMode TIDList blockedTIDS

dataItem
```

```
X               W    [T3]         []

Y               W    [T3]         []

Z               W    [T2]         []

*******************END*******************



*******************BEGIN*******************

operation : e3;

transaction T3 is committed




operation e3




                  Transaction Table

    timeStamp      state blockedBy blockedOperations

TID

T1          1  Committed       []               []

T3          2  Committed       []               []

T2          3     Active       []               []




                 Lock Table

         lockMode TIDList blockedTIDS

dataItem

Z               W    [T2]         []

*******************END*******************



*******************BEGIN*******************
```

```
operation : r2 (X);
Item X read locked by T2



                 Transaction Table
    timeStamp      state blockedBy blockedOperations
TID
T1          1  Committed        []               []
T3          2  Committed        []               []
T2          3     Active        []               []



                Lock Table
         lockMode TIDList blockedTIDS
dataItem
Z               W    [T2]          []
X               R    [T2]          []
******************END******************



******************BEGIN******************
operation : w2 (X);
lock mode is upgraded for the data item X



                 Transaction Table
    timeStamp      state blockedBy blockedOperations
TID
T1          1  Committed        []               []
T3          2  Committed        []               []
```

```
T2          3     Active       []               []


               Lock Table
        lockMode TIDList blockedTIDS
dataItem
Z               W    [T2]         []
X               W    [T2]         []
******************END******************


******************BEGIN******************
operation : e2;
transaction T2 is committed


operation e2


                Transaction Table
   timeStamp      state blockedBy blockedOperations
TID
T1          1  Committed       []               []
T3          2  Committed       []               []
T2          3  Committed       []               []


               Lock Table
Empty DataFrame
Columns: [lockMode, TIDList, blockedTIDS]
```

```
Index: []
******************END******************
```

Input-2:

```
b1;
r1(Y);
w1(Y);
r1(Z);
b2;
r2(Y);
b3;
r3(Z);
w1(Z);
e1;
w3(Z);
e3;
```

output-2:

```
******************BEGIN******************
operation : b1;
Begin transaction : T1



         Transaction Table
  timeStamp   state blockedBy blockedOperations
TID
T1        1  Active       []              []



        Lock Table
```

```
Empty DataFrame
Columns: [lockMode, TIDList, blockedTIDS]
Index: []
******************END******************



******************BEGIN******************
operation : r1(Y);
Item Y read locked by T1



                Transaction Table
    timeStamp   state blockedBy blockedOperations
TID
T1          1  Active       []                []



                Lock Table
         lockMode TIDList blockedTIDS
dataItem
Y               R    [T1]          []
******************END******************



******************BEGIN******************
operation : w1(Y);
lock mode is upgraded for the data item Y



                Transaction Table
    timeStamp   state blockedBy blockedOperations
TID
T1          1  Active       []               []



                 Lock Table
         lockMode TIDList blockedTIDS
dataItem
Y               W    [T1]         []
*******************END*******************



*******************BEGIN*******************
operation : r1(Z);
Item Z read locked by T1



                  Transaction Table
    timeStamp   state blockedBy blockedOperations
TID
T1          1  Active       []               []



                 Lock Table
         lockMode TIDList blockedTIDS
dataItem
Y               W    [T1]         []
Z               R    [T1]         []
*******************END*******************
```

```
******************BEGIN******************

operation : b2;

Begin transaction : T2



                 Transaction Table

    timeStamp   state blockedBy blockedOperations

TID

T1          1  Active       []               []

T2          2  Active       []               []



                Lock Table

         lockMode TIDList blockedTIDS

dataItem

Y                W    [T1]         []

Z                R    [T1]         []

******************END******************



******************BEGIN******************

operation : r2(Y);

deadlock is encountered



                 Transaction Table

    timeStamp    state blockedBy blockedOperations

TID

T1          1   Active       []               []
```

```
T2         2  Blocked      [T1]          [r2(Y);]



                 Lock Table

         lockMode TIDList blockedTIDS

dataItem

Y                W    [T1]       [T2]

Z                R    [T1]         []

******************END******************



******************BEGIN******************

operation : b3;

Begin transaction : T3



                  Transaction Table

    timeStamp    state blockedBy blockedOperations

TID

T1         1   Active       []               []

T2         2  Blocked      [T1]          [r2(Y);]

T3         3   Active       []               []



                 Lock Table

         lockMode TIDList blockedTIDS

dataItem

Y                W    [T1]       [T2]

Z                R    [T1]         []

******************END******************
```

```
******************BEGIN******************
operation : r3(Z);
Item Z read locked by T3



                Transaction Table
    timeStamp    state blockedBy blockedOperations
TID
T1          1   Active       []              []
T2          2  Blocked      [T1]         [r2(Y);]
T3          3   Active       []              []



                Lock Table
         lockMode   TIDList blockedTIDS
dataItem
Y               W      [T1]        [T2]
Z               R  [T1, T3]          []
******************END******************



******************BEGIN******************
operation : w1(Z);
deadlock is encountered
T3 is aborted



                Transaction Table
    timeStamp    state blockedBy blockedOperations
TID
T1          1   Active       []               []
T2          2  Blocked     [T1]         [r2(Y);]
T3          3  Aborted       []               []



                Lock Table
         lockMode TIDList blockedTIDS
dataItem
Y               W    [T1]       [T2]
Z               W    [T1]         []
******************END******************



******************BEGIN******************
operation : e1;
transaction T1 is committed
******************BEGIN******************
operation : r2(Y);
Item Y read locked by T2



                 Transaction Table
    timeStamp      state blockedBy blockedOperations
TID
T1          1  Committed       []               []
T2          2     Active       []               []
T3          3    Aborted       []               []
```

```
                 Lock Table
         lockMode TIDList blockedTIDS
dataItem
Y               R    [T2]         []



operation e1



                  Transaction Table
    timeStamp       state blockedBy blockedOperations
TID
T1          1  Committed       []              []
T2          2     Active       []              []
T3          3    Aborted       []              []



                 Lock Table
         lockMode TIDList blockedTIDS
dataItem
Y               R    [T2]         []
*******************END*******************



*******************BEGIN*******************
operation : w3(Z);
T3 already aborted
*******************END*******************
```

```
*******************BEGIN*******************
operation : e3;
T3 is already aborted
*******************END*******************
```