**PART I – STACK**

A. Basics

**Q1: How does MTN MOMO app back navigation show LIFO nature?**

* Because the last step you entered (like payment details) is the first one removed when you press “back,” just like popping from a stack.

**Q2: Why is UR Canvas back action similar to popping?**

* Each step in navigation is stored on the stack; pressing “back” removes the most recent step (top element), just like a stack pop.

B. Application

**Q3: How could a stack enable undo function?**

* Each action is pushed onto the stack. When undo is pressed, the top action is popped (reversed), restoring the previous state.

**Q4: How can stacks ensure forms are correctly balanced?**

* Push each opening bracket/field onto the stack. When a closing bracket/field appears, check and pop the matching opening one. If the stack is empty at the end, the form is balanced.

C. Logical

**Q5. Which task is next (top of stack)**

Sequence: Push (“CBE notes”), Push(“Math revision”), Push(“Debate”), Pop(), Push(“Group assignment”).

* Tasks in stack (from bottom to top): CBE notes → Math revision → Group assignment. Answer: Next task is “Group assignment”.

**Q6. Answers remains in the stack after undoing**

Student undoes 3 recent actions.

* Popping removes the last three actions, leaving only earlier ones in the stack.

Answer: The first actions remain; the last three are gone.

D. Advanced Thinking

**Q7: How does stack enable backtracking in Rwanda Air booking?**

* Each form step is pushed onto the stack; popping allows retracing step-by-step back through the booking process.

**Q8. Show how stack algorithm reserves the proverb**

Reverse “Umwana ni umutware” with stack:

* Push: “Umwana” → “ni” → “umutware”

Pop: “umutware” → “ni” → “Umwana”

Answer: Reversed: “umutware ni Umwana”.

**Q9: Why stack suits DFS better than queue?**

* DFS requires going deep into one branch before backtracking. A stack naturally supports this by remembering the path and popping back when dead ends are reached.

**Q10: Suggest feature using stacks in BK Mobile transaction history.**

* A “back/forward navigation” feature: recent transactions are pushed onto a stack; popping lets users return step-by-step through their transaction views.

**Part II – QUEUE**

A. Basics

**Q1: How does restaurant service show FIFO?**

➡ The first customer to arrive is the first to be served, just like dequeueing from the front of a queue.

**Q2: Why is YouTube playlist like dequeue?**

➡ The next video (front of queue) plays automatically, then is removed, just like dequeueing.

B. Application

**Q3: How is RRA tax line a real-life queue?**

➡ People join at the end (enqueue) and are served from the front (dequeue), preserving fa Q irness.

**4: How do queues improve customer service at MTN/Airtel?**

➡ By serving people in arrival order, queues reduce conflict, ensure fairness, and keep service organized.

C. Logical

**Q5: Sequence: Enqueue(“Alice”), Enqueue(“Eric”), Enqueue(“Chantal”), Dequeue(), Enqueue(“Jean”).**

➡ Initial queue: Alice → Eric → Chantal.

Dequeue removes Alice. Queue: Eric → Chantal.

Enqueue Jean. Queue: Eric → Chantal → Jean.

Answer: Front = Eric.

**Q6: How does a queue ensure fairness in RSSB applications?**

➡ Each application is handled in the order it arrives (FIFO), so no one skips ahead.

D. Advanced Thinking

**Q7: Real-life mappings:**

Linear queue: People at a buffet; once served, they leave permanently.

Circular queue: Buses loop routes, rejoining the queue at Nyabugogo.

Deque: Boarding buses from front or rear, allowing insertions/removals at both ends.

**Q8: How can queues model Kigali restaurant orders?**

➡ Orders are enqueued when placed. The kitchen dequeues them in arrival order when ready.

**Q9: Why CHUK emergencies are a priority queue?**

➡ Critical patients bypass others, so service is based on priority, not just order of arrival.

**Q10: How would queues fairly match drivers and students in moto app?**

➡ Students are enqueued when requesting rides; drivers dequeue in order. This ensures first-come-first-served matching.