Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen
Sandeep Sadanandan

Sommer Semester 2009
Problem Sheet 1
May 16, 2009

Python For Fine Programmers

Deadline: May 21, 2009

Problem 1 (6 Points)

Write a set of classes in Python for chess coins. Arrange them so that a base class consists of all the basic details (eg. Color of coin and the Color of square where it is). And the subclasses could inherit from it - all the basic properties and then the subclasses implement the special methods for them. (Queen, Bishop, Knight, Rook, King and Pawn)

Problem 2 (8 Points)

Write a class for Rational numbers. Implement any 3 of the methods which would permit one to use the mathematical operators ($+, *, /, \ldots$ or similar ones)

Problem 3 (8 Points)

Flattening a list.
Write a program, which accepts a list (with sublists, and subsublists) as an input and outputs a single list which has the members of the sublists/subsublists as elements.
And input of [1, [2, 3], [[4, [5], 6], 7], 8] would give an output of [1, 2, 3, 4, 5, 6, 7, 8].

Problem 4 (4 Points)

Given is a list of integers (any type would be fine). Write a python function which accepts the list and creates a list of tuples of the elements of the list and their respective indices.
Eg.: A list [8, 3, 5] will give [(8, 0), (3, 1), (5, 2)].

Problem 5 (4 Points)

Write a class for a node of a binary tree. (Not necessarily a binary search tree). Implement the basic methods.

Problem 6 (5 Points)

Bonus Question.

- Prove that a knight starting to move from a square of a chess board would reach a square of the same color, if and only if it has made even number of moves

- Devise a method to enable a knight, to go through every square of a chess board exactly once in 64 moves.