CISC 235 Assignment #2

Instructions

This assignment is due at 3:00 p.m. on Tuesday, February 15th. Please email your submission to the Teaching Assistant assigned to your lab section (Lab A (Monday): 8jjl1@queensu.ca, Lab B (Friday): zi@acm.org).

Do not compress the files before attaching them to your email (please include each of the files as a separate attachment to the email).

The labs provide an opportunity for you to work on assignments with the supervision of our Teaching Assistants.

Part 1 (4 Marks): BST Insertion and Search

```
public class Tree
{
	private class Node
	{
		public Node left;
		public Node right;
		public int data;

		Node( int data )
		{
			this.data = data;
			this.left   = null;
			this.right = null;
		}
	}

	private Node head;

	public Tree()
	{
		head = null;
	}
}
```

Write insert and search methods for the Tree class above. These methods must maintain the properties of a binary search tree (e.g. smaller values in the left subtrees and larger values in the right subtrees). You can work from the methods found in the class slides. You may not use or modify source code that is not found in the slides.

Note: in order to debug your code, it might be helpful to write a method to display the contents of the tree. This is not required for the assignment.

Part 2 (12 Marks): Finding Values in a Binary Search Tree

In the main method for your program, insert 1,000 nodes into the binary search tree, setting each node to contain a randomly generated Integer with a value between 1 and 1,000. Display the height of the tree and the heights of the root node's left and right subtrees.

Next, search for 10,000 randomly generated values between 1 and 1,000. Display the minimum, maximum, and average number of comparisons performed during the searches.

Note: you will need to write code to calculate the heights of the subtrees and count the number of comparisons.

Save this program (Part 1 and Part 2) to a file named "BinarySearchTree.java".

Part 3 (4 Marks): Thinking about Search

Write answers to the following questions. Each answer should not be longer than four sentences.

1. Describe the relationship between the height of the tree and the minimum number of comparisons required to perform a search.

2. Describe the relationship between the height of the tree and the maximum number of comparisons required to perform a search. How does this relationship differ from (1)?

3. Describe the relationship between the height of the tree and the average number of comparisons required to perform a search. How does this relationship differ from (1) and (2)?

4. On average, will the difference between the heights of the root node's subtrees change if the 1,000 inserted Integers have values between 1 and 10,000 instead of 1 and 1,000? Explain why or why not.

Save this paragraph to a file named "SearchThoughts.doc".