EXAM 1: 3/9, closed book.

Chapters 1-4, except B-trees, TreeSet and Map.

7 multiple choice, 4 short answers, and 2 programs.

Ch1: Review of java:

Function objects

Ch2: complexity

Given a code segment, find the complexity.

No proof type questions.

Ch3: lists, stacks, queue.

Stack application:

postfix and infix

convert infix to postfix using two technique

evaluating postfix expressions using stack

balanced expression (use stack to check if delimiters match)

Queues:

What is a circular array?

How to determine a queue is full or empty.

Implementation using an unused array location.

Lists:

Some programs in lab 3such as print, union, intersection, etc.

Ch4: Trees

Terminologies:

Depth: For any n, length of the unique path from root to ni where root has depth 0.

Height: longest path from ni to a leaf where leaves have height 0.

Traversal

Expression trees:

Given an infix expression, show expression tree.

Know how to write methods for BST such as insert and remove.

AVL Trees:

Know the 4 cases and rotations needed for each one.

Huffman Tree

Examples:

Given the following infix expression, what is the expression tree?

A – (B \* C / F + (E – M)) + M

1st step: convert to postfix.

(E – M) 🡪 (EM–)

(B \* C / F + (EM–)) 🡪 (B C \* F / (E M –) +)

A – (B C \* F / (E M –) +) 🡪 A (B C \* F / (E M –) +) –

(A (B C \* F / (E M –) +) –) + M 🡪 (A (B C \* F / (E M –) +) –) M +

(A (B C \* F / (E M –) +) –) M + 🡪 A B C \* F / E M – + – M +