T-106.270 Basic Course in Programming T3

Examination, May 9th, 2003

1. (12p) Give a short answer to following questions (max. 100 words per question):

   (a) What do the terms *static* and *dynamic* mean in the context of programming languages. Give some examples related to object-oriented languages.

   (b) How do messages in object-oriented programming differ from procedure calls in traditional programming languages?

   (c) What are *cohesion* and *coupling* of classes? What good and bad varieties of cohesion and coupling do you know?

   (d) What is the difference between *overloading* and *overriding*?

   (e) What is a *polymorphic variable* and *pure polymorphism*?

   (f) How do the following programming abstractions differ from each other: a *module*, an *abstract data type*, and a *class*?

2. (6p) Essay. Explain the concepts of a *subclass*, a *subtype*, and the *substitution principle* and describe the connections and differences between them. What is the meaning of these concepts in object-oriented programming languages and their implementation as well as in object-oriented programming?

3. (6p) Design exercise. *Product structures* are used for presenting data about products like computers, cars, refrigerators, and pieces of furniture. A product structure presents how the product is divided into components and parts. For instance, a chair may consist of legs, a seat, a back, screws, etc. These parts may consist of smaller parts. The product structure is often drawn as an exploded view (see the picture).

   Your task is to design an object-oriented model for presenting product structures. The model should be capable for doing the following kind of tasks:

   (a) List the parts that are needed for assembling a product.

   (b) What product does a given part belong to? In which products are parts of a given type used?

   (c) How much does the parts of a product cost? The prices of the simplest parts are known.

   (d) What is the weight of a product or some part of it? The weights of the simplest parts are known.

   (e) Which subcontractors are needed in the production of a product?

   (f) Drawing a picture of a product.

   Describe (in English) what classes, objects, properties, relationships, and other structures are needed? In addition to the description, present the classes either in UML, Java, or C++. You do not have to present all details of the classes.

   Describe methods for doing some of the tasks listed above.

   Try to justify your design.

   Use some product as an example, but remember that your approach should be applicable for presenting also the structure of other products.

4. (6p) Concept map. The accompanying concept map tries to present object-oriented concepts and relationships between these concepts. How could you improve the map? Which concepts or relationships would you add, remove, or change? The name of some concepts and relationships are missing; these are replaced by a question mark. How would you name them?