بسم الله الرحمن الرحيم

Information and Computer Science Department
Fall Semester 071
ICS 324 – Database Systems
Sections 2 and 3
Final Exam
Monday, Jan 28, 2008
Time: 120 minutes

Name:

ID#:

Question	1	2	3	4	5	6	7	8	9	10	Total
Grade											

Question # 1 [10 points]
Consider the following relations:
STUDENT (SID, SName, Major, GPA)

COURSE (CN, CName, Hours)

REPORT (SID, CN, Grade) assume that grade is not null.

I) Given the following SQL query:
```
SELECT SID, SName, Major
FROM STUDENT
WHERE SID NOT IN(
        SELECT SID
        FROM REPORT);
```
Rewrite it using EXISTS.

II) List students (SID) who completed at least 50 credit hours.

Question # 2 [10 points]

Use heuristic rules to optimize the following query. Show the resulting tree after each of the five steps.

```
SELECT C.CN, C.CName, R.Grade
FROM REPORT R, COURSE C, STUDENT S
WHERE R.CN = C.CN AND R.SID = S.SID AND S.SName ='Yahya Garout';
```

Question # 3 [10 points]

Given the following relation:

ID	Name	D_Code	D_Name	GPA	C_Code	C_Name	Grade
231234	Ahmad	ICS	Computer Science	3.2	ICS102	Computer Programming	A
					MATH101	Calculus I	B
245678	Saad	COE	Computer Engineering	2.7	ICS102	Computer Programming	B
					PHYS101	Physics I	B

With the following functional dependencies:

Fd1: **ID → Name, D_Code, D_Name, GPA**

Fd2: **D_Code → D_Name**
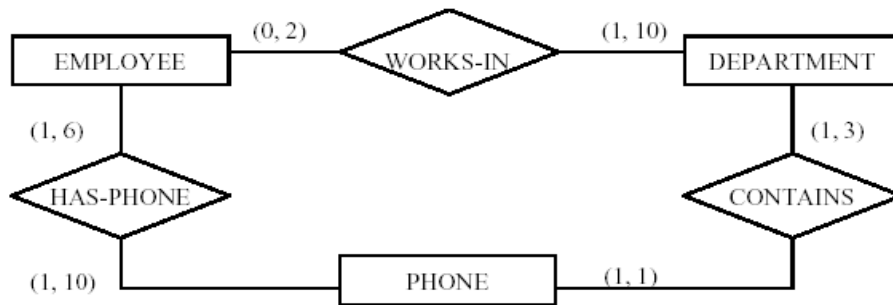
Fd3: **C_Code → C_Name**

Fd4: **ID, C_Code → Grade**

Normalize the above relation up to 3NF only. Show your relations after 1NF, 2NF, and 3NF

Question # 4 [5 points]

Consider the relation R {A, B, C, D} with two candidate keys {A, B} and {C}, List all the super keys.

Question # 5 [5 points]

In the following ER diagram, convert the (min, max) constraints into cardinality and participation constraints:

```
              (0, 2)                    (1, 10)
EMPLOYEE  ─────────  <WORKS-IN>  ─────────  DEPARTMENT

(1, 6)                                              (1, 3)

<HAS-PHONE>                                    <CONTAINS>

(1, 10)  ───────  PHONE  ───────  (1, 1)
```
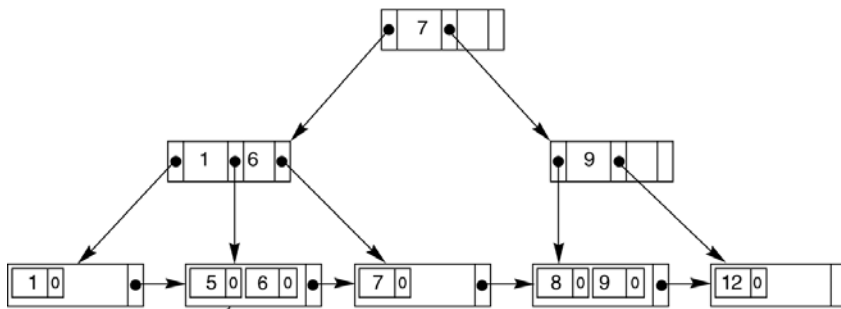
Question # 6 [10 points]

A PARTS file with Part# as hash key includes records with the following Part# values: 4, 6, 2, 7 and 23. Each bucket is one disk block and holds two records. Load these records into expandable hash files based on extendible hashing. Show the structure of the directory at each step. Show the directory at each step, and the global and local depths using the hash function h (K) = K mod 8.

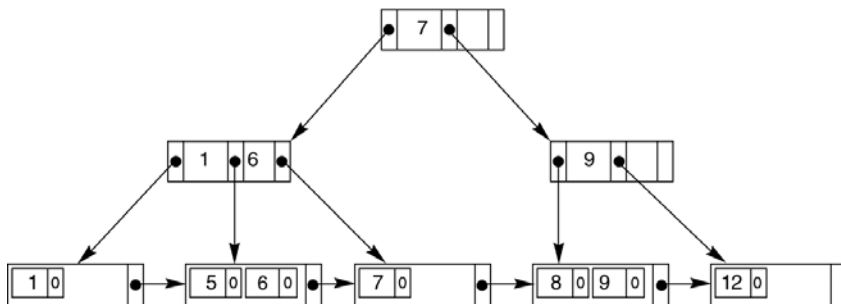| K	h(k)	binary
4	4	100
6	6	110
2	2	010
7	7	111
23	7	111

Question # 7 [5 points]

Show the tree after the value 2 is inserted into the following B+ tree of order p=3 and p leaf =2.

Root: [• | 7 | • |]

Internal left: [• | 1 | • | 6 | •]   Internal right: [• | 9 | • |]

Leaves: [1 | 0 | •] → [5 | 0 | 6 | 0 | •] → [7 | 0 | •] → [8 | 0 | 9 | 0 | •] → [12 | 0]

Question # 8 [5 points]

Show the tree after the value 1 is deleted from the following B+ tree of order p=3 and p leaf =2.

Root: [• | 7 | • |]

Internal left: [• | 1 | • | 6 | •]   Internal right: [• | 9 | • |]

Leaves: [1 | 0 | •] → [5 | 0 | 6 | 0 | •] → [7 | 0 | •] → [8 | 0 | 9 | 0 | •] → [12 | 0]

Question # 9 [5 points]

What are the design decisions about indexing in physical database design?

Question # 10 [10 points]

For both schedules C and D, find if they are serializable and why?

T_1	T_2
read_item(X); X:=X-N;	
	read_item(X); X:=X+M;
write_item(X); read_item(Y);	
	write_item(X);
Y:=Y+N; write_item(Y);	

Schedule C

T_1	T_2
read_item(X); X:=X-N; write_item(X);	
	read_item(X); X:=X+M; write_item(X);
read_item(Y); Y:=Y+N; write_item(Y);	

Schedule D

Time