Name: SHEJA KEVIN Alberto

Reg No: 224004070

Date: 23/09/2025

BIT – Data Structures Exercise

Part I – STACK

A. BASICS

Q1: How does MoMo app show LIFO?

In the MTN MoMo app, whenever you fill payment details step by step, each new detail you enter is placed on top of the previous one, just like stacking papers on a desk. For example, you first enter the amount, then choose the recipient, then confirm with a PIN. If you press back, the app doesn't remove the very first thing you entered, but rather the last step you completed, such as the confirmation screen.
This is a perfect example of the **Last In, First Out (LIFO)** principle. The most recent step is always the first one to be removed. This makes the whole process orderly and predictable, and it is exactly how a stack behaves in data structures.

Q2: Why is pressing back in UR Canvas like a Pop?

When navigating UR Canvas, each course module you open is stored as if it were placed on top of a stack of visited pages. If you press the back button, the system does not jump randomly; it specifically removes the most recent module you visited. This behavior is just like a **Pop operation** in stacks.
The topmost module is undone first, and you continue to move backwards step by step. This makes learning platforms like Canvas easier to use, because students can retrace their steps in the exact order they were taken.

B. APPLICATION

Q3: How can a stack enable undo in banking apps?

Stacks are extremely important for implementing undo functions. Let's take the example of BK Mobile Banking. Suppose you are typing in a payment amount or filling in an account number. If you make a mistake, you usually want to undo the last action, not delete everything.
A stack makes this possible because every action you take is added on top of the stack. When you undo, the system pops only the last action. This step-by-step correction saves time and prevents frustration, especially in financial transactions where accuracy is very important.

Q4: How do stacks check balanced forms?

In programming, stacks are commonly used to check balanced parentheses. The same principle can be applied to online forms, such as those on Irembo. Each time you open a new field to fill in details, it is like pushing an opening bracket onto the stack. When you finish filling it, it is like popping a closing bracket.
If every open field has a closing match, the form is balanced and complete. If one field is left open or unmatched, the stack will show an imbalance. This ensures that users provide all the required information correctly.

C. LOGICAL

Q5: Which task is next after the given sequence?

The sequence given is:

- Push("CBE notes")
- Push("Math revision")
- Push("Debate")
- Pop()

- Push("Group assignment")

Step by step:

1. After the first three pushes, the stack is: CBE notes → Math revision → Debate.
2. Pop() removes "Debate," leaving: CBE notes → Math revision.
3. Push("Group assignment") adds it to the top, so now the stack is: CBE notes → Math revision → Group assignment.

So, The next task at the top is **Group assignment.**

Q6: Which answers remain after 3 undo operations?

If a student undoes three actions during an ICT exam, it means the last three entries are removed. Only the earlier answers that were entered before those three remain in the stack.
This shows how undo does not delete everything, but rather only the recent actions. In real life, this gives students confidence because they know they won't lose all their work by undoing a few mistakes.

D. ADVANCED THINKING

Q7: How does a stack allow retracing in RwandAir booking?

When booking a ticket in the RwandAir app, every detail you fill in—name, passport, seat selection, payment method—is added one after the other like pushes onto a stack. If you want to go back, the system pops off the most recent entry. This allows you to correct mistakes or change information step by step without starting the booking all over again.

Q8: How to reverse the proverb with a stack?

Take the proverb *"Umwana ni umutware."*

- Push each word: Push(Umwana), Push(ni), Push(umutware).
- Now the stack is: bottom → Umwana, middle → ni, top → umutware.
- Pop them one by one: umutware, ni, Umwana.

The reversed proverb is: **"umutware ni Umwana."**
This shows how stacks can reverse words, numbers, or even entire sentences by using the LIFO principle.

Q9: Why is a stack better than a queue for DFS?

Depth-First Search (DFS) means going as deep as possible into a search path before backtracking. In Kigali Public Library, a student might search shelf by shelf, going deeper into one section before returning.
A stack is perfect for DFS because it always returns to the most recent place you visited. A queue, on the other hand, works in breadth-first order, which explores level by level, not deeply. For deep searches, stacks are much better.

Q10: Suggest a stack-based feature in BK Mobile.

In BK Mobile, a stack could be used for transaction navigation. Every transaction you view could be pushed into a stack. If you want to go back, the last viewed transaction could be popped off.
A more advanced idea is to use **two stacks**: one for backward navigation and one for forward navigation. This is the same way browsers allow you to click back and forward buttons. It would make the banking app much more user-friendly.

Part II – QUEUE

A. BASICS

Q1: How does a restaurant queue show FIFO?

At a Kigali restaurant, customers are served in the order they arrive. The first person in line gets served first, and the last person must wait the longest. This is a perfect representation of FIFO: **First In, First Out.**

Q2: Why is a YouTube playlist like a dequeue?

When you play a YouTube playlist, the videos play automatically in the order they were added. The first video added is the first one played, and after it finishes, the next one follows. This is the same as the **Dequeue** operation, where the front item leaves first.

B. APPLICATION

Q3: How is RRA office line a queue?

At the RRA office, taxpayers form a line to pay their dues. Whoever arrived first gets served first. Nobody can skip to the front unless given special priority. This is the simplest real-life queue, where service is fair and organized.

Q4: How do queues improve service at MTN/Airtel?

At MTN and Airtel service centers, queues are used for SIM replacement or customer support. Without queues, people might push or argue over who should be served. With queues, customers are served one by one in order of arrival, improving efficiency and fairness.

C. LOGICAL

Q5: Who is at the front after the sequence?

The sequence is:

- Enqueue(Alice)
- Enqueue(Eric)
- Enqueue(Chantal)
- Dequeue()
- Enqueue(Jean)

Step by step:

1. Queue: Alice → Eric → Chantal.
2. Dequeue() removes Alice. Queue: Eric → Chantal.
3. Enqueue(Jean). Queue: Eric → Chantal → Jean.

So, The person at the front now is **Eric.**

Q6: How does a queue ensure fairness at RSSB?

In RSSB, pension applications are handled in the order they are submitted. A queue ensures fairness because no one who applied later can be served before someone who applied earlier. This builds trust and transparency in the process.

D. ADVANCED THINKING

Q7: Examples of queue types in Rwanda.

- **Linear queue:** Like people standing at a wedding buffet. Each person is served one by one until the line is empty.
- **Circular queue:** Like buses at Nyabugogo. After finishing a route, each bus goes back to the starting point and joins the line again.
- **Deque:** Like boarding a bus from both the front and rear doors. People can enter and leave from both ends.

Q8: How do restaurant orders follow queues?

In a Kigali restaurant, customer orders are queued in the kitchen. The first order placed is the first one prepared and served. When the food is ready, it is dequeued and delivered to the customer. This prevents confusion and ensures fair service.

Q9: Why is CHUK an example of a priority queue?

At CHUK hospital, emergency patients are treated before those with minor illnesses. This is not FIFO but a **priority queue.** Urgent cases are given higher priority to save lives. Without priority queues, critical patients might suffer by waiting in line.

Q10: How would queues match drivers and students in moto apps?

In moto or e-bike apps, both drivers and passengers (students) wait in queues. The first available driver is matched with the first student request. This ensures that no one is skipped unfairly and that rides are assigned in a systematic way.