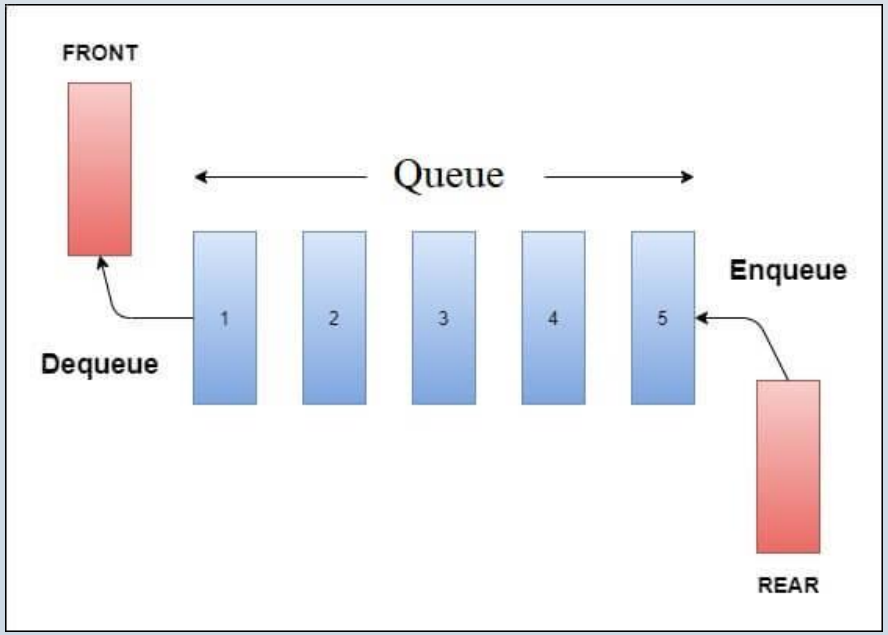
Stack

TUAN NGUYEN

Queue

Queue

- Queue is a collection of objects that are inserted and removed according to the first-in, first-out (FIFO) principle.



Queue Abstract Data Type

- Important methods:
  - **Q.enqueue(e)**: Add element e to the back of queue Q.
  - **Q.dequeue()**: Remove and return the first element from queue Q; an error occurs if the queue is empty.
- Other methods:
  - **Q.first()**: Return a reference to the element at the front of queue Q, without removing it; an error occurs if the queue is empty.
  - **Q.is empty():** Return True if queue Q does not contain any elements.
  - **len(Q)**: Return the number of elements in queue Q; in Python, we implement this with the special method len .

Queue Abstract Data Type (1)

Operation	Return Value	first $\leftarrow Q \leftarrow$ last
Q.enqueue(5)	–	[5]
Q.enqueue(3)	–	[5, 3]
len(Q)	2	[5, 3]
Q.dequeue( )	5	[3]
Q.is_empty( )	False	[3]
Q.dequeue( )	3	[ ]
Q.is_empty( )	True	[ ]
Q.dequeue( )	"error"	[ ]
Q.enqueue(7)	–	[7]

Python Queue Implementation

- The queue class maintains the following three instance variables:
  - **_data**: is a reference to a list instance with a fixed capacity.
  - **_size**: is an integer representing the current number of elements stored in the queue (as opposed to the length of the data list).
  - **_front**: is an integer that represents the index within data of the first element of the queue (assuming the queue is not empty).

Array Circularly

Figure 6.6: Modeling a queue with a circular array that wraps around the end.