

{Mastermind of Design}

Description

The "Mastermind of Design" challenge, you'll harness object-oriented design principles to create a versatile data structure using c++. This data structure is a combination of space-efficient linked lists with stacks as nodes and sub-linked lists acting as queues. Your goal is to design a class hierarchy and implement a real-world application that leverages this unique data structure.

Deadline: {28-NOV-2023} {Pair Task (2-persons only)}

Tasks

- Create a linked list where each node is implemented as a stack, capable of storing elements of any data type.
 - Develop a sub-linked list structure to function as a queue for efficient element management.
- Construct a class hierarchy that encompasses classes for stack nodes, queues, and the main linked list.
 - Utilize inheritance and polymorphism to ensure a clean and extensible design.
- Implement methods for adding elements to the front of the linked list (push operation) and for removing elements from the front (pop operation).
 - Ensure that these operations are efficient and maintain the data structure's integrity.

Bonus

- Implement dynamic resizing of the linked list to adapt to changing numbers of elements.
- Develop error handling mechanisms for scenarios such as stack underflow or queue overflow.

Real-World Application {Smart Factory Manufacturing}

Consider a smart factory project where products move through various manufacturing steps on the factory floor. Your space-efficient linked list can represent the sequence of steps each product undergoes. Each node in the linked list can contain information about a particular manufacturing operation, while the sub-linked list (queue) manages the order in which products enter these operations.

Application Features:

1. **Product Sequence Management:** Utilize the space-efficient linked list to manage the sequence of steps each product undergoes during manufacturing.
2. **Operation Information Storage:** Each node in the linked list stores information about a specific manufacturing operation for a product.
3. **Order Management:** The sub-linked list manages the order in which products enter various manufacturing operations.
4. **Production Optimization:** The data structure optimizes the tracking and management of the manufacturing process, leading to efficient production in an industrial setting.

Bonus:

1. Error Handling and Exception Handling:
2. Implement error handling mechanisms to gracefully manage unexpected scenarios during the manufacturing process.