Part 2: Analysis and Design

10

Process Overview

10.1 We have learned this lesson more times than we would care to admit. Carpenters have a similar maxim: “Measure twice, cut once.” This exercise is intended to get the student to think about the value of software engineering in general. There is no single correct answer. It is probably too early in the book for the student to answer in detail about how software engineering will help. Look for indications that the student appreciates the pitfalls of bypassing careful design.

The effort needed to detect and correct errors in the implementation phase of a software system is an order of magnitude greater than that required to prevent errors through careful design in the first place. Many programmers like to design as they code, probably because it gives them a sense of immediate progress. This leads to conceptual errors which are difficult to distinguish from simple coding mistakes. For example, it is easy to make conceptual errors in algorithms that are designed as they are coded. During testing, the algorithm may produce values that are difficult to understand. Analysis of the symptoms often produces misleading conclusions. It is difficult for the programmer to recognize a conceptual error, because the focus is at a low level. The programmer “cannot see the forest for the trees”.

10.2 The intent of this exercise is get the student to think creatively about object-oriented techniques. There is no single correct answer.

In language design, for example, a variation of class diagrams could be used to describe production rules. (See Exercise 4.15.) Production rules define how non-terminals are constructed from terminals and other non-terminals. There are two basic operators in the rules: *and* and *or.* There is a loose analogy between the and/or structure of production rules and the and/or nature of aggregation and generalization relationships. As an added benefit, the notion of multiplicity could be used to express collections and options in a more elegant fashion than the paradigms used to express them via production rules.