Università di Pisa

Advanced Programming Preliminaries

Giuseppe Attardi

Dipartimento di Informatica

Università di Pisa

Program

0. Basic Programming and Algorithmic Design

1. **Recursive programming**
2. **Modeling with classes and hierarchies**
3. **Methods and Polymorphism**

Self Test Assessment

- **Mandatory Programming Exercises**

  Students must be capable of completing at least 10 of the tests in the section

  http://uva.onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=116

  which have at least 80% Total Users / Solving rate.

Instructors

Giuseppe Attardi
Office: 292
Mail: attardi@di.unipi.it

Binary Numeric Trees

Given binary trees with a numeric value on each node.

1. **define a (set of) classes (just structure, all public fields, no methods) to represent such trees**
2. **replace each value with the sum of the values in a subtree**
3. **compute list of total values for each level**
4. **compute list of values for sum of values in vertical split**
5. **determine if there is a path whose total is a given x**

Mobile sculpture

Given a binary numeric tree where each branch has a numeric length (Dl, Dr)

A tree is balanced if Wl x Dl = Wr x Dr

Check whether a sculpture is balanced

Tree logic function

A binary tree where nodes at each level are labeled by a different logic variable, and leaves are T/F. The value of the logic variable determines whether to descend to the left/right node.

1. **Given an assignment to the set of logic variables, compute the value expressed by a tree.**
2. **Check whether two trees compute the same function.**

Expression Tree

- **Define a set of classes to represent arithmetic expressions with constants and sums**
- **Provide an eval() method that evaluates an expression**
- **Extend the classes to allow multiplication**
- **Extend the classes to include variables**
- **Extend eval() to use an assignment for variables**

Polymorphism

- **Using an inheritance hierarchy, design a program to model 3-dimensional shapes (square pyramid, sphere, rectangular prism, cube, cylinder, circular cone).**
- **Make a top level shape interface with methods for getting the area and the volume (+ methods toString and equals).**
- **Next, build classes and subclasses for the above 3-dimensional shapes.**
- **Make sure that you place common behavior in superclasses whenever possible.**
- **Also, use abstract classes as appropriate.**
- **Add methods to subclasses to represent unique behavior particular to each 3-dimensional shape.**
- **Write the definitions of these classes and a client program showing them in use.**

Visitor Pattern

- **Design a set of classes providing polymorphic methods for implementing a visitor pattern on a table, made of rows and cells. Hint: define interface Visitable, with method accept(Table table).**
- **Use the pattern to implement a visitor that sets the property of color gray to all the cells of a table.**

Google Interview Tests

- **Given a binary tree, test whether it is a binary search tree.**
- **Find the fifth largest element in a binary search tree, efficiently and without using extra space.**
- **Check whether two binary trees are the same.**
- **Given two linked lists, return their intersection.**
- **Given an array a[i] of N integers, construct an array b[i] such that b[i] = a[0]*…a[N-1]/a[i], using constant space, O(N) complexity and no divisions.**
- **Given an array, find the longest increasing subsequence.**