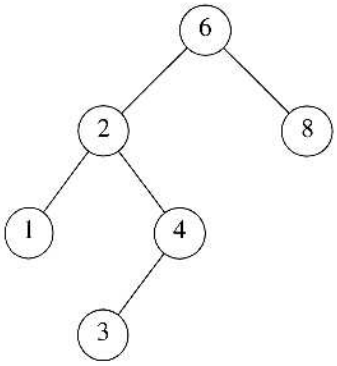
CISC 810: Fundamentals of Computational Science, Assignment 4

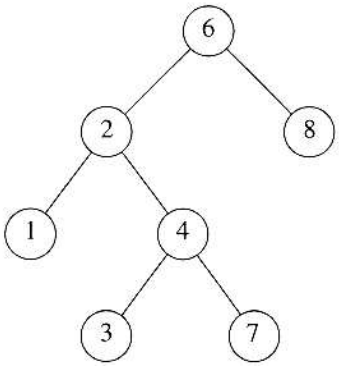
Set September Nov 6th due Nov 20th

For all programming questions ensure code is commented well and appropriate function/subroutine calls are made.

Q1. Linked lists. Download the file particles.dat.gz from the class website. Write a code that reads in this file, and then creates a singly linked list that is ordered so that the first particle in the linked list has the largest z coordinate, while the last particle in the linked list has the lowest z coordinate. Send me your code and provide a hardcopy.

Q2. Binary trees. (a) One of the following trees is not a binary search tree - which one is it? Why?





(b) In the lecture we gave an example of deletion in a *binary search tree* when a node only has one child, but we did not look into detail at the case of a node having two children. Workout the algorithm for deletion of a node when it has two subtrees.
(c) Develop an non-recursive inorder traversal algorithm and implement it in either C/FORTRAN. Send me the code and provide a hardcopy.

Q3. Hash tables. A very useful statistic for measuring the effectiveness of hash tables is how many probes, on average, it takes to find a key in the hash table (this is a *successful search*). In this question, assume linear probing is used so that if a key resolves to an already filled table value then the next highest table value is tested until an empty slot is found. The series of keys is 555,272,520,713,283,201,582,794,949.
(a) What is the average number of probes required for the above keys and the hash function h(k)=k mod 17 (assume the linear probing obeys wrap around behaviour). Repeat for h(k)=k mod 13 and h(k)=k mod 11.
(b) Find a hash function that has no collisions for these keys (without creating an overly large hash table).