CISP 430
Data Structures
Assignment 4

Binary Trees and Dictionaries

Tree is a set of relations derived from set $V\{v^1, v_2, .., v^n\}$ and $E(e_1, e^2, .., e^n\}$ represented by set $T\ \{t^1, t^2, .., t^n\}$ where v^i is a vertex of v, e^i is an edge of E. Each t^i, which is a mapping between a vertex of V and edges of e, is also a tree, with tuples form $\{V^i, E^i, E^j\}$

Binary Trees are well suited for representing a dictionary in an "internal" form. Any software which organizes data using a tree form must resolve the issues of tree insert, delete, and print.

An AVL tree is a height-balanced binary tree. Write a program which is capable of building a word dictionary using a balanced binary tree as the internal form subject to the following requirements:

1) Implement the AVL tree as an object ADT.

2) The words of the dictionary must be accessed from a file. These words, randomly ordered, are as follows:

Automaton	Mhz	Parallel Architecture
Psuedo Coding	Substrate	Dope Vector
GIGO	State Machine	DFD
Key Field	Neural Net	OS/2
Pass 1	T2 Link	Uplink
Vectored Processor	Flip Flop	Control Layer
FDDI	OSI	MilStd 2167A
OOD	Abstraction	Global Variable
Isolation	Normalization	Distributed Processing

3) Print the dictionary in ascending and descending order in the form of a report - be certain to include the balance value of each node in the tree.

4) Print the tree again in ascending and descending order after deletion of the following terms from the tree:

   Isolation
   Normalization
   Distributed Processing

5) Draw the nodal hierarchy of the tree using the report produced in #4 above.