

Analysis Questions and Answers for Task 2

* Why is a Queue suitable for managing treatment requests?

-) A queue operates according to the first in, first out (FIFO) principle, which is the most correct and effective way to handle a sequence of requests. In a medical or service setting, this guarantees equity. Treatment will begin with the patient who arrives first.

The LinkedList structure used by the TreatmentQueue implementation allows for:

- The front indicates which patient will be treated next.
- The rear indicates the most recent arrival.

By keeping a chronological order that is predictable for both staff and patients, this structure avoids line cutting and ensures that no request is left waiting forever while newer ones are handled.

* What issues would occur if a Stack were used instead? Compare their time complexities.

-) If a stack were used, last in first out (LIFO) logic would be applied. Serious issues like injustice would result from this. The patient at the bottom of the line would be the one who arrived first. As long as new patients kept coming in, the first ones would never be seen. A treatment schedule that gives instant service to the newest patient is invalid.

-) Both data structures are highly efficient for the primary purposes when implemented correctly. However, their usefulness is determined by their behaviour. Even though the BigO time complexities are the same $O(1)$ for adding and removing, the Stack logically fails. Enqueue and dequeue are $O(1)$ because the methods maintain references to both front and rear. In a stack, you would only retain the top reference. In a real clinic, a Stack would cause a FIFO disaster even though it is computationally quick.

Output of The Test Scenario

Queue size: 0

Queue size: 8

Queue size: 5

Treatment Queue Requests (Front -> Rear): * Patient's ID: 104, Arrival Time: 1800, Priority: true * -> * Patient's ID: 105, Arrival Time: 2030, Priority: false * -> * Patient's ID: 106, Arrival Time: 1030, Priority: false * -> * Patient's ID: 107, Arrival Time: 830, Priority: false * -> * Patient's ID: 108, Arrival Time: 900, Priority: false *