

This is a closed book exam: the only resources allowed are blank paper, pens, and your head, but you may use a handwritten A4 page with information that you consider useful for solving the exam exercises. Explain your reasoning. Write clearly, in the sense of logic, language, and legibility. The clarity of your explanations affects your grade. Good luck!

Write your name and student number on *all* solution sheets and here.

Name:

At the end of the exam, hand in *all* sheets that you received, including this one.

Student number:

Problem 1 [30%] Design the Entity-Relationship schema of an application for the Time Travel Agency, which is interested in maintaining the information about time travel (TT) visits carried out by time travelers to different epochs. Each *epoch* is identified by a year, and we are also interested in its description. Each *time traveler* is identified by their tta-code, and they have a name and a date of birth. There are exactly two types of time travelers: *ordinary time travelers* (OTTs), who participate in TT-visits, and they must have participated in at least one; and *specialized time travelers* (STTs), who have a specialization and who may lead TT-visits (but otherwise do not participate in them). Each *TT-visit* is led by exactly one STT, is to a certain epoch, and has a start-date and a location. Notice that there may not be two TT-visits with the same start-date led by the same STT. As mentioned, we are interested in the OTTs who participate to a TT-visit (who may be zero, one, or more). The Time Travel Agency also maintains artifacts that have been obtained from TT-visits, where each *artifact* is identified by an id that is unique within the TT-visit from which it has been obtained. Of artifacts, we are also interested in their value and in the time travelers who are studying them.

Problem 2 [40%] Carry out the logical design of the database, producing the complete relational schema with constraints, taking into account the following indications: (i) We access STTs separately from OTTs. (ii) Every time we access a TT-visit, we always want to know the epoch to which it was done.

In your design you should follow the methodology adopted in the course, and you should produce:

1. [7%] the restructured Entity-Relationship schema (possibly with external constraints),
2. [25%] the direct translation into the relational model (possibly with external constraints), and
3. [8%] the restructured relational schema (again with constraints).

You should motivate explicitly how the above indications affect your design.

Problem 3 [15%] Consider a database D containing the two relations:

- (i) $\text{Teacher}(\text{code}, \text{cityOfBirth}, \text{dateOfBirth})$, which stores the code (primary key), the city of birth, and the date of birth of a set of teachers;
- (ii) $\text{Competition}(\text{teacherCode}, \text{year}, \text{score})$, which stores the code of the teacher, the year, and the score obtained in a university competition in which the teacher has participated (the rule is that no teacher can participate in university competitions more than once in a year).

We know that when the score for a competition for a teacher is not known, the attribute *score* for that competition stores the null value, while in the other attributes of the two relations, null values never appear. We also know that a foreign key constraint is defined from *teacherCode* in *Competition* to *code* in *Teacher*.

Write in **SQL** the following queries over D :

1. Compute the *code* of all teachers, each with the *number of competitions* in which they have participated.
2. For each teacher born in Bolzano, compute the *code* and the *minimum score* obtained in any competitions they have participated in.
3. For those teachers who have participated in at least 5 competitions, compute the *code*, the *date of birth*, and the *average score* obtained in the competitions in which the score is known.

Problem 4 [15%] Consider the conceptual schema S shown below and carry out the logical design starting from S , considering that you should produce the logical schema (with integrity constraints) with as few relations as possible.

