COSC 2P03 - Assignment #2

Due Date: Nov 10, 2008, 3pm
Late Date: Nov 13, 2008, 3pm

Please read the following questions carefully. **Print all your source code (including comments) as well as a few samples of execution and include them in your hardcopy assignment submission.**

The input will be provided in a file called "input.txt". This file may contain multiple lines (representing multiple test cases) and each line will contain a comma-separated list of integers. All your output has to be written into another file called "output.txt". Don't forget to write appropriate happiness messages.

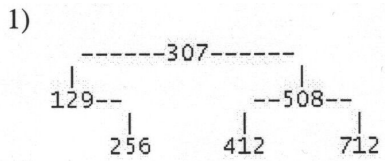
1) **Binary Search Tree Generation**. You are to generate a binary search tree, where each node will contain a unique 3-digit integer (100 – 999). The insertion into the tree has to occur in the order the integers appear in the file. You have to handle all incorrect input (even the case where the input file doesn't exist) and whenever such a situation occurs, the insertion has to be skipped and an appropriate message should be written to the output file.

2) **Visualize Tree**. Write a method that will print a tree using the following rules:

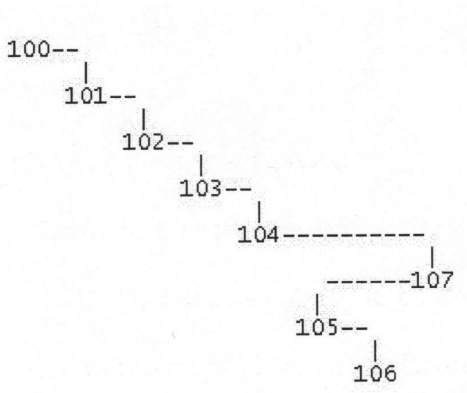
   - The root will be at the top.
   - There is an extra line between levels, which will be used to draw vertical lines above the integers.
   - A parent node "connects" to its children by drawing horizontal lines that connect to the vertical lines of those nodes.
   - For each node *p*, all the nodes that are in the left subtree of *p* will be written to the left of that node, and all the nodes that are in the right subtree of *p* will be written to the right of it.

Examples:

1)



2)

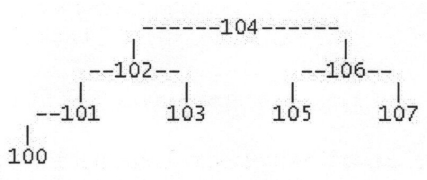


3) **Full Tree**. Write a method to determine whether a tree is full. This method should return true if the tree is full and false otherwise. Use the tree from part 1 as your input.

4) **Complete Tree**. Write a method to determine whether a tree is complete. This method should return true if the tree is complete and false otherwise. Use the tree from part 1 as your input.

5) **Create Complete Tree**. This method should take a non-complete binary search tree and insert the same nodes into a new binary search tree in such an order, that the new tree will be complete. Use the tree from part 1 as your input.

   Using the second example from part 2, this method will return the following new binary search tree:



6) **Reverse Tree Recursively**. Write a recursive method that will reverse a tree. Use the tree from part 1 as your input.

7) **Reverse Tree Iteratively**. Write an iterative method that will reverse a tree. Use the tree from part 6 as your input. The tree resulting form part 7 should be identical to the tree from part 1.

Assignment Submission Guidelines

- **This assignment must be submitted both in hardcopy and softcopy (electronically).**
- A submitted assignment with either hardcopy or softcopy version missing is deemed incomplete and will be penalized.
- **To make an electronic submission, execute the program 'submit2p03' on sandcastle.cosc.brocku.ca from the directory that contains all the files that you want to submit, and only these files.** This program will copy the contents of the current directory to the marker's account.
- **The hardcopy submission must include all source code (including comments) as well as a few examples of execution**. These should be printed on a laser or inkjet printer.
- All the papers must be stapled together at the upper left-hand corner of the page and should be placed in a 9" x 12" sealed envelope.
- A standard assignment cover-page (http://www.cosc.brocku.ca/forms/cover) should be printed, signed and stapled to the front of the outside of the envelope.
- The submission should be placed in the Assignment Box outside of J332, in the slot labeled COSC 2P03, before the due time indicated above. Only one submission (i.e. to the box) should be made per assignment.
- Assignments not including a coversheet will NOT be marked.
- Assignments that do not follow these guidelines will be penalized.
- Familiarize yourself with the department's policy on plagiarism and the university regulations on plagiarism and academic misconduct.