Assignment 5: Queue

Instructions:-

1. **You need to upload a pdf file for this assignment**
2. **Format:- Q1 code, the screenshot of Q1 output, Q2 code, Q2 screenshot of output, and so on in the sequence.**
3. **File Naming convention:- ID_Lab05.pdf**
    a. **Eg:- 202011002_Lab05.pdf**

1. Create a class called IntQueue, implement an integer queue using an array.
    - Implement the following functions:
        i. create: Create a new, empty queue object.
        ii. empty: Determine whether the queue is empty; return true if it is and false if it is not.
        iii. enqueue: Add a new element at the rear of a queue.
        iv. dequeue: Remove an element from the front of the queue and return it. (This operation cannot be performed if the queue is empty.)
        v. front: Return the element at the front of the queue (without removing it from the queue). (Again, this operation cannot be performed if the queue is empty.)

2. Implement Circular Queue
    - Implement the above-mentioned functions of Question 1

3. Implement Cache using Queues of size 3 and return the number of page faults.
    - The Caching scheme is to remove the least recently used frame when the cache is full and a new page is referenced which is not there in the cache.
    - Page fault:- Page fault occurs when the required page is not available in the queue.

i. If the queue is not full and the required page is not available then add it into the queue. It is also counted as a page fault.
ii. If the queue is full and the required page is not available, then remove the least recently accessed page.

INPUT:- 1 2 3 4 2 5 1

OUTPUT:- 6

EXPLANATION: empty -> 1 (first page fault) -> 1 2 (second page fault) -> 1 2 3(third page fault) -> 2 3 4(fourth page fault) -> 2 3 4 -> 3 4 5 (fifth page fault) -> 4 5 1(sixth page fault)

4. Implement stack using queues(do not change queue functionalities ie:- remove the item from the front only not from the rear)

5. In the last lab, we implemented the stack. Now we have stack and queue. Our task is to compute an alternating series using the numbers that you entered. An alternating series switches the sign of the number being added for every other number. For example, if we enter the numbers: 1, 3, 15, 9 then we would compute the alternating series as 1 - 3 + 15 - 9 = 4. Write the Queue code so that it computes the sum of an alternating series of the numbers as they are removed from the queue (so the first number is positive, the second is negative, etc.) in the deque function. Display the alternating series to the screen on a single line. So if the numbers 1, 3, 15, and 9 were entered by the user, the code should display: 1 - 3 + 15 - 9 = 4

INPUT:- 1 3 15 9

OUTPUT:- 1 - 3 + 15 - 9 = 4