CS585

Database Systems

Spring 2006

Final Exam

Name: ______________________

Student ID: __________________

	Maximum	Received
Problem 1	10	
Problem 2	15	
Problem 3	15	
Problem 4	20	
Problem 5	10	
Problem 6	15	
Problem 7	15	

Note: Students can bring a one-sided 8.5”x11” sheet of notes to the exam.

1) Short answer questions

a. 4pts

Give two examples of the types of applications that run on top of a data warehouse. Give a brief description of each application type.

b. 3 pts

If a distributed database can provide visibility into a distributed set of data, then why is there ever a need to bring the data into a data warehouse?

c. 3 pts

OLAP engines can speed up certain queries by pre-computing certain results. Explain what type of queries this technique works best for.

2) 15pts

a-Which of the following index structures are balanced:

- KDB tree
- Point quad tree
- PR quad tree
- R Tree
- R^+ Tree

b- For each of the index trees above that are NOT balanced give a an example that shows the worst case scenario in terms of the index tree height. Your examples will be a sequence of points in the X-Y plane. Show both the index tree and the location of the points labeled with their order.

3) 15 pts
Given the following DTD, write an XML documents that is valid with respect to this DTD. You should include at least two different students with a minimum of 5 recorded courses total.

```
<?xml version="1.0" encoding="UTF-8"?>

<!-- Define the university as a collection of school elements. -->
<!ELEMENT allstudents (student)+>

<!-- Speficy information about schools -->
<!ELEMENT student (first, middle?, last, major*, email*, GPA, courses)>
<!ATTLIST student sid CDATA #REQUIRED>
<!-- for example: s123456 -->

<!ELEMENT first (#PCDATA)>
<!ELEMENT middle (#PCDATA)>
<!ELEMENT last (#PCDATA)>
<!ELEMENT major (#PCDATA)>
<!ELEMENT email (#PCDATA)>
<!ELEMENT GPA (#PCDATA)>

<!-- Speficy information about courses -->
<!ELEMENT courses (term+)>

<!ELEMENT term (course+)>
<!ATTLIST term name CDATA #REQUIRED>

<!-- Speficy information about each course -->
<!ELEMENT course (id,grade)>
<!ELEMENT id (#PCDATA)>
<!ELEMENT grade (#PCDATA)>
<!-- grade should be only one of A,B,C,D,F -->
```

Additional space from problem 3

4) 20 pts
Given an instance of an XML document as described in problem 3, i.e. valid with respect to the DTD given. Write queries to

a) Find the email addresses of all students who have declared their major to be CS. (Return email elements)

b) Find the last names of all students who have GPA of 3.5 or higher. (Return “last” elements)

c) Verify the GPA for each student, i.e., compute the GPA from scratch for each student and compare the result with the GPA in the student's record. For simplicity, assume that there are only A,B,C,D,F grades that give 4, 3, 2, 1, 0 points respectively. Also assume that all courses have the same number of credits (i.e. you only need to divide by the number of all courses attempted).
Return: Only for students whose GPA is miscalculated, return entries in the following format:

```
<student sid="s1234">
    <newgpa>3.66</newgpa>
    <oldgpa>3.5</oldgpa>
</student>
```

where newgpa is the GPA computed by you, and oldgpa is value for the GPA stored in the student's record.

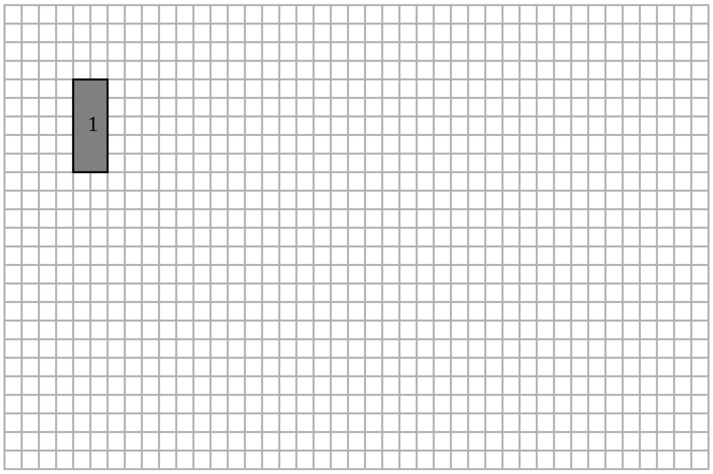
5) 10 pts
Assume a relation *Population* that stores the number of male and female voters for each city. The relation has three attributes: *city* (of type string), *num_female*, and *num_male* (both of type number). Assume two applications that do certain calculations on this table. The following predicates are used in these two applications:

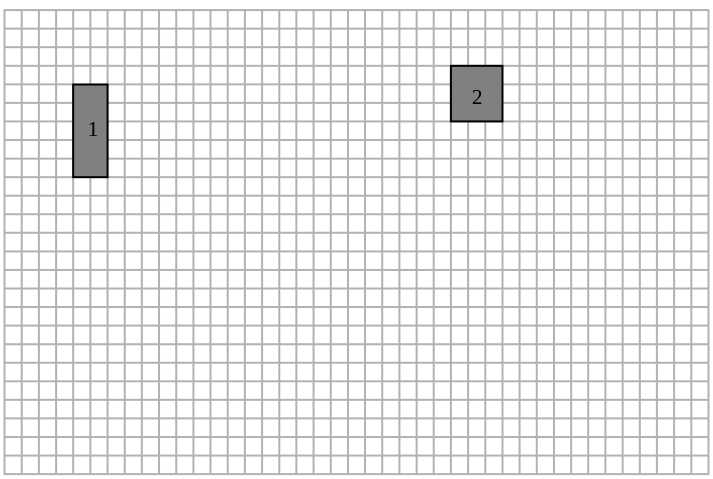
P_1: $2 * num\_male < num\_female$
P_2: $num\_male = num\_female$

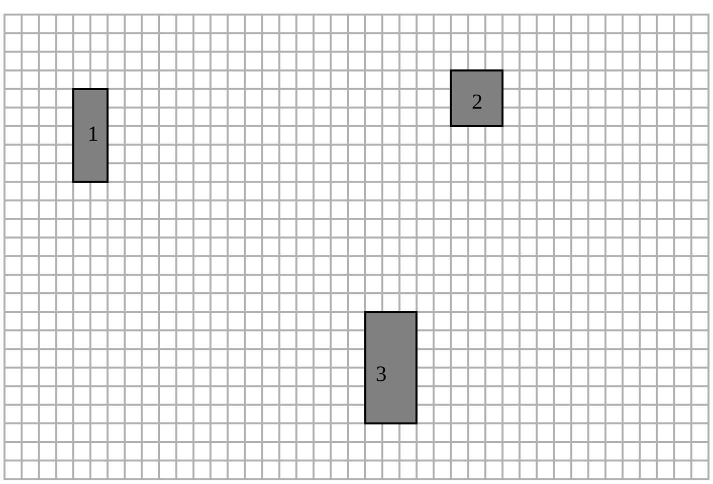
Given these two simple predicates, determine all minterms such that the correctness rules for horizontal fragmentation are satisfied.

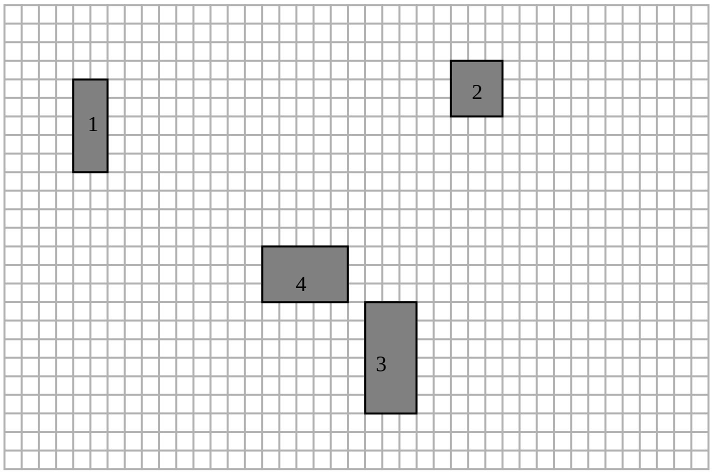
6) 15 pts

9 rectangles are insterted into a spatial DBMS that uses an R-Tree of size (2,4) as an index structure to store them. Show the R-Tree and all MBRs after each insertion in the order given.

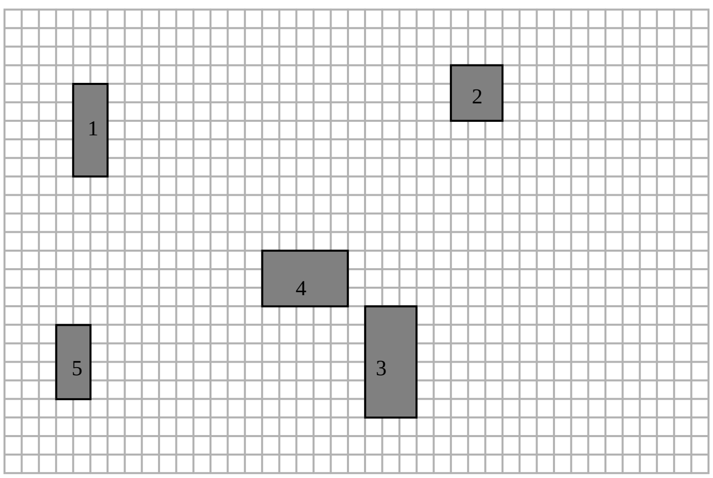




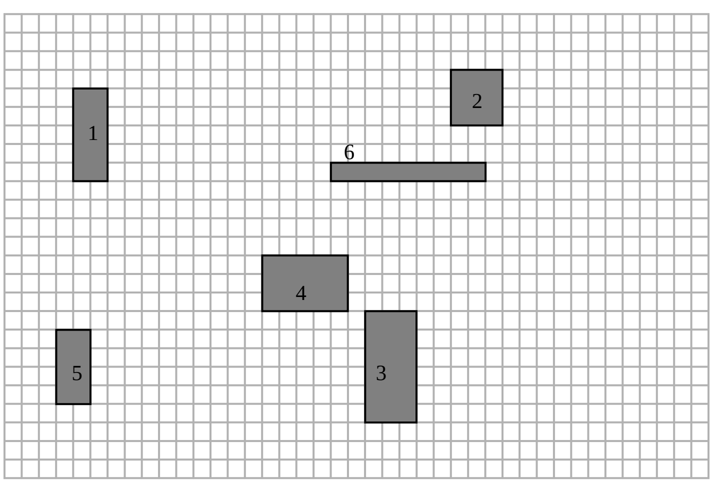




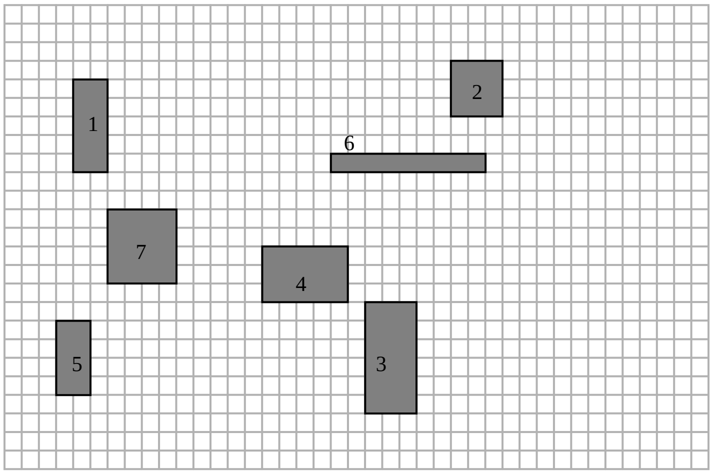
1
2
4
3

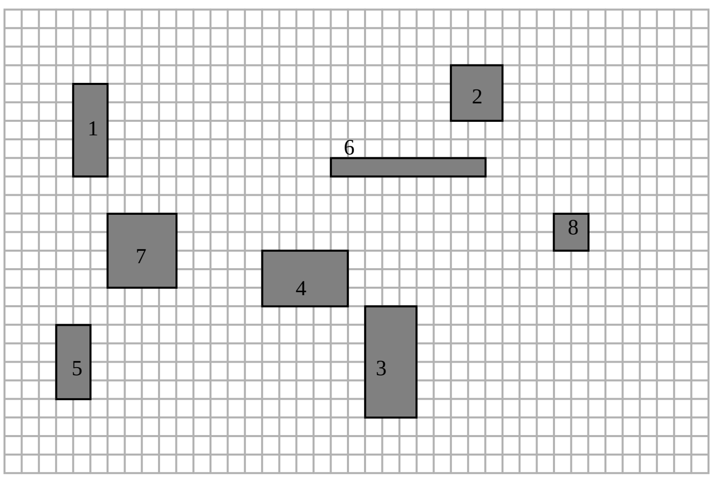


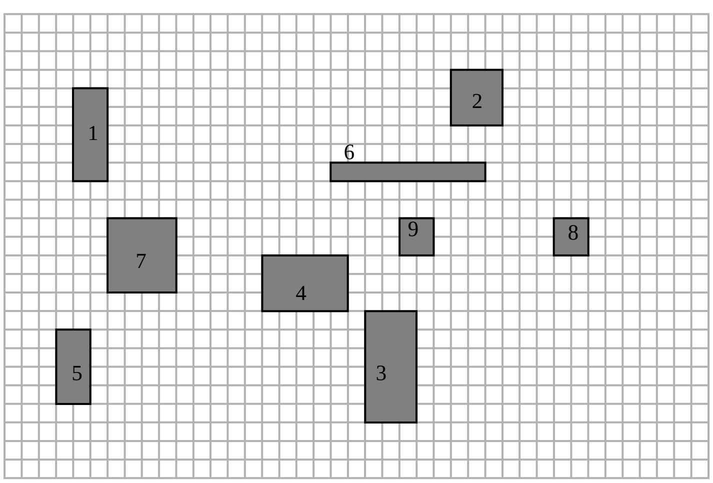
1
2
4
5
3



1
2
6
4
5
3


2
1
6
7
4
5
3


2
1
6
8
7
4
5
3


2
1
6
9
8
7
4
5
3

7) 15 pts
Given the relations EMP, ASG, and PROJ

EMP

ENO	ENAME	TITLE	SALARY
E1	J. Doe	Elec. Eng	95,000
E2	M. Smith	Syst. Anal.	105,000
E3	A. Lee	Mech Eng.	85,000
E4	J. Miller	Programmer	125,000
…			

ASG

ENO	PNO	RESP	DUR
E1	P1	Manager	12
E2	P2	Analyst	24
E3	P3	Analyst	6
…			10

PROJ

PNO	PNAME	BUDGET	Due Date
P1	Instrumentation	1,000,000	12-10-07
P2	Database Dev	2,000,000	10-1-06
P3	CAD/CAM	1,500,000	10-1-07
…			

And the query

```
SELECT    EMP.ENO
FROM      EMP  E, ASG  A, PROJ  P
WHERE     A.DUR > 14
          AND E.TITLE =  Programmer
          AND P.BUDGET < 1,500,000.
          AND P.PNO = A.PNO
          AND A.ENO = E.ENO
```

And the following assumptions:
- EMP relation is of size 1000
- ASG and PROJ relations are of size 20
- Durations are between 2 and 24 Weeks
- Budgets are between 100,000 and 2,000,000
- 25% of employees are Programmers

a- Specify an efficient order of operations to execute the above query if all data was held centrally. You can either present a query tree or a relational algebra expression. (State any other assumptions you make) b- Given the following distribution of relations across the network, give an efficient order of operations to execute the same query and return the results to site 1. Use a query tree and show all transfer of data from site to site.

- PROJ tuples are equally split between sites 1 and 2. Site 5 gets all PNO>40
- EMP tuples are equally split between sites 3 and 4. Site 1 gets all ENO>220
- ASG tuples are split between sites 5 and 6. Site 3 gets all those with ENO>220

Additional space

Additional space