

## Analysis Questions and Answers for Task 3

\* Why is a Stack suitable for managing discharge records?

- ) A stack is suitable for many administrative and medical applications because it follows the last in first out (LIFO) principle. This is especially helpful when it is necessary to access or examine the most recent executed record first. For instance, if an administrator must undo a discharge or review the most recent patient activity to ensure that paperwork was filed correctly, the Stack provides instant O(1) access to the most recent entry. With the most recent event at the top it creates history.

\* What would happen if you replaced it with a Queue? Discuss BigO comparisons.

- ) Both data structures offer high performance for their primary functions when implemented with a LinkedList. The functional outcome varies for both, but the time complexity remains O(1). Records are displayed chronologically by printStack(). Records in a queue would be shown in the same order, from oldest to newest. Converting from a stack to a queue wouldn't have a negative effect on performance, but it would significantly change the priority of the data. If the goal is to always handle the most recent discharge first. The Stack remains the best choice.

## Output of The Test Scenario

There are no records.

Records (Top > Bottom): \* Patient's ID: 105, Discharge Time: 1400 \* ->

\* Patient's ID: 104, Discharge Time: 1300 \* ->

\* Patient's ID: 103, Discharge Time: 1200 \* ->

\* Patient's ID: 102, Discharge Time: 1100 \* ->

\* Patient's ID: 101, Discharge Time: 1000 \*

Records (Top > Bottom): \* Patient's ID: 103, Discharge Time: 1200 \* ->

\* Patient's ID: 102, Discharge Time: 1100 \* ->

\* Patient's ID: 101, Discharge Time: 1000 \*