

11 Lecture - CS301

Important Subjective

1. **What is a Priority Queue?**

Answer: A Priority Queue is an abstract data type that stores a collection of elements with priority values assigned to each element. Elements with higher priority values are given precedence over those with lower priority values.

2. **What is the difference between a Queue and a Priority Queue?**

Answer: A Queue is a data structure that follows the First-In-First-Out (FIFO) principle, while a Priority Queue follows the priority-based ordering principle. In a Queue, elements are added to the back and removed from the front, while in a Priority Queue, elements are removed based on their priority values.

3. **What are the commonly used data structures to implement a Priority Queue?**

Answer: The commonly used data structures to implement a Priority Queue are binary heap, Fibonacci heap, and sorted array.

4. **What is a binary heap?**

Answer: A binary heap is a complete binary tree where the parent node has a higher priority value than its children nodes.

5. **What are the operations that can be performed on a Priority Queue?**

Answer: The operations that can be performed on a Priority Queue are inserting an element, deleting the element with the highest priority, and changing the priority of an element.

6. **What is the time complexity of inserting an element into a Priority Queue?**

Answer: The time complexity of inserting an element into a Priority Queue depends on the implementation. For a binary heap, the time complexity is $O(\log n)$, while for a Fibonacci heap, it is $O(1)$.

7. **What is the time complexity of deleting the element with the highest priority from a Priority Queue?**

Answer: The time complexity of deleting the element with the highest priority from a Priority Queue depends on the implementation. For a binary heap, the time complexity is $O(\log n)$, while for a Fibonacci heap, it is $O(\log n)$ amortized.

8. **What is the difference between a Max Heap and a Min Heap?**

Answer: In a Max Heap, the parent node has a higher priority value than its children nodes, while in a Min Heap, the parent node has a lower priority value than its children nodes.

9. **What is the application of Priority Queues?**

Answer: Priority Queues are used in various applications such as task scheduling, Dijkstra's shortest path algorithm, Huffman coding, A* search, and many more.

10. **How can a Priority Queue be implemented in C++?**

Answer: A Priority Queue can be implemented in C++ using the `std::priority_queue` class from the Standard Template Library (STL).