Questions for data structures and algorithms

Question

8. **Define "space complexity" in the context of algorithms.** (Answer should explain that it describes how much memory an algorithm uses relative to the input size.)
2. **Describe what a stack data structure is and give a real-world example of its use.** (Answer should mention LIFO principle and an example like function call stack or undo/redo functionality.)
1. **Explain the difference between an array and a linked list.** (Answer should highlight fixed vs. dynamic size, memory allocation, and access time complexities.)
4. **Explain the concept of a binary tree.** (Answer should mention nodes, branches, root, and the property of having at most two children per node.)
7. **Define "time complexity" in the context of algorithms.** (Answer should explain that it describes how the runtime of an algorithm scales with input size.)
6. **What is the purpose of a hash table (or hash map)?** (Answer should describe its use for fast key-value lookups.)
3. **What is a queue and how does it differ from a stack?** (Answer should mention FIFO principle and contrast it with LIFO of the stack.)
9. **What is the difference between a recursive and an iterative algorithm?** (Answer should mention the use of function calls within itself for recursion vs. loops for iteration.)
10. **Briefly explain the concept of Big O notation.** (Answer should mention that it's a way to describe the upper bound of an algorithm's time or space complexity, focusing on dominant terms as input size grows large.)
5. **What is a graph data structure? Give a simple example.** (Answer should mention nodes and edges, and a simple example like a social network or a map.)