

CS_M71 2004/05
Design Patterns and Generic Programming

(Attempt 2 questions out of 3)

Question 1 The Visitor design pattern

- (a) Assume that you have a class hierarchy with base class **B** and derived classes $D_1 \dots D_k$. You want to apply an algorithm f to objects of type D_i . Describe solutions using
- (i) traits classes
 - (ii) freestanding functions
 - (iii) virtual member functions
 - (iv) the visitor design pattern.

Discuss the trade-offs of these approaches (regarding functionality, extensibility and efficiency), and which solution to choose under which circumstances.

[12 marks]

- (b) Consider the problem that within some algorithm a there are event points e_1, \dots, e_n where customisable operations f_1, \dots, f_n are to be called. Describe the approach to this problem via the template design pattern, and how to transform this pattern into (basically) the pattern of an algorithmic visitor. Discuss the different options.

[8 marks]

- (c) Discuss the relation of an algorithmic visitor to the problem and the four solutions discussed under (a).

[5 marks]

Question 2 Polymorphism and generic programming

(a) Forms of polymorphism

- (i) Explain the forms of polymorphism we discussed in the lecture, give examples for each case, and explain the examples.

[9 marks]

- (ii) Explain how “tagging classes” are used to avoid a combinatorial explosion of concepts, and how we can make functions (freestanding or members) be aware of this kind of “tagging polymorphism”.

[6 marks]

- (b) Consider the task of finding in a sequence some element fulfilling a certain property.

- (i) Design a generic algorithm `find_if` for this task (using iterators).

[6 marks]

- (ii) In general, searching through a list of n objects needs n tests. However, if we can use some order on the sequence and binary search becomes applicable, then we can search much faster. How can you make your generic `find_if` aware of this situation?

[4 marks]

Question 3 Classes

(a) RAII

(i) Explain the “Resource Acquisition Is Initialisation” idiom, and discuss its importance.

[4 marks]

(ii) Describe in general terms how RAII can be used in the following applications:

1. performing some finalising action when leaving a function;
2. connecting to some external resource.

[3 marks]

(b) Copying objects

(i) Explain what is meant by “deep copying” and “shallow copying”.

[2 marks]

(ii) Give an example where we create a “dangling pointer” by shallow copying.

[3 marks]

(iii) Explain the idea of “Copy on Write” (COW).

[2 marks]

(iv) Explain why the assignment operation is especially critical with respect to “exception safety”.

[3 marks]

(c) Member resolution

(i) Explain how the static and how the dynamic type of an object is determined, and under what circumstances the static or the dynamic type of an object is used.

[4 marks]

(ii) Discuss the meaning and usage of abstract classes.

[4 marks]