

NAMES:MUTUYINEZA JOSELYNE

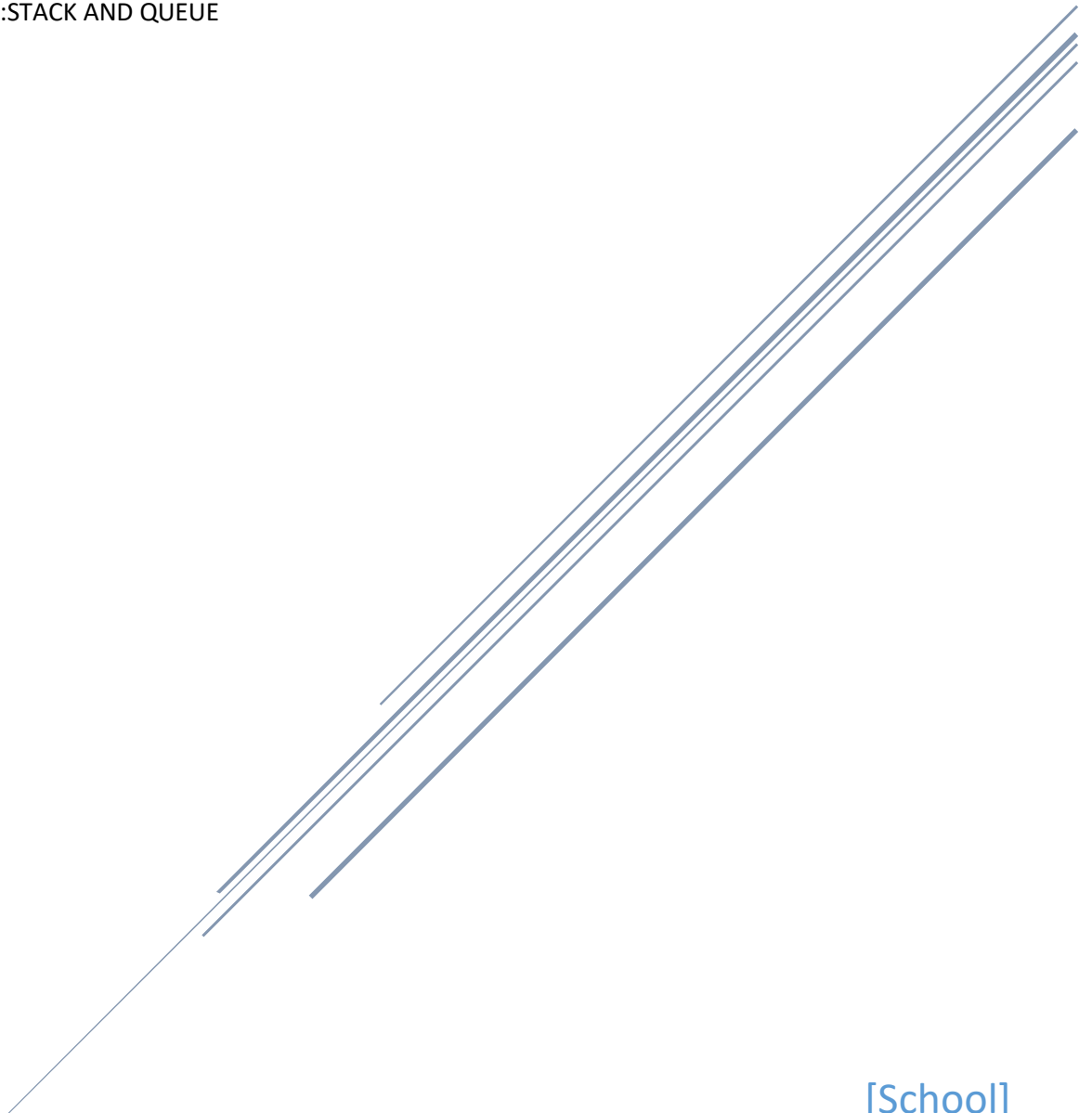
Reg No:224012018

DEPARTMENT:BIT

COLLEGE:CBE

MODULE TITLE:DATA STRUCTURE AND ALOGORTHM

EXSERCISES:STACK AND QUEUE



[School]
[Course title]

STACK

1. In the MoMo app, when you go step by step entering payment details, the last step you entered is the first one removed when you press back. This shows the LIFO principle because the most recent action is undone before older ones.
2. In Canvas, pressing back removes the last visited page or module. This is similar to Pop because the top item (last action) is removed first, leaving earlier steps in the stack.
3. In banking apps, each action like a transaction is pushed onto a stack. If a mistake is made, the undo function pops the last action, restoring the system to the previous state.
4. Balanced parentheses use stacks by pushing each opening bracket and popping when a matching closing bracket appears. Forms like Irembo can use this logic to ensure every opening field has a closing match, keeping data balanced.

5. The sequence ends with “CBE notes”, “Math revision”, then after popping “Debate” and pushing “Group assignment”. The top of the stack is Group assignment, which will be handled next.
6. If a student undoes three recent actions, the last three pushed items are popped out. Only the older answers remain in the stack, showing the state before the last three steps.
7. When booking on RwandAir, each form step is pushed. If the user wants to go back, the stack pops steps one by one, letting them retrace in exact reverse order.
8. To reverse “Umwana ni umutware”, push each word into the stack. Then popping out gives the words in reverse order: “umutware ni Umwana”.
9. DFS uses a stack because it explores one path deeply before backtracking. A queue would process level by level, but stacks suit deep search where backtracking is needed.
10. In BK Mobile, a stack can be used to move through transaction history. Each new page is pushed, and popping allows users to navigate backward step by step.

QUEUE

1. In restaurants, the first person to enter the line is the first served. This shows FIFO behavior where the earliest arrival is processed first.
2. In YouTube playlists, the next video plays after the current one ends. This is like dequeuing because the front item (first video in the line) is removed and played.
3. At RRA offices, people waiting form a line, where each new person joins the rear.
This is the same as enqueueing in a queue, and service happens in order.
4. Queues improve service by ensuring fairness—customers are served in the order they arrive. This prevents confusion and disorder.
5. After the operations, Alice leaves, so the order becomes [Eric, Chantal, Jean]. The person at the front now is Eric.
6. A queue processes items in the order they came. In RSSB applications, this ensures fairness because no one can jump ahead of others.
7. Linear queue works like a straight line at a buffet. Circular queue is like buses looping in Nyabugogo—after the last stop, it goes back to start. Deque is like boarding a bus from either the front or rear.

8. At a restaurant, orders are enqueued as customers request. When food is ready, the order is dequeued and delivered, keeping the process organized.
9. At CHUK, emergency patients are handled before others, even if they arrived later.
This is a priority queue because it doesn't follow simple FIFO but serves by urgency.
10. In a moto/e-bike app, students' requests are enqueued, and drivers dequeue them in order. This ensures fair matching without skipping.