
DEPARTMENT OF INFORMATICS

Prof. Dr. Michael Böhlen

Binzmühlestrasse 14

8050 Zurich

Phone: +41 44 635 4333

Email: boehlen@ifi.uzh.ch



University of
Zurich^{UZH}

Temporal and Spatial Data Management
Fall 2017

Final Exam
January 8, 2018

Name: _____ Matriculation number: _____

Advice

You have 60 minutes to complete the exam. The following rules apply:

- Answer the questions on the exam sheets or the backside. Mark clearly which answer belongs to which question. Additional sheets are provided upon request. If you use additional sheets, put your name and matriculation number on each of them.
- In your solutions you can abbreviate names of relations and attributes (first character is sufficient).
- Check the completeness of your exam (?? numbered pages).
- Use a pen in blue or black color for your solutions. Pens in other colors and pencils are not allowed.
- Stick to the terminology and notations used in the course.
- A foreign dictionary is allowed. The dictionary will be checked by a supervisor.
- No additional items are allowed, notably calculators, computers, pdas, smart-phones, audio-devices or similar devices may not be used. Any cheating attempt will result in a failed test (meaning 0 points and a grade of 1).
- Make sure to hand in all sheets at the end of your exam.
- Put your student legitimization card ("Legi") on the desk.

Signature:

Correction slot

Please do not fill out the part below

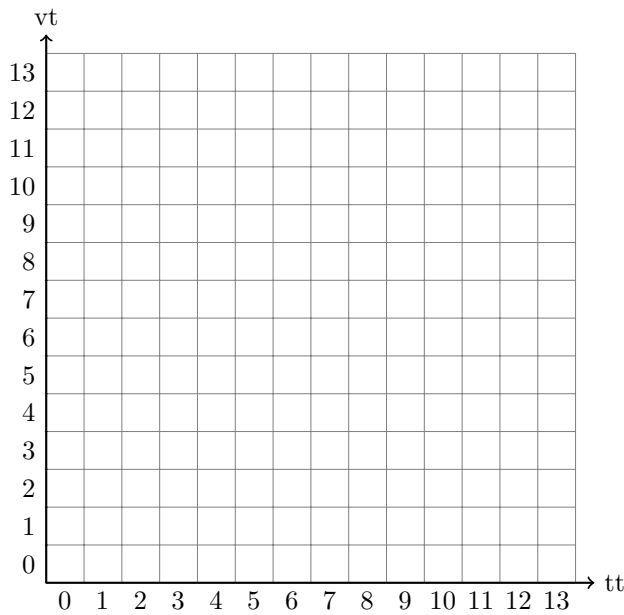
Exercise	1	2	3	4	Total
Points Achieved					
Maximum Points	15	15	15	15	60

Task 1**15 points**

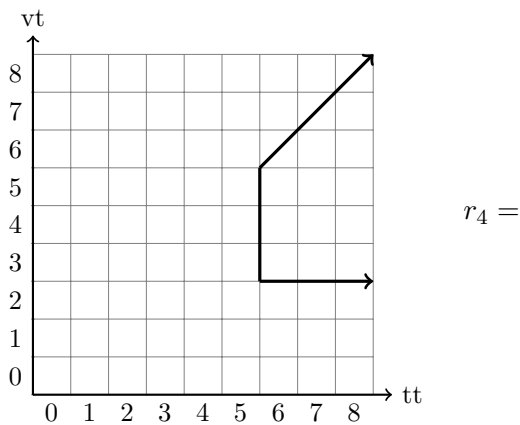
Consider relation R with non-temporal attribute X , transaction time interval attribute TT , and valid time interval attribute VT :

R			
	X	TT	VT
r_1	a	$6 - now$	$5 - 11$
r_2	b	$3 - 5$	$now - 10$
r_3	c	$8 - now$	$now - 12$

Task 1.1 (6 points): Use the Cartesian plane below to graphically illustrate the timestamps of the tuples in R .



Task 1.2 (2 points): Add a tuple to relation R that would yield the following timestamp:



Name:

Matriculation number:

Task 1.3 (3 points): Assume there are no applications that use intervals where the start point is equal to *now*. Explain why the DBMS should still support time intervals where the start point may be equal to *now*. Give an SQL example that illustrates the issue.

Task 1.4 (4 points): A common approach that DBMSs use to deal with *now* is to instantiate *now* to the current time whenever it is accessed. List two important advantages and two important disadvantages of such a solution.

Task 2**15 points**

A non-temporal semi-join between two relations returns the rows from the first relation for which there is at least one matching row in the second relation. Algebraically, a non-temporal semi-join between $R(A_1, \dots, A_k, C_1, \dots, C_m)$ and $S(C_1, \dots, C_m, B_1, \dots, B_l)$, where $A_i \neq B_j$, can be defined with the help of a projection and natural join as follows:

$$R \ltimes S = \pi_{A_1, \dots, A_k, C_1, \dots, C_m}(R \bowtie S)$$

Consider relations *Enrollment* and *Lecturer* as follows:

Enrollment				Lecturer			
	Student	Module	VT		Module	Professor	VT
e_1	s1	SE	5 – 8	l_1	SE	Fritz	6 – 12
e_2	s1	RE	12 – 15	l_2	RE	Glinz	10 – 13
e_3	s1	DB	12 – 16	l_3	RE	Seyff	14 – 18
e_4	s2	SE	14 – 20				

Task 2.1 (3 points): Write down the result of the temporal semi-join $Enrollment \ltimes^T Lecturer$. The temporal semi-join must satisfy three properties: snapshot reducibility, change preservation, and extended snapshot reducibility.

Task 2.2 (4 points): Formally define snapshot reducibility and extended snapshot reducibility for the temporal semi-join $Enrollment \ltimes^T Lecturer$. For each definition state its intuitive meaning in natural language.

Name:

Matriculation number:

Task 2.3 (5 points): The lineage set for the temporal semi-join is defined as follows:

$$\begin{aligned} \mathsf{L}[R \ltimes^T S](z, t) = \{ \langle r, s \rangle \mid & r \in R \wedge t \in r.T \wedge \\ & s \in S \wedge t \in s.T \wedge s.\mathbf{C} = r.\mathbf{C} \wedge \\ & z.\mathbf{A} = r.\mathbf{A} \wedge z.\mathbf{C} = r.\mathbf{C} \} \end{aligned}$$

Explain the intuition of the definition in natural language:

Determine the lineage set for $E \ltimes^T L$:

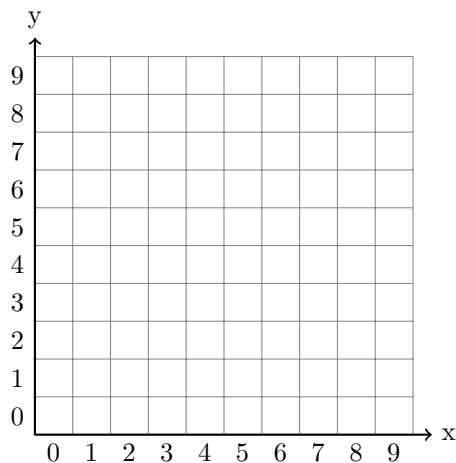
Task 2.3 (3 points): Assume the temporal semi-join does not satisfy extended snapshot reducibility. Illustrate with an algebraic example to what problems this would lead.

Task 3**15 points**

Consider relation Q (attribute P is a 2D point with an x and y coordinate):

Q	
W	P
a	(1,2)
b	(3,5)
c	(6,2)
d	(4,2)
e	(6,5)
f	(3,9)

Task 3.1 (6 points): Show a kd-tree for relation Q . Assume that all leaf pages have a capacity of two entries. Illustrate the kd-tree in the Cartesian plane and as a tree. Split the y dimension first. On each split plane at least one database point must lie.

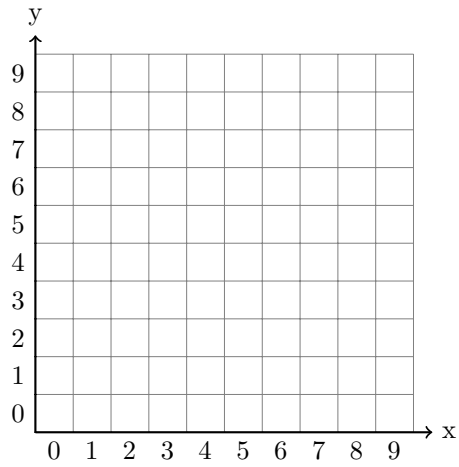


Task 3.2 (3 points): What happens if at each step you choose the median to split the space when constructing a kd-tree? State one advantage and one disadvantage when choosing the median.

Name:

Matriculation number:

Task 3.3 (6 points): Construct an R-tree for relation Q . Illustrate the R-tree in the Cartesian plane and as a tree. Assume that all nodes may hold up to two entries.



Task 4**15 points**

Consider relation R with reservations:

R

Name	Country	Period
Miller	USA	2017/12/22 – 2017/12/29
Hauser	CH	2017/12/22 – 2017/12/23
Gubler	CH	2017/12/23 – 2018/01/07

Hint: In algebra expressions and SQL statements you can write $upper(Period)$ and $lower(Period)$ to access start and end point of a period, respectively.

Task 4.1 (6 points): Construct algebraic expressions that determine for each country and each point in time the average length of the active reservations. Proceed as follows:

- First, formulate a temporal relational algebra expression to answer the query.
- Next, transform the temporal relate algebra expression into a non-temporal algebra expression that uses the adjustment primitives. Show the transformation for each step and explain the steps.

Name:

Matriculation number:

Task 4.2 (4 points): Formulate an SQL query that determines for each country and each point in time the average length of the active reservations (use SQL extended with operations for interval adjustment).

Task 4.3 (5 points): Determine and explain the space complexity of temporal normalization, i.e., determine the upper bound for the number of tuples in $\mathcal{N}(R, S)$.

Hint: Construct an example where each tuple in S splits as many tuples in R as possible. Use this example to derive the upper bound.

Reduction rules for temporal algebra operators:

Operator	Reduction rule	
Selection	$\sigma_{\theta}^T(R)$	$= \sigma_{\theta}(R)$
Projection	$\pi_{\mathbf{B}}^T(R)$	$= \pi_{\mathbf{B},T}(\mathcal{N}_{\mathbf{B}}(R, R))$
Aggregation	$\mathbf{B} \vartheta_F^T(R)$	$= \mathbf{B},T \vartheta_F(\mathcal{N}_{\mathbf{B}}(R, R))$
Difference	$R \text{--}^T S$	$= \mathcal{N}_{\mathbf{A}}(R, S) - \mathcal{N}_{\mathbf{A}}(S, R)$
Union	$R \cup^T S$	$= \mathcal{N}_{\mathbf{A}}(R, S) \cup \mathcal{N}_{\mathbf{A}}(S, R)$
Intersection	$R \cap^T S$	$= \mathcal{N}_{\mathbf{A}}(R, S) \cap \mathcal{N}_{\mathbf{A}}(S, R)$
Cartesian Product	$R \times^T S$	$= \alpha(\phi_{\top}(R, S) \bowtie_{R.T=S.T} \phi_{\top}(S, R))$
Inner Join	$R \bowtie_{\theta}^T S$	$= \alpha(\phi_{\theta}(R, S) \bowtie_{\theta \wedge R.T=S.T} \phi_{\theta}(S, R))$
Left Outer Join	$R \Join_{\theta}^T S$	$= \alpha(\phi_{\theta}(R, S) \Join_{\theta \wedge R.T=S.T} \phi_{\theta}(S, R))$
Right Outer Join	$R \Join_{\theta}^T S$	$= \alpha(\phi_{\theta}(R, S) \Join_{\theta \wedge R.T=S.T} \phi_{\theta}(S, R))$
Full Outer Join	$R \Join_{\theta}^T S$	$= \alpha(\phi_{\theta}(R, S) \Join_{\theta \wedge R.T=S.T} \phi_{\theta}(S, R))$
Anti Join	$R \triangleright_{\theta}^T S$	$= \phi_{\theta}(R, S) \triangleright_{\theta \wedge R.T=S.T} \phi_{\theta}(S, R)$