

## 3 Complex Data

### 3.1 Problem

You want to write a program that will evaluate arithmetic, logical, and relational expressions, such as  $(3 + 5)$ ,  $(\text{true and not false})$ ,  $(3 < (7 * 2))$ .

Here is the description of some data you may use:

- An expression is either a value expression or an operation.
- A value expression represents either a number (integer) value or a boolean value (true or false).
- There are three kinds of operations: arithmetic, logical, and relational each operation is given by three pieces of data: the operator and two operands, each of which is again an expression.

Design the Java class hierarchy to represent these expressions. Again, you may find it helpful to design the data definitions in Scheme before you think of Java classes. Do not represent operators as a String.■

### 3.2 Problem

You want to represent your own Java program as data. The class hierarchy in figure 1 allows you to represent as data programs that contain classes and interfaces (without any methods).

Convert this class diagram into Java class definitions. Make examples of Java programs that can be represented as this kind of data (written as Java code) and translate it into an instance of the class Program.

Define the program in the Problem 2.3 (MapQwest) as data in this class hierarchy.■

### 3.3 Problem (10.1)

Exercise 2.3 in the book referred to the following class definition:



Design the following methods for this class:

1. *isPortrait*, which determines whether the image is taller than wider;
2. *size*, which computes how many pixels the image contains;
3. *isLarger*, which determines whether one image contains more pixels than some other image.

Before you design the method, draw a complete class diagram. ■

---

```
// clock time
class ClockTime {
  int hour;
  int minute;
  int second;

  ClockTime(int hour, int minute, int second) {
    this.hour = hour;
    this.minute = minute;
    this.second = second;
  }
}
```

Figure 2: Representing a time

---

### 3.4 Problem

A natural question concerning *ClockTimes* (see figure 2) is when one occurs earlier than another one. Develop a method that determines whether one time is earlier than another time.

Hint: The first possibility is that hour of the first time is smaller than the hour of the second. Next, consider what is possible if the hours are the same.

Next, develop a method that adds to the starting time the duration of the event (given again as an instance of *ClockTime*) and produces the ending time of the event. ■

### 3.5 Problem

In the problem 1.6 you modified the class that represents a book so it would contain a field that represents an instance of an author. Modify this further, to include authors date of birth (day, month, year, not just the year) and then design the methods to solve the following problems:

1. determine whether our book written by an author born earlier than the author of another book?
2. compare our book with another given book and produce the book written by the author whose name is earlier in lexicographical ordering.
3. Was the author of our book born in the given month? ■

### 3.6 Writing Problem

*Writing assignments are separate from the rest of the assignment for the week. You should work on this assignment alone, and submit your work individually on the due date for the rest of the homework. The answer should be about two paragraphs long — not to exceed half a page or 300 words.*

Think about the ways in which the medical profession has changed through the use of computers. Select one area where the changes improved the delivery of health care, read about it and write a brief summary of what you learned. Include references to at least two different resources you consulted. ■