Instructions:
Put your assignment in an envelope labeled with your name and course number and drop it in the CS2210 locker # 310 by the midnight on the due date.

1. (20 marks) Consider the algorithm $Multiply(A,n)$:

   ```plaintext
   Algorithm Multiply(A,n)
   Input: Array A storing integers. Its size is at least n.
   if n < 4 return 0
   else
     temp ← A[n − 1] * Multiply(A, n − 3)
   return temp
   ```

   (a) (5 marks) Write down recurrence equations that describe the running time of algorithm $Multiply(A,n)$.

   (b) (15 marks) Solve these recurrence equations and give the best asymptotic (big-O) characterization of the time complexity of the algorithm.

2. (15 marks) Prove that in a proper binary tree of height $h$, the minimum number of nodes is $1 + 2^h$, and the maximum number of nodes is $2^{h+1} − 1$, where $h$ is the height of the tree. That is prove $1 + 2^h \leq n \leq 2^{h+1} − 1$, where $n$ is the number of nodes.

3. (10 marks) Draw a binary search tree where each node stores a character key, and preorder traversal visits nodes in the order $DCBFEZ$, and postorder traversal visits nodes in the order $BCEZFD$.

4. (10 marks) Suppose that a heap stores 1000 elements. What is its height?

5. (15 marks) Let $H$ be a heap of height $h > 3$. State all the levels of the heap at which the 3rd largest key element can be located. Recall that larger keys correspond to lower priority.
6. (30 marks)

(a) (20 marks) Suppose $T$ is a tree that at each node stores a positive integer key. Let us call a node EvenOdd if the sum of keys in its left subtree is even, but the sum of the keys in its right subtree is odd. For example, in the tree above, the root node is EvenOdd, since the sum of keys in its left subtree is 4 and in the right subtree is 5. None of the other nodes are EvenOdd. Write an algorithm that takes as an input the root of the tree and returns the number of EvenOdd nodes. Assume that leaves are never EvenOdd. For the tree above, your algorithm should return 1. If you need, you can store an additional variable $v.temp$ at each node.

(b) (10 marks) Analyze the running time of your algorithm in part (a) as a function of the number of tree nodes $n$. 