

浙江大学 2018 - 2019 学年 春夏 学期

《数据库系统》课程期末考试试卷（A 卷）

参考答案及评分细则

Answers of Problem 1:

(16 points, 4 points per part)

1) $\Pi_{\text{Title}}(\sigma_{\text{director}=\text{'Yimou Zhang'}}(\text{movie}) \bowtie \sigma_{\text{grade} \geq 4}(\text{comment}))$

2) Update comment set grade=0 where grade is null

3) Select type from movie, comment

Where movie.title=comment.title

Group by title

Having avg(grade) >=all (Select avg(grade)

From movie, comment

Where movie.title=comment.title

Group by title)

4) Select title from movie

Except

Select title from movie

Where exists (select *

From comment A, comment B

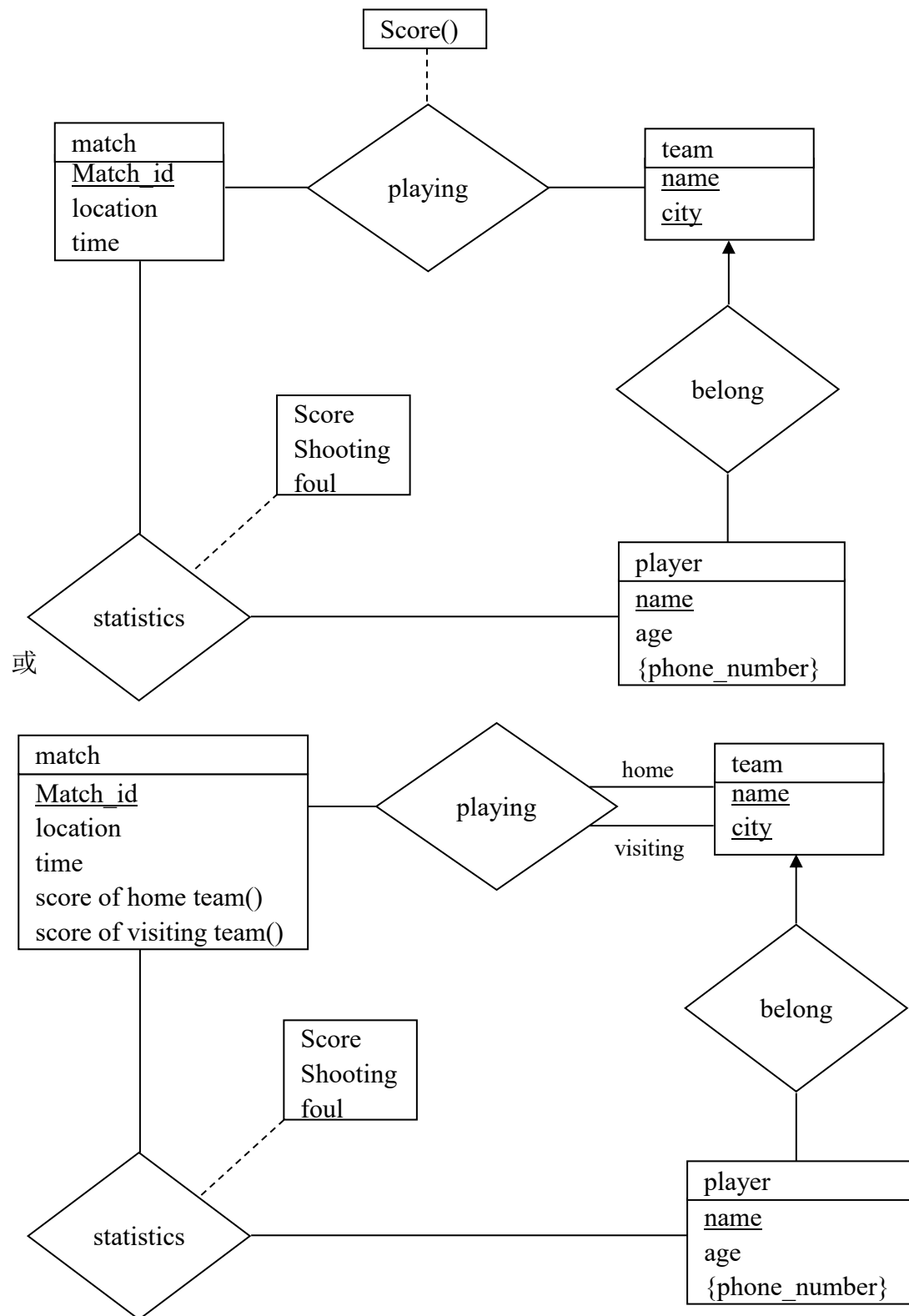
Where A.title=movie.title and A.user_name = B.user_name

And B.title=' the avenger'

And A.grade <=B.grade)

Problem 2: E-R Model (9 points)

1) (5 points)



2) (4 points)

Match(match_id, location, time)

Team(name, city)

Playing(match_id, name, score)

Player(name, age)

Phone(player_name, phone_number)

Statistics(match_id, player_name, score, shooting, foul)

或者其中的 match 和 playing 改为:

Match(match_id, location, time, home_team_name, visiting_team_name,
score_of_home_team, score_of_visiting_team)

Each match has one home team and one visiting team.

Problem 3: Relational Formalization (12 points, 4 points each)

1) {C E}

2) Decompose R into R1(A, B) and R2(A, C, D, E), decompose R2 into R21(A, C) R22(C, D, E), and further decompose R22 into R221(C, D) and R222(C, E)

3) The decomposition is dependency preserving.

Problem 4: XML (12 points, 4 points each)

1)

<!DOCTYPE movie_comment[

```

<!ELEMENT      movie_comment ( movie*)>
<!ELEMENT      movie (type, director, comment+)>
<!ATTLIST      movie title ID #REQUIRED>
<!ELEMENT      type (#PCDATA)>
<!ELEMENT      director (#PCDATA)>
<!ELEMENT      comment (user_name, grade)>
<!ELEMENT      user_name (#PCDATA)>
<!ELEMENT      grade (#PCDATA)>
|>

```

评分细则:

错 1-2 处扣 1 分，较多错误酌情扣 2-3 分

2) `/movie_comment/movie[type="action" and ./comment/user_name="Alice" and ./comment/grade=5]/@title`

评分细则:

漏一个条件扣 1 分，路径错误扣 1 分

3) `for $p in /movie_comment/movie[director="Yimou Zhang"]
 where count($p/comment[grade=5])>=1
 return $p/@title`

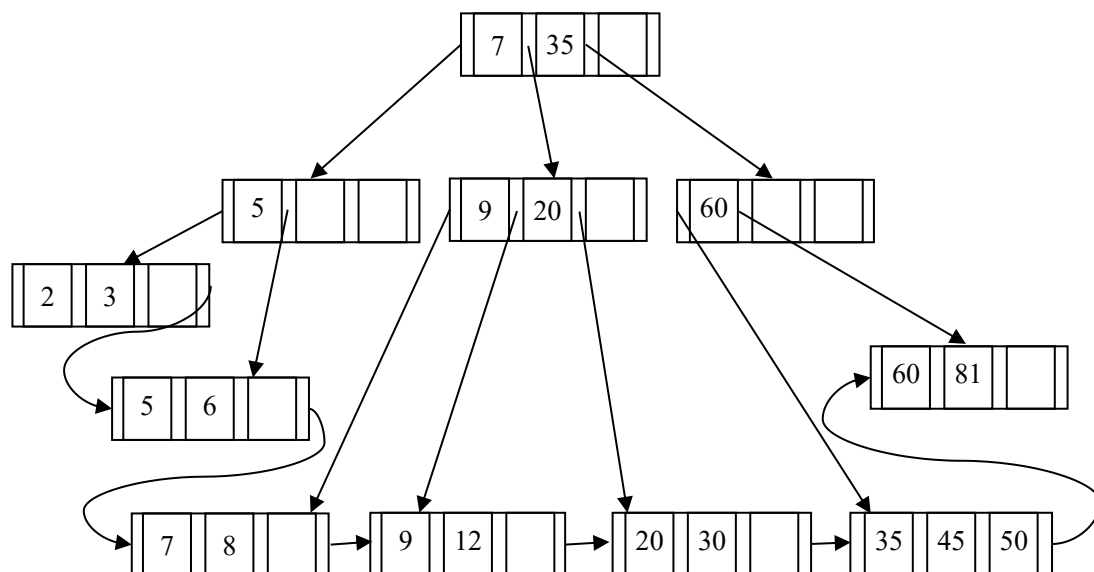
评分细则:

少一个条件扣一分，逻辑错误扣 2-3 分

Problem 5: B⁺-Tree (12 points, 3 points each)

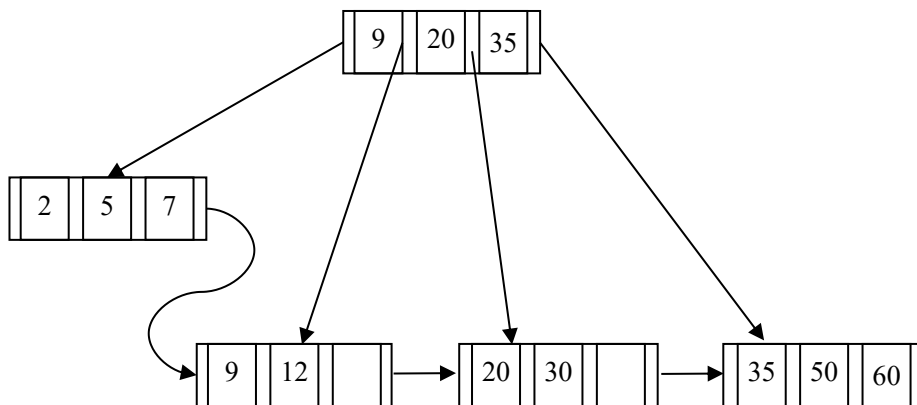
1)

After inserting 8, 6 and 3:



2)

After deleting 81 and 45:



3) Maximal number of key values: $4*4*4*4*3=768$

Minimal number of key values: $2*2*2*2*2=32$

4) $(3 + 1) + 1 = 5$ 或 $(3 + 1) + 2 = 6$

Problem 6: Query Processing (12 points, 4 points each)

1) $5,000/500/5 = 2$

2) Number of blocks of movie is $5000/50=100$

Number of blocks of comment is $1,000,000/100=10,000$

Since the equi-join attribute title forms a key on inner relation, we can stop inner loop on the first match.

Assign 10 blocks to comments, 1 block to movies, and 1 block for output.

Number of block accesses: $(10000/10)*100+10000 = 110000$ 或

$$10000 * 100/10 + 100 = 100100$$

Number of seeks: $2*10000/10=2000$

3) Minimal height = $\log_{60}(5000) \rightarrow 3$ (向上取整)

Max height = $\log_{30}(5000) \rightarrow 3$ (向上取整)

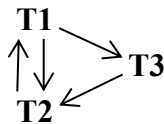
So, the height of the B⁺-tree index on movie(title) is 3.

Number of block accesses: $10000+1000000/500*3+1$

Number of seeks: $10000+1000000/500*3+1$

Problem 7: Concurrency Control (12 points, 4 points each)

1)



The schedule is not serializable, because there are cycles in the graph.

2) The schedule is not cascadeless.

3) No. This is because the schedule in 1) exists cycles.

Problem 8: Aries Recovery Method (15 points, 3 points each)

1) 1002

2) 1010

3) T4

4) “102.1” = 62, “102.2” = 73

5)

1015: <T4, 102.1, 62>

1016: <T4, abort>