# Introduction to Databases

## Tutorial 2

Dr Paolo Guagliardo

Fall 2020 (week 4)

**Problem 1 (mandatory).** Consider the following schema:

CUSTOMER( · , · , · ) of arity 3, where the first position is the customer's *ID*, the second is the customer's *Name*, and the third is the *City* where the customer lives.

ACCOUNT ( · , · , · , · ) of arity 4, in which the first position is the account's *Number*, the second is its *Branch*, the third is the ID of the customer owning the account, and the fourth is the *Balance* on the account.

Write the following queries in relational calculus:

(1) "ID and name of customers who own an account in a branch in their city."

(2) "ID and name of customers who do **not** own any account."

(3) "ID and name of customers who own an account with a balance which is no less than the balance of any other account."

*Note:* Write the queries directly in relational calculus, without translating from relational algebra.

**Problem 2 (optional).** Given a schema consisting of a binary relation $R$ and a ternary relation $S$, write a relational calculus query that computes the active domain.

**Problem 3 (mandatory).** Consider the schema of Problem 1 and assume that CUSTOMER is over attributes *ID*, *Name*, *City* (in this order) and ACCOUNT is over attributes *Number*, *Branch*, *CustID*, *Balance* (in this order). Express the following relational algebra query in relational calculus:

$$\text{CUSTOMER} \bowtie \big(\pi_{\text{ID,City}}(\text{CUSTOMER}) \cap \rho_{\text{CustID}\to\text{ID},\,\text{Branch}\to\text{City}}\big(\pi_{\text{Branch,CustID}}(\text{ACCOUNT})\big)\big) \tag{1}$$

*Note:* Use the translation rules from RA to RC we have seen in class.

**Problem 4 (optional).** Given a relation $R$ over attribute $A$, and a relation $S$ over attributes $A, B$ (in this order), translate the following relation calculus query to relational algebra:

$$\{\, x, z \mid R(x) \wedge \neg\exists y\; S(y, z) \,\}$$