The assignment is to be turned in before Midnight (by 11:59pm) on February 17th , 2015. Late submissions are accepted, but there is a 5 point deduction for each day the assignment is late. You should turn in the solutions to this assignment as a pdf file through the TEACH website. The solutions should be produced using editing software programs, such as LaTeX or Word, otherwise they will not be graded.

1: Indexing (1 point)

Using B+ trees with degrees of two (2) whose keys are integer values, give an example of a B+ tree whose height changes from 2 to 3 when the value 25 is inserted. Show your structure before and after the insertion. A B+ tree with a single node has height of 1.

2: Query processing (2 points)

Consider the join of the relation R and S on attributes R.a=S.b, given the following information about the relations to be joined. Relation R contains 10,000 tuples and has 10 tuples per block. Relation S contains 2000 tuples and also has 10 tuples per block. Attribute b of relation S is the primary key for S. Neither relation has any indexes built on it. 52 buffer pages are available in main memory. What is the cost of joining R and S using a sort-merge join? You should use a version of sort-merge algorithm that provides the minimum cost. The cost metric is the number of block I/Os.

3: Query optimization (1 points)

Consider the following relational schema and SQL query:

Suppliers(sid, sname, city)
Supply(sid, pid)
Parts(pid, pname, price)

SELECT S.sname, P.pname
FROM Suppliers S, Parts P, Supply Y
WHERE S.sid = Y.sid AND Y.pid = P.pid

How many different join orders, assuming that cross-products are disallowed, does a System R style query optimizer consider when deciding how to process the given query? List the join orders.

4: Concurrency control (2 points)

Consider the schedule shown at Table 1.

(a) What is the equivalent serialization order for this schedule? If no order is possible, you may state so.

(b) When all transactions run in the above schedule, identify the transactions with degree 3 consistency. Answer the same question when transaction T3 does not exist in the schedule.

	T1	T2	T3
0		start	
1		read X	
2	start		
3	read Y		
4		write X	
5			start
6			read X
7			write X
8			Commit
9	read X		
10	write Y		
11	write X		
12	Commit		
13		read Y	
14		write Y	
15		Commit	

Table 1: Transaction schedule

5: Recovery (4 points)

In this problem, you need to simulate the actions taken by ARIES. Consider the following log records and buffer actions:

time	LSN	Log	Buffer actions
0	00	update: T1 updates P7	P7 brought in
1	10	update: T0 updates P9	P9 brought in; P9 kicked out
2	20	update: T1 updates P8	P8 brought in; P8 kicked out
3	30	begin_checkpoint	
4	40	end_checkpoint	
5	50	update: T1 updates P9	P9 brought in
6	60	update: T2 updates P6	P6 brought in
7	70	update: T1 updates P5	P5 brought in
8	80	update: T1 updates P7	P6 kicked out
9		CRASH RESTART	

(a) For the actions listed above, show Transaction Table (XT) and Dirty Page Table (DPT) after each action. Assume that DPT holds pageID and recLSN, and XT contains transID and lastLSN.

(b) Simulate Analysis phase to reconstruct XT and DPT after crash. Identify the point where the Analysis phase starts scanning log records and show XT and DPT after each action.

(c) Simulate Redo phase: first identify where the Redo phase starts scanning the log records. Then, for each action identify whether it needs to be redone or not.

(d) Simulate Undo phase: identify all actions that need to be undone. In what order will they be undone?