Database Management Systems
Written Examination

18.09.2007

First name		Last name	
Student number		Signature	

Instructions for Students

- Write your name, student number, and signature on the exam sheet.
- Write your name and student number on every solution sheet you hand in.
- This is a closed book exam: the only resources allowed are blank paper, pens, and your head. Use a pen, not a pencil.
- Write neatly and clearly. The clarity of your explanations affects your grade.
- You have 120 minutes for the exam.

Reserved for the Teacher

Exercise	Max. points	Points
1	20	
2	8	
3	12	
4	8	
5	20	
6	8	
7	12	
8	12	
Total	100	

Exercise 1 Answer the following questions:

a. Mention and briefly describe two buffer-replacement policies?

b. Describe how to locate a record with search-key value K using a sparse index.

c. What is the number of block accesses for a merge-join, assuming that the relations are sorted and the join attribute(s) in one relation forms a key.

d. What are the two forms of materialized view maintenance?

e. What are the three steps of query processing?

f. What is specified in a query plan (evaluation plan)?

g. Describe the A property of the ACID properties?

h. What is a cascading rollback?

i. What are the two phases of the two-phase locking protocol?

j. Which two of the ACID properties are ensured by the recovery system?

Exercise 2 Consider the following file of variable-length records that stores the `exams` relation of exercise 4 using the byte string representation.

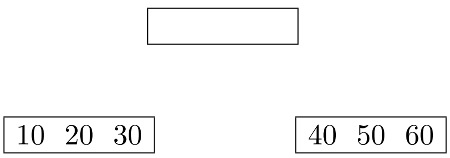
r_0	Jan	AI	C	3	DB	A	2	DMS	B	2	$\perp$
r_1	Ann	AI	A	3	CSA	D	1	$\perp$			
r_2	Bob	OS	C	1	DB	C	2	$\perp$			
r_3	Sue	DMS	A	2	ITP	A	1	$\perp$			
r_4	Joe	DMS	B	2	$\perp$						

a. Show the file structure with fixed-length representation with reserved space.

b. Show the file structure with fixed-length represenation with pointers and anchor and overflow block.

Exercise 3 Consider a B$^+$-tree, where at most 4 pointers fit into a single node.

a. The following is an incomplete B$^+$-tree for the keys $\{10, 20, 30, 40, 50, 60\}$. Complete the tree.



b. Insert 25 and 18 (in that order). Show the resulting tree.

c. Delete 30. Show the resulting tree.

Exercise 4 Consider the following relation `exams`:

	Name	Course	Grade	Year
0	Jan	AI	C	3
1	Jan	DB	A	2
2	Jan	DMS	B	2
3	Ann	AI	A	3
4	Ann	CSA	D	1
5	Bob	OS	C	1
6	Bob	DB	C	2
7	Sue	DMS	A	2
8	Sue	ITP	A	1
9	Joe	DMS	B	2

a. Create a bitmap index over the two most suitable attributes. Explain your choice.

b. Describe the evaluation of $\sigma_{Course=DMS \wedge Grade=A \wedge Year=2}(\texttt{exams})$ using the index.

Exercise 5 Consider a relation $r(A, B)$ with a primary B^+ tree index on attribute A and the following characteristics: $|r| = 4,000,000$; 100 index entries per block; 20 data tuples per block; the values of both attributes A and B are uniformly distributed in the range $[1 - 10,000,000]$; A forms a key; seek time = 0.016 sec; latency = 0.008 sec; transfer time = 0.001 sec.

a. Determine the number of blocks (= number of nodes) used for the B^+-tree if index blocks are filled up to 80%.

b. Describe the evaluation and determine the number of block IOs (distinguishing between index blocks and data blocks) for the following queries:

- Q1: $\sigma_{A>2Mio \wedge A<4Mio}(r)$
- Q2: $\sigma_{A<2Mio \vee A>4Mio}(r)$
- Q3: $\sigma_{A=5Mio \wedge B>5Mio}(r)$
- Q4: $\sigma_{A>5Mio \vee B>5Mio}(r)$

c. Determine the execution time for the queries in b.)

Exercise 6 For each of the following formulas, show that it is or is not a valid equivalence rule for query optimization, and if necessary, specify conditions for which this holds or does not hold.

a. $\sigma_\theta(E_1 - E_2) = \sigma_\theta(E_1) - E_2$
b. $\sigma_{\theta_1 \wedge \theta_2}(E_1 \bowtie E_2) = \sigma_{\theta_1}(E_1) \bowtie \sigma_{\theta_2}(E_2)$
c. $\pi_L(\sigma_\theta(E)) = \sigma_\theta(\pi_L(E))$
d. $\pi_A(E_1 - E_2) = \pi_A(E_1) - \pi_A(E_2)$

Exercise 7 Let relations $r(A, B)$ and $s(A, C)$ have the following properties: r has 10,000 tuples and 5 tuples of r fit into one block; s has 125 tuples and 10 tuples of s fit into one block. In relation r we have a hash index on attribute A, where 100 (search-key,pointer)-pairs fit into one bucket. Furthermore, the A-values in r are equally distributed over the A-values in s, and 20 blocks fit into main memory.
Compute the cost of the following evaluation plans for $r \bowtie s$:

- Plan p1: Nested loop with r as outer loop
- Plan p2: Nested loop with s as outer loop and hashed lookup in r
- Plan p3: Hash join
- Plan p4: Merge join (assume r and s are already sorted)

Exercise 8 Given is the following schedule that involves transactions T_1, T_2, T_3:

T_1	T_2	T_3
	read(Z)	
	read(Y)	
	write(Y)	
		read(Y)
		read(Z)
read(X)		
write(X)		
		write(Y)
		write(Z)
	read(X)	
read(Y)		
write(Y)		
	write(X)	

a. Draw the precedence graph of this schedule.

b. Show that the schedule is not conflict serializable.

c. Design a concurrent schedule of T_1, T_2 and T_3 that is conflict serializable. Specify also the equivalent serial schedule.

d. Is the schedule possible under the timestamp ordering protocol? Explain your answer.

Solution 1

a. LRU strategy: Replace the block least recently used.
MRU strategy: Replace the block most recently used.

b. First, find index record with largest search-key value $< K$. Then, search the data file sequentially starting at the record to which the index record points.

c. $b_r + b_s$, where b_r and b_s are the number of blocks in relation r and s, respectively.

d. Recomputation on every database update and incremental view maintenance.

e. (i) Parsing and translation, (ii) optimization, (iii) evaluation

f. Query plan defines what algorithm is used for each operation and how the execution of the operations is coordinated.

g. Atomicity: A transaction's changes to the state of a DB are atomic, i.e., either all operations of a transaction are properly reflected in the DB or none are.

h. A single transaction failure leasds to a series of transaction rollbacks.

i. Growing phase and shrinking phase.

j. Atomicity and durability.

Solution 2

a. Fixed-length representation with reserved space

0	Jan	AI	C	3	DB	A	2	DMS	B	2
1	Ann	AI	A	3	CSA	D	1	$\perp$	$\perp$	$\perp$
2	Bob	OS	C	1	DB	C	2	$\perp$	$\perp$	$\perp$
3	Sue	DMS	A	2	ITP	A	1	$\perp$	$\perp$	$\perp$
4	Joe	DMS	B	2	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$	$\perp$

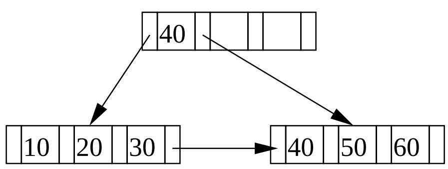
b. Fixed-length representation with pointers ($\uparrow x$ denotes a pointer to record x)

r_0	Jan	AI	C	3	$\uparrow o_0$
r_1	Ann	AI	A	3	$\uparrow o_2$
r_2	Bob	OS	C	1	$\uparrow o_3$
r_3	Sue	DMS	A	2	$\uparrow o_4$
r_4	Joe	DMS	B	2	$\perp$

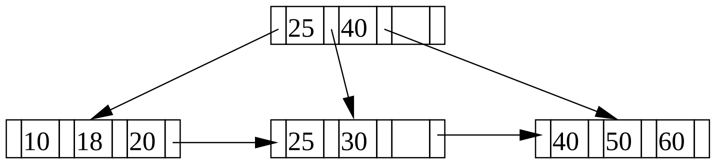
o_0	DB	A	2	$\uparrow o_2$
o_1	DMS	B	2	$\perp$
o_2	CSA	D	1	$\perp$
o_3	DB	C	2	$\perp$
o_4	ITP	A	1	$\perp$

Solution 3

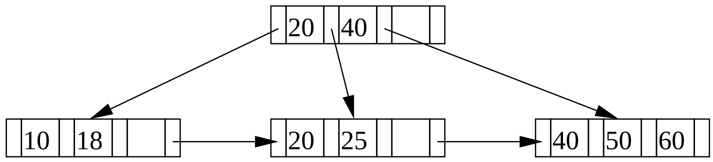
a.



b.



c.



Solution 4

a. *Grade* and *Year* are the most suitable attributes for a bitmap index, since they have the smallest number of different values keeping the index as small as possible.

Grade A: [0 1 0 1 0 0 0 1 1 0]
Grade B: [0 0 1 0 0 0 0 0 0 1]
Grade C: [1 0 0 0 0 1 1 0 0 0]
Grade D: [0 0 0 0 1 0 0 0 0 0]

Year 1: [0 0 0 0 0 0 0 0 1 0]
Year 2: [0 1 1 0 1 1 1 1 0 1]
Year 3: [1 0 0 1 0 0 0 0 0 0]

b. Take the bitmap vector of Grade A and Year 2 and compute the logical AND:
[0 1 0 1 0 0 0 1 1 0] AND [0 1 1 0 1 1 1 1 0 1] = [0 1 0 0 0 0 0 1 0 0]

Retrieve the tuple(s) with a '1' in the resulting bitmap vector, i.e., tuple no. 1 and tuple no. 7, and then select those tuples that have Course=DMS, i.e., $(Sue, DMS, A, 2)$.

Solution 5

a. Index blocks:
$0.8 * 100 = 80$ index entries per block
Index blocks required at each level:
- level 4: $\lceil 4,000,000/80 \rceil = 50,000$ blocks (leaf nodes)
- level 3: $\lceil 50,000/80 \rceil = 625$ blocks
- level 2: $\lceil 625/80 \rceil = 8$ blocks
- level 1: $\lceil 8/80 \rceil = 1$ block
$\Rightarrow 50,634$ index blocks are needed in total

b. Total number of data blocks: $\lceil 4,000,000/20 \rceil = 200,000$

$Q1$: Use index to locate the first tuple with $A > 2$ Mio, then scan the data relation sequentially.
4 index blocks (one at each level)

40,000 data blocks (= 20% of the data blocks)
$\Rightarrow 4 + 40,000 = 40,004$ blocks in total

$Q2$: Scan data relation from the beginning until $A \geq 2Mio$; then use index to locate the first tuple with $A > 4$ Mio and scan the data relation sequentially from that point.
Condition $A < 2$ Mio: 40,000 data blocks
Condition $A > 4$ Mio: 4 index blocks + 120,000 data blocks = 120,004 blocks
$\Rightarrow$ 160,004 blocks in total

$Q3$: Use the index to find the tuple (if there is one) with $A = 5$ Mio and check this tuple for the second condition, $B > 5$ Mio.
4 index blocks (one at each level); 1 data block
$\Rightarrow 4 + 1 = 5$ blocks in total

$Q4$: Scan all data blocks; index is not useful.
$\Rightarrow$ 200,000 data blocks in total

c. Execution time: 1 IO $= 0.016 + 0.008s + 0.001s = 0.025s$
Q1: $40,004 \times 0.025s = 1,000.1s$
Q2: $160,004 \times 0.025s = 40,000.1s$
Q3: $5 \times 0.025s = 0.125s$
Q4: $200,000 \times 0.025s = 5,000s$

Solution 6

a. Always correct. We proof this by showing set containment in both directions:
($\longrightarrow$) Assume that $\exists t \in \sigma_\theta(E_1 - E_2)$. Then t satisfies θ and $t \in E_1$ and $t \notin E_2$. Therefore we get $t \in \sigma_\theta(E_1)$. We get also $t \in \sigma_\theta(E_1) - E_2$ since $t \notin E_2$.
($\longleftarrow$) Assume that $\exists t \in \sigma_\theta(E_1) - E_2$. Then t satisfies θ and $t \in E_1$ and $t \notin E_2$. Therefore we get $t \in (E_1 - E_2)$, and since t satisfies θ we get further $t \in \sigma_\theta(E_1 - E_2)$.

b. Correct, if θ_1 uses only attributes in E_1 and θ_2 uses only attributes in E_2.

c. Correct, if θ uses only attributes in L.

d. Not correct. Consider the following couter example including two relations with schema (A, B): $E_1 = \{(1,2),(1,5)\}$ and $E_2 = \{(1,2)\}$. The left-hand side is empty, whereas the right-hand side gives (1).

Solution 7
Blocks required for r: $\lceil 10,000/5 \rceil = 2,000$ blocks
Blocks required for s: $\lceil 125/10 \rceil = 13$ blocks

Plan p1:
$cost(p1) = n_r * b_s + b_r = 10,000 * 13 + 2,000 = 132,000$ block transfers

Plan p2:
$cost(p2) = b_s + n_s * c = 13 + 125 * 16 = 2,013$ block transfers
(c is the cost for the hashed lookup and retrieval of machting tuples: cost for

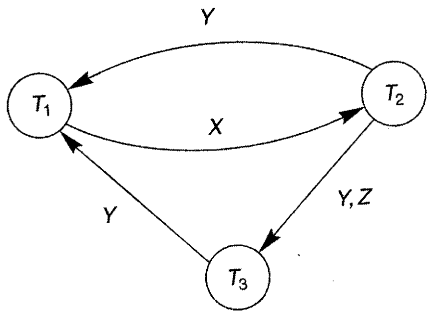
lookup is 1; each s-tuple has on avg. $10,000/125 = 80$ matching r-tuples, which corresponds to $80/5 = 16$ blocks)

Plan p3:
$cost(p3) = 3 * (b_r + b_s) = 3 * (2,000 + 13) = 6,039$

Plan p4:
$cost(p4) = b_r + b_s = 2,000 + 13 = 2,013$

Solution 8

a. Precedence graph:



b. There is a cycle in the precedence graph: $T_1 \xrightarrow{X} T_2$ and $T_2 \xrightarrow{Y} T_1$. Hence, the schedule is not conflict serializable.

c. The following schedule is conflict serializable to the serial schedule $\langle T_3, T_1, T_2 \rangle$

T_1	T_2	T_3
		read(Y)
		read(Z)
read(X)		
write(X)		
		write(Y)
		write(Z)
	read(Z)	
read(Y)		
write(Y)		
	read(Y)	
	write(Y)	
	read(X)	
	write(X)	

d. No. There are conflicting read and write operations on Y in T_1 and T_2, which are not executed in timestamp order.