This is a closed book exam: the only resources allowed are blank paper, pens, and your head. 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 information system relating to food supply companies. Of each *company*, we are interested in the code (identifier), the name, the year of foundation, the person who directs it and the dishes that the company has in its catalog. Each *dish* is identified by a code and is characterized by the name and by the steps of the recipe used to prepare it (at least one). Each *recipe step* of a dish is identified by a sequence number and has associated a description. There are exactly 4 types of dishes: first course, second course, dessert, and other. Of each *first course* we are interested in the amount of carbohydrates it contains, of each *second course* in the type of protein it contains, of each *dessert* in the amount of sugar it contains, and of each *other dish* in the amount of calories it contains. Each *person* is identified by the tax code and is characterized by their name, surname, and date of birth. *Customers* are persons who book meals with food supply companies, and we are interested in their current occupation. When a customer books a meal of a certain type (for example, lunch, dinner, snack, aperitif, etc.) with a company for a certain date, the company establishes the cost of the *booked meal*. Please note that a customer cannot book more than one meal of the same type with the same company and for the same date. There are two types of booked meals, depending on the set of dishes they are composed of: fixed menu meals and free menu meals. The *fixed menu meals* include a first course and a second course, while the *free menu meals* can include any set of dishes (at least one). A customer can subscribe to a company. Of each *subscription*, the start date is of interest, and also the end date, but only if the subscription has been terminated. Notice that a customer can subscribe at most once to the same company in a given period. For meals booked by customers who are currently subscribers of the company at which they booked the meal, we are interested in the discount applied to the meal.

Problem 2 [42%] Carry out the logical design of the database, producing the complete relational schema with constraints, taking into account the following indications: *(i)* We want to avoid null values in the database. *(ii)* When accessing a fixed menu meal, we always want to know which are the corresponding first course and second course.

As steps in your design you should produce:

1. [7%] the restructured ER schema (possibly with external constraints),
2. [25%] the direct translation into the relational model (possibly with external constraints), and
3. [10%] the restructured relational schema (again with constraints).

Motivate explicitly how the above indications affect your design.

Problem 3 [18%] Consider the relation `Votes(newspaper,player,day,score)`, each tuple of which records the score assigned by a newspaper to a player in a certain day of the championship.

1. Let us call "top player" a player who in all the days of the championship had an average score (average calculated on all newspapers in that day) equal or higher than the average of the other players in the same day. Write a SQL query that calculates all top players.

2. Describe in words what the following query computes:

```
SELECT DISTINCT V1.player
FROM Votes V1
WHERE 7 <= ALL(SELECT score  FROM Votes V2  WHERE V2.player = V1.player) AND
      NOT EXISTS (SELECT V3.player  FROM Votes V3
                  WHERE V3.day = V1.day AND V3.newspaper = V1.newspaper AND
                        V3.score < V1.score)
```

Problem 4 [10%] Consider the conceptual diagram S shown below. Provide the list of all pairs $\langle e_1, e_2 \rangle$ of entities of S such that e_1 and e_2 are disjoint, i.e., such that in each instance I of schema S, the set $instances(I, e_1)$ of instances of e_1 in I is disjoint from the set $instances(I, e_2)$ of instances of e_2 in I.