

CSCI 4127/5127 Homework 4

DUE: April 16 (Monday) via D2L dropbox, see Turnin/Grading. The dropbox will have an extended date to allow for late submissions (with penalty).

Assignment Objectives:

- To understand how indexing and hashing works.
- To understand the consequences of table/row level locking
- To understand the deadlock resolution process.

Assignment Reference:

- *Oracle Database 11g, The Complete Reference, Loney, K*
- In class notes

1. (10 pts) Use a simple modulo hashing algorithm with associated buckets of size 2 and 3 prime pages, show the results of storing records with key values of :13, 6, 7, 8, 11, 5, 4

Page number	Page type	Next Available Overflow page	Number of prime pages	Record 1	Record 2
0	header	6	3	unused	unused
Page number	Page type	Associated Overflow page	Number of Records	Record 1	Record 2
1	prime	-	1	6	-
2	prime	5	3	13	7
3	prime	4	3	8	11
4	overflow			5	
5	overflow			4	
6	overflow			5	
7	overflow			4	
8	overflow				

CSCI 4127/5127 Homework 4

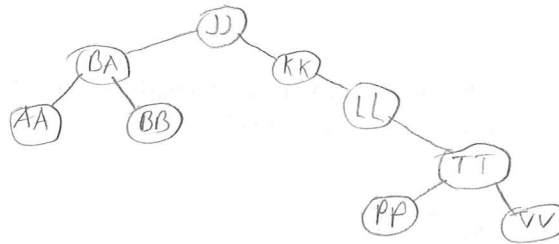
Use the following information for problems 1 and 2.

We wish to store the values with the following arrival order:

JJ, KK, LL, BA, TT, BB, AA, VV, PP

2. Binary tree: (12 pts.)

a. (8 pts.) Show the resulting binary tree.

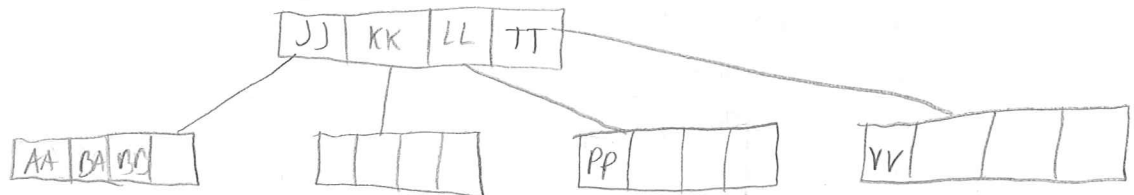


b. (2 pts.) What is the maximum number of nodes to access for a search? 5

c. (2 pts.) What value[s] take the longest to find? PP & VV

3. B-Trees (12 pts.):

b. (8 pts.) Using an order 2 B-tree, show the resulting B-tree.



c. (2 pts.) What is the maximum number of nodes to access for a search? 4

d. (2 pts.) What nodes will be accessed to locate "LL"? the top node's first three elements

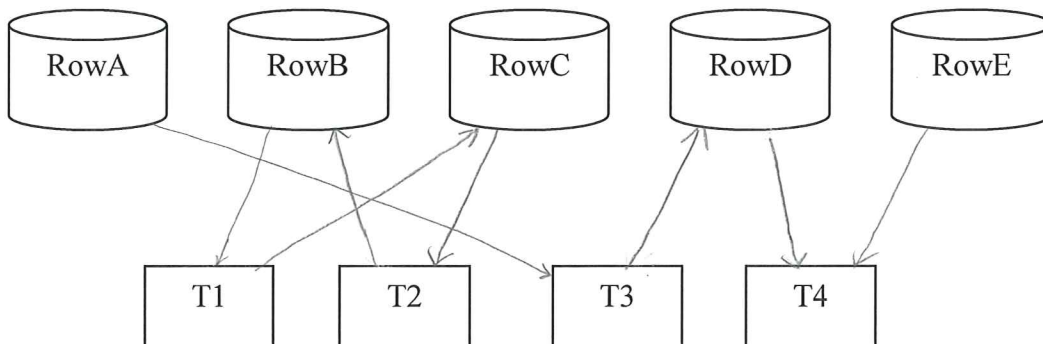
CSCI 4127/5127 Homework 4

4. Deadlock Detection. Given the following:

- Four transactions called T1, T2, T3, T4
- Five rows called RowA, RowB, RowC, RowD, and RowE
- And the following sequence of transactions (you may stop processing if/when a deadlock is encountered):

Step	Transaction	Requests Resource
1.	T1	RowB
2.	T3	RowA
3.	T2	RowC
4.	T4	RowD
5.	T2	RowB
6.	T3	RowD
7.	T4	RowE
8.	T1	RowC
9.	T3	RowC
10.	T2	RowA

Using the Resource Allocation Process Wait method



CSCI 4127/5127 Homework 4

- a. (5 pts) Which step causes a deadlock (if no deadlock exists, state "no deadlock")

Step 8

- b. (5 pts) Which transactions are involved in the deadlock (if no deadlock exists, state "no deadlock")

$T1 \rightarrow B, T2 \rightarrow C, T2 \rightarrow B, T1 \rightarrow C$

- c. (5 pts) Which transactions are impacted by the deadlock, but not directly involved (if no deadlock exists, state "no deadlock")

Row A, D, & E have no effect & are free to complete.

- d. (5 pts) Which transaction should be killed? (if no deadlock exists, state "no deadlock") T1 or T2

The latest transaction of T1.

- e. (5 pts) If a deadlock exists, show a possible order of completion of the remaining jobs. If no deadlock exists show a possible order of completion of the jobs as given.

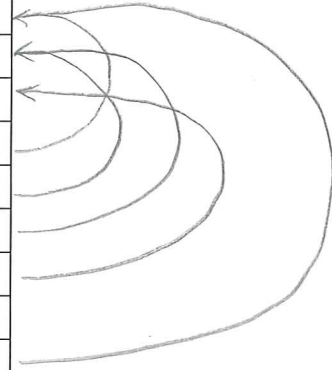
$T3 \rightarrow A, T4 \rightarrow D, T3 \rightarrow D, T4 \rightarrow E$

CSCI 4127/5127 Homework 4

5. Deadlock Detection. Given the following:

- Four transactions called T1, T2, T3, T4
- Five rows called RowA, RowB, RowC, RowD, and RowE
- And the following sequence of transactions (you may stop processing if/when a deadlock is encountered):

Step	Transaction	Requests Resource
1.	T2	RowC
2.	T4	RowD
3.	T3	RowB
4.	T1	RowA
5.	T3	RowD
6.	T2	RowB
7.	T1	RowB
8.	T4	RowA
9.	T3	RowE
10.	T2	RowD



Using the matrix reduction method:

- a. (5 pts) Which step causes a deadlock (if no deadlock exists, state “no deadlock”)

Show the final matrix.

Matrix on next page, no deadlock

- b. (5 pts) Which transactions are involved in the deadlock (if no deadlock exists, state “no deadlock”)

no deadlock

- c. (5 pts) Which transactions are impacted by the deadlock, but not directly involved (if no deadlock exists, state “no deadlock”)

no deadlock

- d. (5 pts) Which transaction should be killed? (if no deadlock exists, state “no deadlock”) T1 or T2

no deadlock

- e. (5 pts) If a deadlock exists, show a possible order of completion of the remaining jobs. If no deadlock exists show a possible order of completion of the jobs as given.

i. *T2 → C, T4 → D, T3 → B, T1 → A, T3 → D, T2 → B, T1 → B, T4 → A, T3 → E, T2 → D*

waiting on	1	2	3	4
1			X	
2			X	
3				X
4	X	X		