1. (5%) Name and show two data structures that can be used to implement a tree.

2. (10%) Given the following expression tree

```
    statement
     /  
    var  assignment
     /    
    X    expression

       |      /  
      |     expression  
      |    /    
      |   expression  
      |  /      
      | expression

Y + Z
```

The internal nodes are used to represent the type of operation such that one can understand the semantic meanings of the operation when the node will be visited in the future. Write down the results if the expression tree is traversal in
i) Pre-order, ii) In-order, and iii) Post-order.

3. (10%) A file-spanning tree of a connected undirected graph G is a sub-graph (or tree) of G that contains all files needed for a program's execution from a vertex Vi. A minimum file-spanning tree is a file-spanning tree that has no subset of file-spanning trees that can identify from it. Given the following graph with files distribution,

```
V1
\( f_1, f_2 \)

V2
\( f_3 \)

V3
\( f_1, f_4 \)

V4
\( f_2, f_3 \)
```

Suppose a program requires the access of files \( f_1, f_2, \) and \( f_3 \) in order to complete its execution starting from vertex V1, identify all file-spanning trees and mark which of them are minimal file-spanning trees in this case.

4. (6%) Given the definition of \( \Theta \) [Theta] is as follows:
\( f(n) = \Theta(g(n)) \) if and only if there exist positive constants \( c_1, c_2, \) and \( n_0 \) such that \( c_1g(n) \leq f(n) \leq c_2g(n) \) for all \( n, n \geq n_0 \). Show that the following equalities are incorrect.
(a) \( n^2/\log n = \Theta(n^2) \)
(b) \( n^2 + \epsilon + n^3\log n = \Theta(n^3 + \epsilon) \)
5. (6%) What is an abstract data type? Please give an example (i.e. a stack or a queue) to explain your answer.

6. (7%) Write a recursive function to compute \( \text{powerset}(W) \). Where \( W \) is a set of \( n \) elements, and the \( \text{powerset} \) of \( W \) is defined as the set of all possible subsets of \( W \). For example, if \( W = \{x, y, z\} \) then \( \text{powerset} (W) = \{\{\}, \{x\}, \{y\}, \{z\}, \{x, y\}, \{x, z\}, \{y, z\}, \{x, y, z\}\} \).

7. (6%) Given \( 5 \times 6 \) matrix \( A \) as follows

\[
\begin{pmatrix}
0 & 0 & 2 & 0 & 0 & 3 \\
4 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 5 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 6 & 0
\end{pmatrix}
\]

Please use a link list to represent matrix \( A \).

8. (9%) Give definition to the following terms: NP, polynomially reducible, NP-hard and NP-complete? If you know a set of NP-complete problems, why and how you can use it to prove another NP-complete problem?

9. (8%) Let \( G = (V, E) \) be an undirected graph such that \( |V| = 2n \) and the degree of each vertex is at least \( n \). Explain why a perfect matching exist and then write an algorithm to find a perfect matching in this graph.

10. (8%) Explain how and why the Fourier transform can be used to compute the product of two polynomial functions in time complexity \( O(n \log n) \)? Note that you don’t need to write the detailed algorithm.

11. (12%) Let \( T(n) = \Theta(f(n)) \). Derive \( f(n) \) in the simplest formula for each of the following \( T(n) \).
   a. \( T(n) = 7 \ T(n/2) + n^2 \), \( T(c) = c \), if \( c<2 \).
   b. \( T(n) = 4 \ T(n/4) + n / \log n \), \( T(c) = c \), if \( c<4 \).
   c. \( T(n)= \sum_{i=1}^{2 \log_2 n} ((\log_2 n) - i) * 2^i \)
   d. \( T(n)= \sum_{i=1}^{n} \frac{1}{i} \)
12. (5%) Given a finite set A and a mapping function f from A to itself, describe an algorithm to find a subset S of A with maximum size such that f is one-to-one when restricted to S.

13. (8%) Now, you are inside a Buffet restaurant. Assume that your stomach can only accept food with maximum size M and there are n kinds of food with sizes and values as \((s_1, v_1), (s_2, v_2), \ldots, (s_n, v_n)\), in the restaurant. Design an algorithm to find the highest value of food that you can eat. Only need to demonstrate your algorithm based on the case: \(M = 22\) and 4 kinds of food with \((3, 4), (4, 5), (6, 9), (8, 13)\). Hint: this is equivalent to the knapsack problem.