Question 1

Below are two types of layouts for perfectly balanced trees on a grid, where nodes are at grid line intersections and edges are on grid lines.
Question 1 (cont’d)

Such layouts are used in embedding circuits on silicon. In this application we want a layout that minimizes the area used by the layout.

The goal in this question is to compare the above two designs by computing the area used by each design as a function of the number of leaves in the tree.
Question 1 (cont’d)

Generalize Layout 1 for \( n \) leaves, \( n \) a power of 2.

• Give recurrences for the height \( H(n) \) and the width \( W(n) \) of Layout 1.

• Solve the recurrences to obtain a closed-form formulas for \( H(n) \) and \( W(n) \).
Question 1 (cont’d)

• Generalize Layout 2 for $n$ leaves, $n$ a power of 4.

• Let $L(n)$ be the length of the side of the square in Layout 2.

• Give a recurrence for $L(n)$.

• Solve the recurrence to obtain a closed-form formula for $L(n)$. 
• Compute the area $A_1(n)$ of Layout 1 as a function of the number of nodes $n$.

• Compute the area $A_2(n)$ of Layout 2 as a function of the number of nodes $n$.

• Which design has smaller area?
Question 2

Consider non-negative integers represented in binary (i.e., the numbers are given as sequences of bits).

• How fast can you double an n-bit number?

• How fast can you add two n-bit numbers?
So doubling is faster than arbitrary addition.

Consider a similar question about multiplication: Is squaring faster than multiplying?

Prove that the answer is “no”: If you can square $n$-bit numbers in $S(n)$ time then you multiply $n$-bit numbers in $\Theta(S(n) + n)$ time.

Hint: Reduce multiplying to squaring (and some $\Theta(n)$ time operations).
Question 3

Minimum spanning tree problem: Given an undirected, connected graph with edge weights, find a subset of the edges that form a tree with minimum weight.

Proposed D&C MST algorithm:

• If the graph has one node, return the empty tree
• Otherwise,
  
  – partition the set of nodes of the graph into any two sets $V_1$ and $V_2$ of about the same size
  
  – recursively find MSTs of the graphs induced by $V_1$ and $V_2$, and join them by the minimum weight edge connecting a node in $V_1$ to a node in $V_2$

Prove or disprove: This is a correct MST algorithm.
Question 4

Demonstrate the polynomial multiplication algorithm based on the FFT by using it to multiply the polynomials $x^2+1$ and $x+1$.

- Use the FFT to evaluate each polynomial at an appropriate set of points, and show the resulting vectors;
- multiply the vectors component-wise; and
- use the inverse FFT to obtain the coefficients of the final result.
Question 5

Recall that $z$ is a **primitive** $n$-th root of 1 if

(a) $z^n = 1$; and

(b) $z^k \neq 1$ for all $k, 0 < k < n$.

Thus, if $z$ is a primitive $n$-th root of 1, then $z^0, z^1, z^2, \ldots, z^{n-1}$ is the list of **all** the $n$-th roots of 1.

Prove the following facts (used in our discussion of the FFT):

1. If $z$ is a primitive $n$-th root of 1, then $z^{-1}$ is also a primitive $n$-th root of 1.

2. If $z$ is a primitive $n$-th root of 1 and $n$ is even, then $z^2$ is a primitive $n/2$-th root of 1.