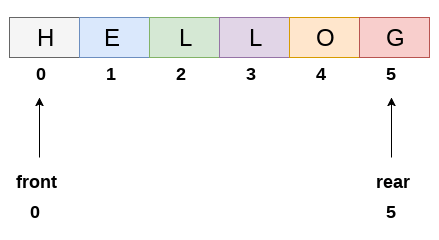
**CS106.3**

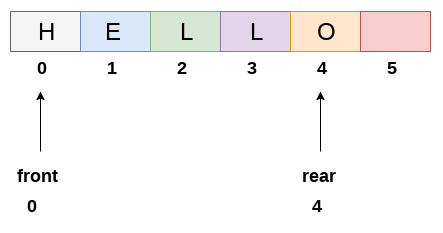
**Algorithms and Data Structures**

**Tutorial 3**

**Queue**

1. Draw the logical representation of Queue?





1. There are three instances which queue could be empty. Mention those?

front is at -1

front becomes greater than rear

Front is equal to rear

1. Implement Queue data structure using enqueue() dequeue() and display() functions. Use Switch case to get test cases from user.

**void** **enqueue**(**int** item)

{

**if** ((back + 1) % capacity == front)

print("Queue if full!")

**else**

{

back = (back+1) % capacity

arr[back] = item

**if**(front == -1)

front = back

}

}

**int** **dequeue**()

{

**if**(isEmpty() == True)

{

print("Queue is empty!")

**return** 0

}

**else**

{

**int** item = arr[front]

**if**(front == back)

front = back = -1

**else**

front = (front + 1) % capacity

**return** item

}

}

**int** **display**()

{

**if** (isEmpty() == True)

{

print("Queue is empty!")

**return** -1

}

**else**

**return** arr[front]

}

bool **isEmpty**()

{

**if** (front == -1)

**return** True

**else**

**return** False

}