Homework 5

Please submit your solution via email to the instructor with CC to ly603@nyu.edu.

The deadline for Homework 5 is March 11.

Problem 1 Red-Black Tree Sets (20 Bonus Points)

This is an optional challenge exercise. Partial solutions are also accepted. The goal of this exercise is to extend the binary search tree model of Homework 4 to a more sophisticated implementation of set containers that is based on self-balancing red-black trees. You can find a comprehensive description of red-black trees and their operations, e.g., in the Wikipedia article `http://en.wikipedia.org/wiki/Red-black_tree`.

(a) Start from the signature `Tree` of the previous exercise and extend `Tree` with two additional fields `Black` and `Red`, each holding a set of `Data` elements. The purpose of these fields is to model the color of nodes in the tree. Write a predicate `isRedBlackTree` that takes a tree `t` as argument and holds true if and only if `t` is a red-black tree. Recall that a red-black tree is a sorted binary tree that satisfies the following additional properties:

(i) every node in the tree is either black or red;

(ii) the root of the tree is black;

(iii) every red node in the tree has only black successors;

(iv) for every node n in the tree, every path from n to one of the nodes in the subtrees of n that have less than two successors, visits the same number of black nodes.

Simulate some interesting red-black trees. **(5 Bonus Points)**

(b) Model the insert operation on red-black trees, including the rebalancing operation. Write an assertion that expresses that your insert operation preserves the red-black tree property. Check your assertion. *Hint: the rebalancing operation is recursive and difficult to model using transitive closure. One solution is to model all operations as events with one event for each recursive call to the rebalancing operation.* **(10 Bonus Points)**

(c) Write a module with an abstraction function relating red-black trees and `Set` containers, and an assertion expressing conformance of your insert operation on red-black trees with the insert operation on `Sets`. Check that your assertion holds. **(5 Bonus Points)**