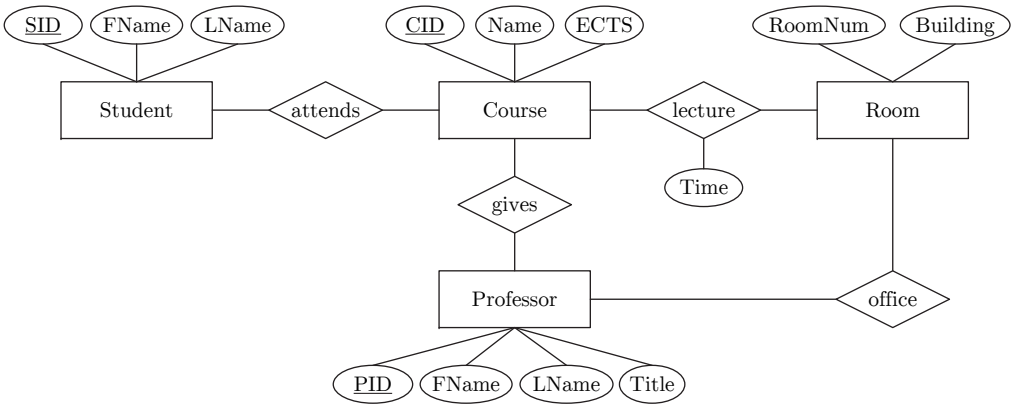
Univ.-Prof. Dr.-Ing. Matthias Boehm
Graz University of Technology
Computer Science and Biomedical Engineering
Institute of Interactive Systems and Data Science
BMVIT endowed chair for Data Management

September 16, 2019

Exam INF.01014UF Databases (Summer 2019, V3a)

Important notes: The working time is 90min, and lecture materials or any kind of mobile devices are not allowed. Please make sure to put your name and matriculation number on the top right of each piece of paper. You may give the answers in English or German, as well as directly write into the task description.

Task 1 Data Modeling (25 points)



(a) Given the above Entity-Relationship diagram, specify the cardinalities in Modified Chen notation based on the following information (if some aspect is unspecified, please model the more general case). (**8 points**)

- A student attends zero or many courses and a course is attended by at least 3 students.
- A course is given by exactly one professor and a professor might give between 1 and 7 courses.
- A course takes place in exactly one room, but many courses might be taught in the same room (at different times).
- Every professor has an office room, and professors never share offices.

(b) Map the given Entity-Relationship diagram into a relational schema, including data types, primary keys, and foreign keys. (**10 points**)

(c) Assume an additional multi-valued attribute Professor.Phone (lists of phone numbers). Bring your schema in third normal form, describe the schema modifications (delta) or the full schema, and explain why it is in third normal form. (**7 points**)

Task 2 Structured Query Language (30 points)

Employees

EID	FName	LName	Age	PID
4	Isabella	Brown	30	2
2	Olivia	Johnson	30	1
1	Emma	Smith	35	3
3	Ava	Williams	20	1
5	Sophie	Jones	35	2
6	Taylor	Miller	55	4
7	Charlotte	Davis	40	2

Projects

PID	Name	Customer
1	UX Design	B
2	App Backend	B
3	Data Storage	A
4	ML Pipeline	C
5	UX Design	C

(a) Given the Employees and Projects tables above, and compute the results for the following three queries: (**15 points**)

```
Q1: SELECT DISTINCT P.Customer, P.Name
      FROM Employees E, Projects P
      WHERE E.PID = P.PID
        AND E.FName IN('Ava','Emma','Sophie')
```

```
Q2: SELECT Age, count(*) FROM Employees
      GROUP BY Age
      ORDER BY count(*) DESC, Age ASC
```

```
Q3: SELECT P.Name, round(avg(E.Age)) --avg=sum/count
      FROM Employees E, Projects P
      WHERE E.PID = P.PID
      GROUP BY P.Name
```

(b) Given the Employees and Projects table schemas above, write SQL queries to answer the following questions: (**15 points**)

- Q4: Which employees work on projects for customer B (the SQL query should return the first names and last names)?

- Q5: Which customers have more than one project (the SQL query should return the distinct customer names)?

- Q6: What is the age range of employees per project other than UX Design (the SQL query should return the age range in years for each project)?

Task 3 Query Processing (20 points)

(a) Assume tables R(a,b), and S(c,d,e), draw a logical query tree in relational algebra for the following query: (**5 points**)

```
Q7: SELECT R.a, S.d FROM R, S
      WHERE R.b = S.c AND S.e > 3
    INTERSECT
    SELECT R.a, S.d FROM R, S
      WHERE R.b = S.c AND S.e < 7
```

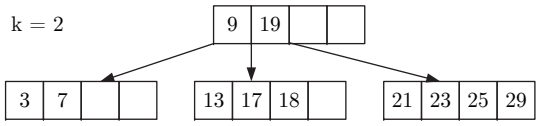
(b) Draw an optimized logical query tree for the above query in relational algebra by eliminating the intersect operation. (**3 points**)

(c) Define the derived relational operators (1) join and (2) set intersection via basic relational operators (cartesian product, union, difference, projection, selection). (**4 points**)

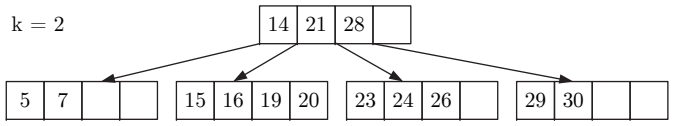
(d) Describe the conceptual idea of a hash join operator, including the resulting time and space complexity and how this operator fits into the Volcano iterator model. (**8 points**)

Task 4 Physical Design (15 points)

(a) Given the B-tree below, insert the key 20 and draw the resulting B-tree. (**6 points**)



(b) Given the B-tree below, delete the key 14 and draw the resulting B-tree. (**4 points**)



(c) Briefly discuss the concept of horizontal table partitioning. Recall table Employees R from Task 2, and provide relational algebra expressions for (1) horizontally partition R into R_1 and R_2, and (2) reconstruct R from R_1 and R_2. (**5 points**)

Task 5 Stream Processing (10 points)

(a) Explain the basic system architecture and execution model of stream processing engines with a special focus on continuous queries and window semantics. (**4 points**)

(b) Describe means (i.e., techniques) for handling overload situations in stream processing engines, where more input tuples arrive than the system can process. (**6 points**)