

# PROGRAMMING CONCEPTS AND EMBEDDED PROGRAMMING IN

C, C++ and JAVA:

## Lesson-3: Data Structures: Queues, Pipes and Sockets

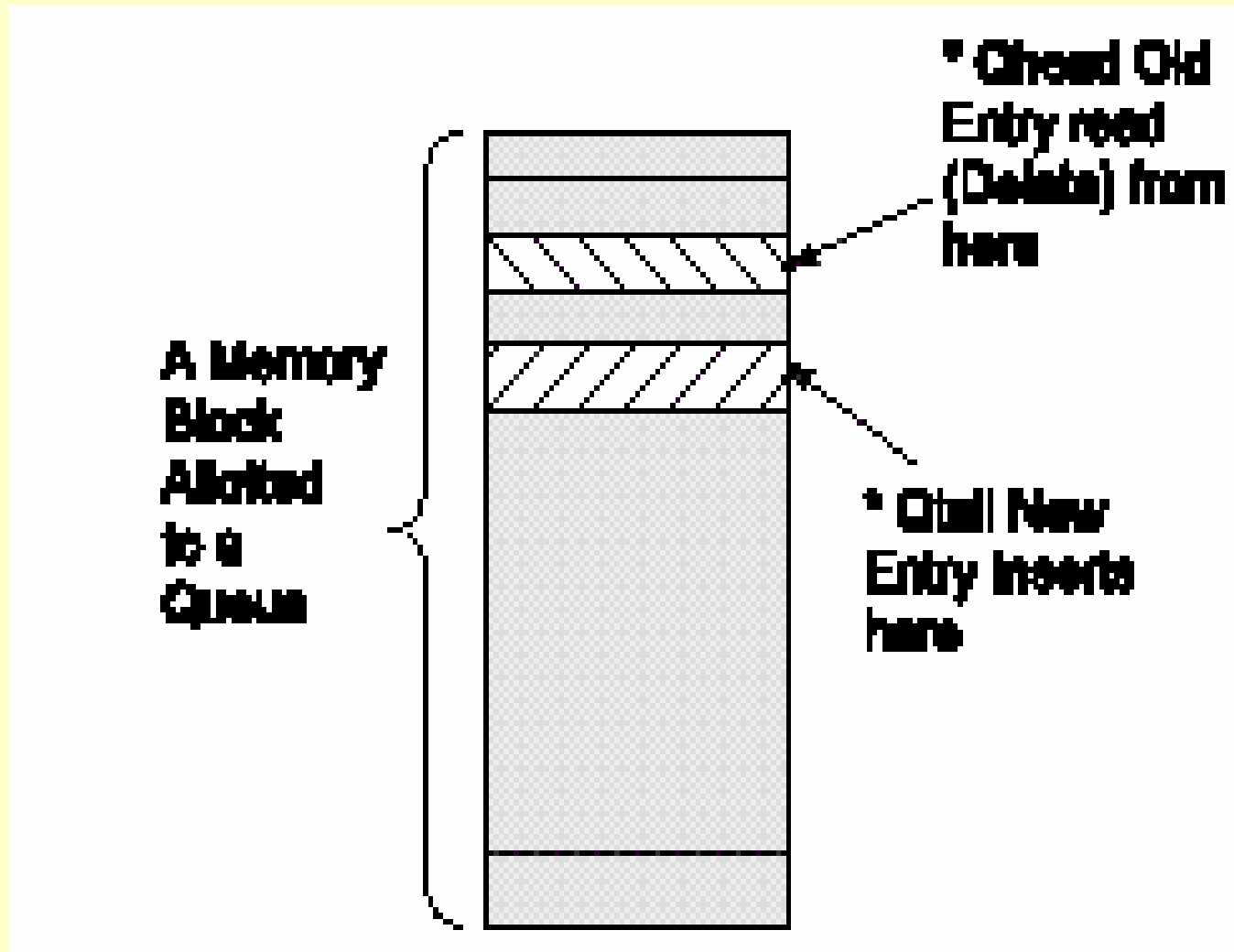
# 1. QUEUE

# QUEUE

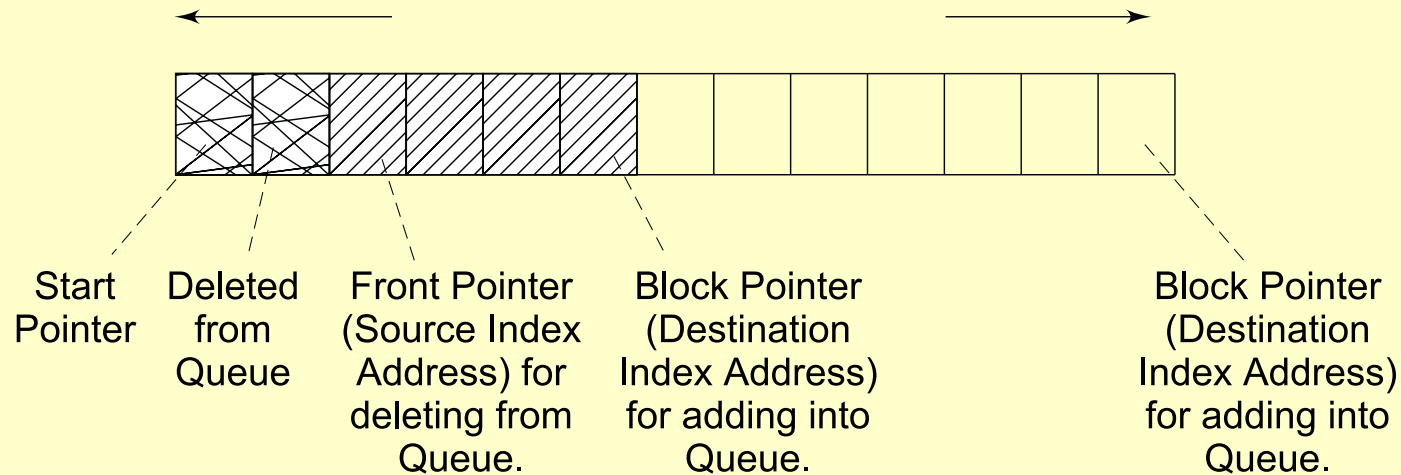
- A structure with a series of data elements with the first element waiting for an operation.
- Used when an element is not to be accessed by index with a pointer directly, as in an array, but only through FIFO (first in first out) mode through a queue-head pointer.

- An element can be inserted only at the end (also called tail or back) in series of elements waiting for an operation and deleted from front (queue-head).
- There are two pointers, one for *deleting* after the read operation from the head and other for *inserting* at the tail. Both increment after an operation.

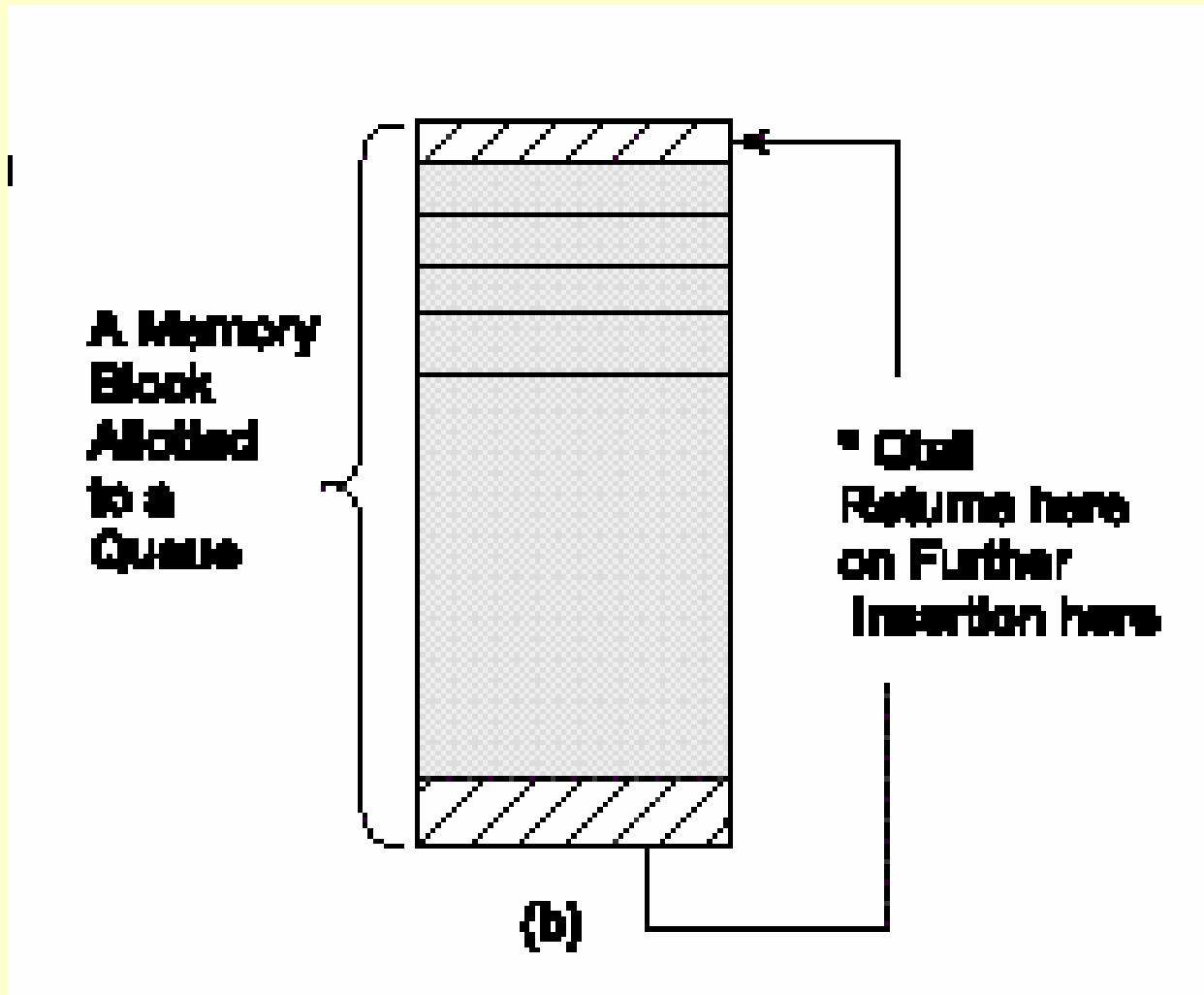
## Queue at a memory block



**A queue at a memory block with two pointers to point its two elements at the front and back. A data word always retrieves in FIFO mode from a queue**

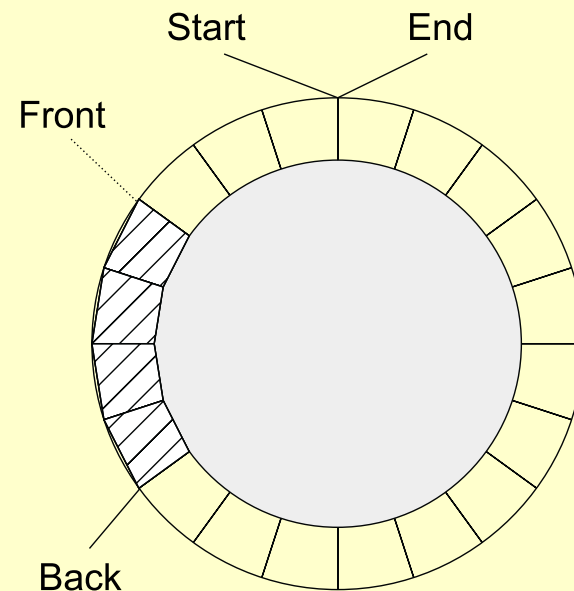


## Circular Queue at a memory block



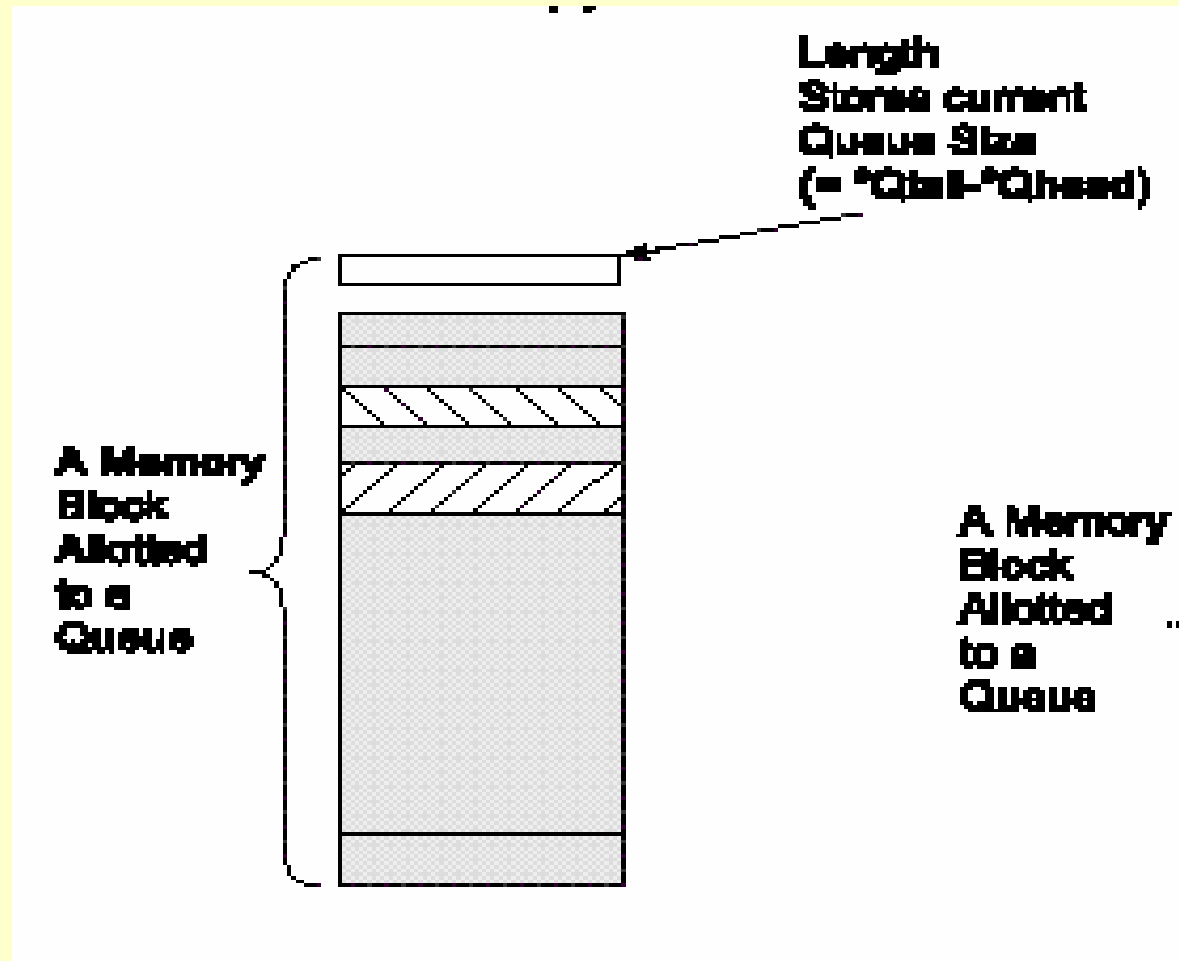
**A circular queue at a memory block with two pointers to point its two elements at the front and back. A pointer on reaching a limit of the block returns to the start of the block**

For Circular Queue, when back Attempts to exceed end, back becomes equal to start.

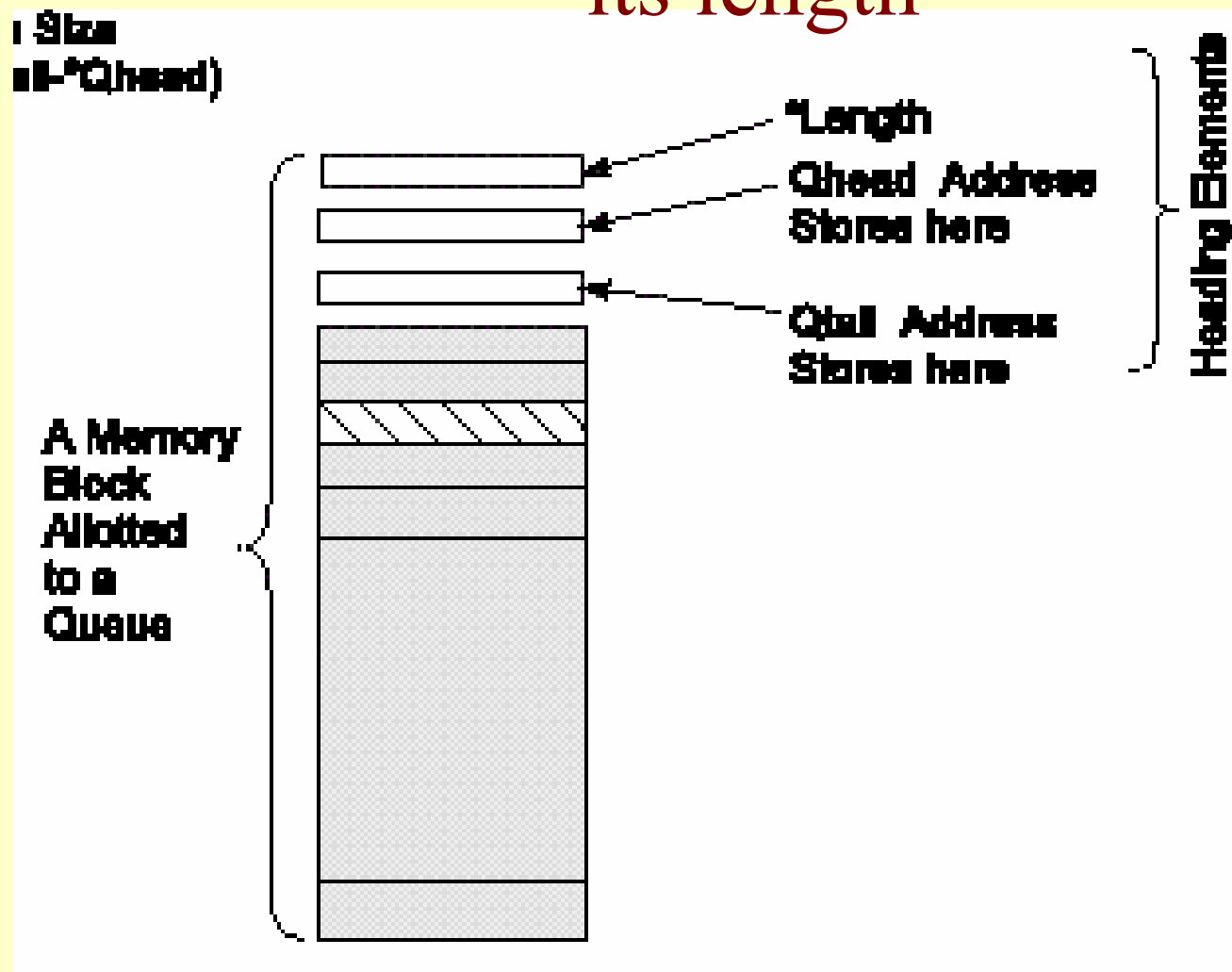




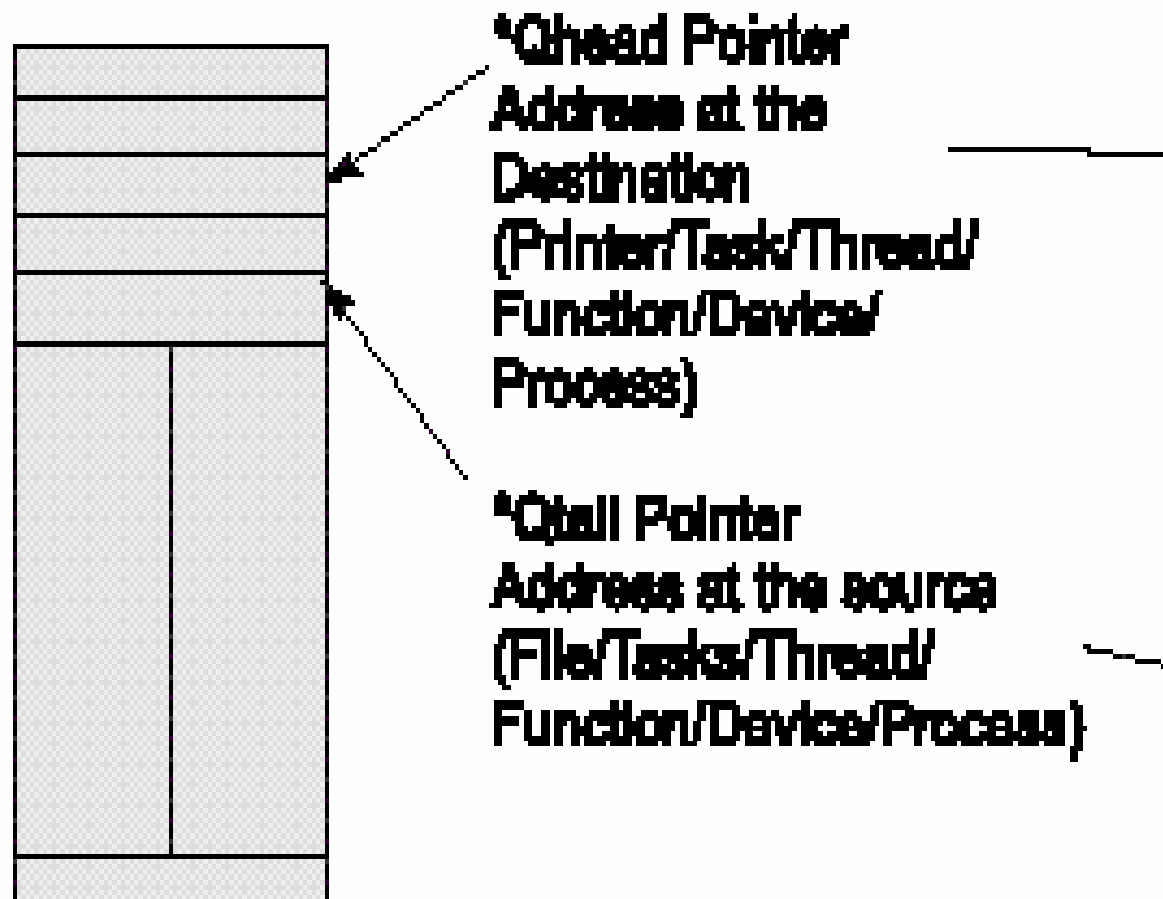
## Queue with a header for its length at a memory block



# Memory block for Queue with a header for its length



## Queue with a header for its length at a memory block



## **Queue with a header for length and source-address and destination-addresses at a memory block**

- When a byte stream is sent on a network, the bytes for the header for the length of the stream and for the source and destination addresses of the stream are must. [Note: There may be other header bits, for example, in the IP protocol. There may trailing bytes for examples, in HDLC and Ethernet protocols.]

## Standard Functions used in a queue

1. QELInsert – Insert an element into the queue as pointed by \*qtail and increment the qtail pointer address
2. QELReturn – Return an element into the queue as pointed by \*qhead and the element deletes from queue on increment in the qhead pointer address— return also means delete

3. isQNotEmpty— Return true or false after the check for the queue not empty.

# Priority Queue

When there is an urgent message to be placed in a queue, we can program such that a priority data element inserts at the head instead of at the tail. That element retrieves as if there is last-in first-out.

## Application Example of a Queue

- Networking applications need the specialized formations of a queue. On a network, the bits are transmitted in a sequence and retrieved at the other end in a sequence. To separate the bits at the different blocks or frames or packets, there are header bytes.



## Queue with header bytes

- Queues of bytes in a stream play a vital role in a network communication.
- Network queues have headers for length, source and destination addresses and other bytes as per the protocol used

## Queue with header bytes

- The header with the queue data elements (forming a byte stream) follows a protocol. A protocol may also provide for appending the bytes at the queue tail. These may be the CRC (Cyclic Redundancy Check) bytes at the tail.

# Data streams flow control

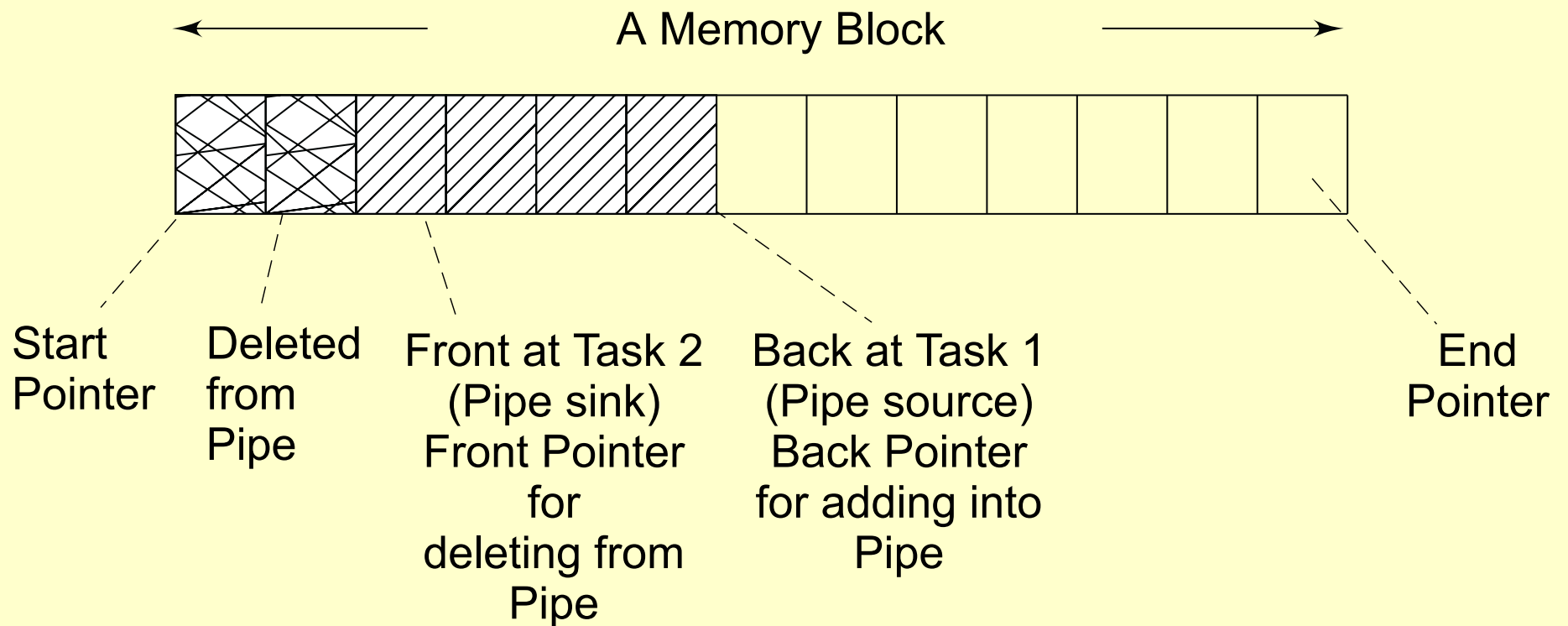
- Use a special construct FIPO (First-in Provisionally Out)
- FIPO is a queue special construct in which deletion is provisional and head pointer moves backward as per the last acknowledged (successfully read) byte from source of deletion in the network

# 3. Pipe

# Pipe

- A pipe is a virtual device for sending byte or data-element streams at the tail and retrieving from head but using pipe- device driver functions [create ( ), open ( ), connect ( ), read ( ), write ( ), close ( )] and device descriptor, *fd*
- Elements from a pipe deletes from the head at a destination and inserts into the tail at the source.
- Source and destination may be two physical devices on two systems or at the same system.

## *A memory block with a pipe*



# Pipe Device driver Functions

- Create ( )
- Connect ( ) for connect pipe between two addresses for read and write, respectively
- Open ( )
- read ( )
- Write ( )
- Close ( )

## 4. Socket



## Socket

- A socket is a virtual device used to send the packets from an addressed port at the source to another addressed port at the destination. Packets are the blocks of byte-streams. Each byte stream having the header and/or trailing bytes.

# Socket Device driver Functions

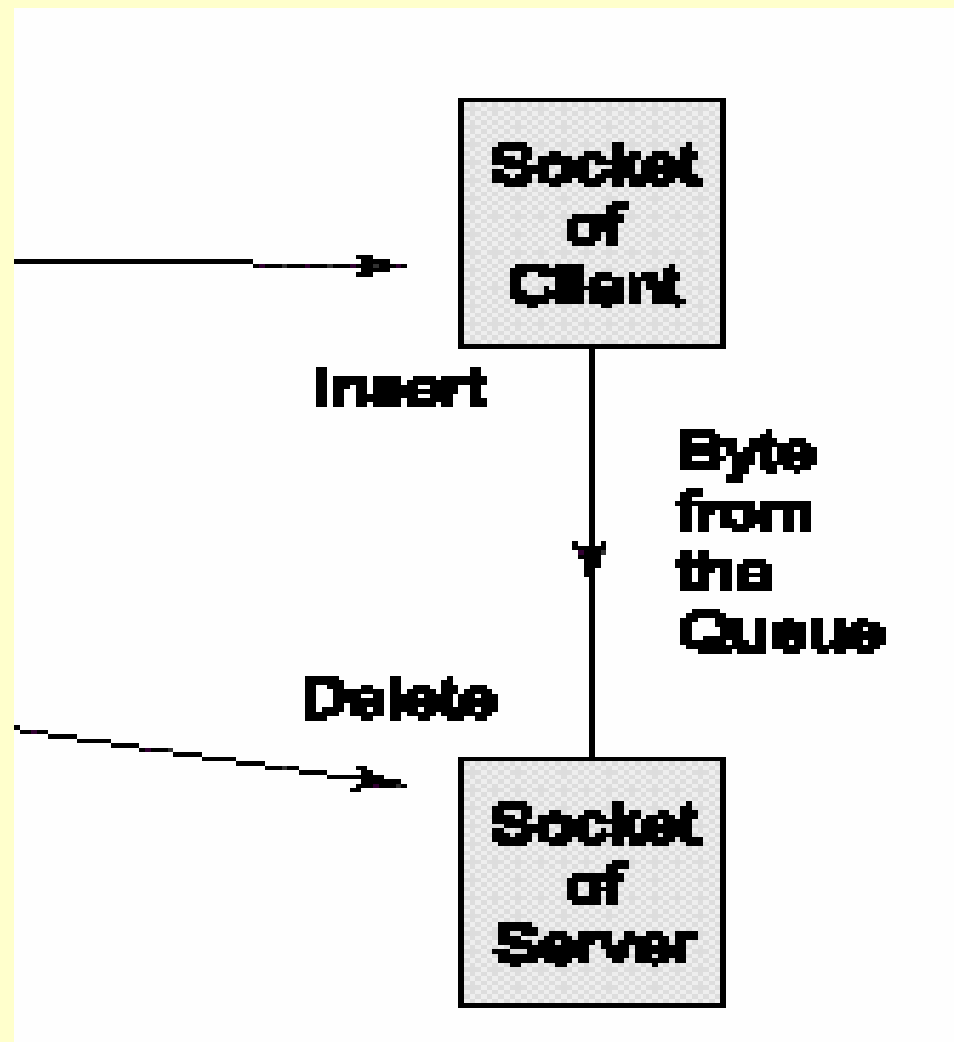
- create ( )
- connect ( ) for two ports at two addresses for read and write, respectively
- listen ( )
- accept ( )
- open ( )
- read ( )
- write ( )
- close ( )

# Socket Device driver Functions

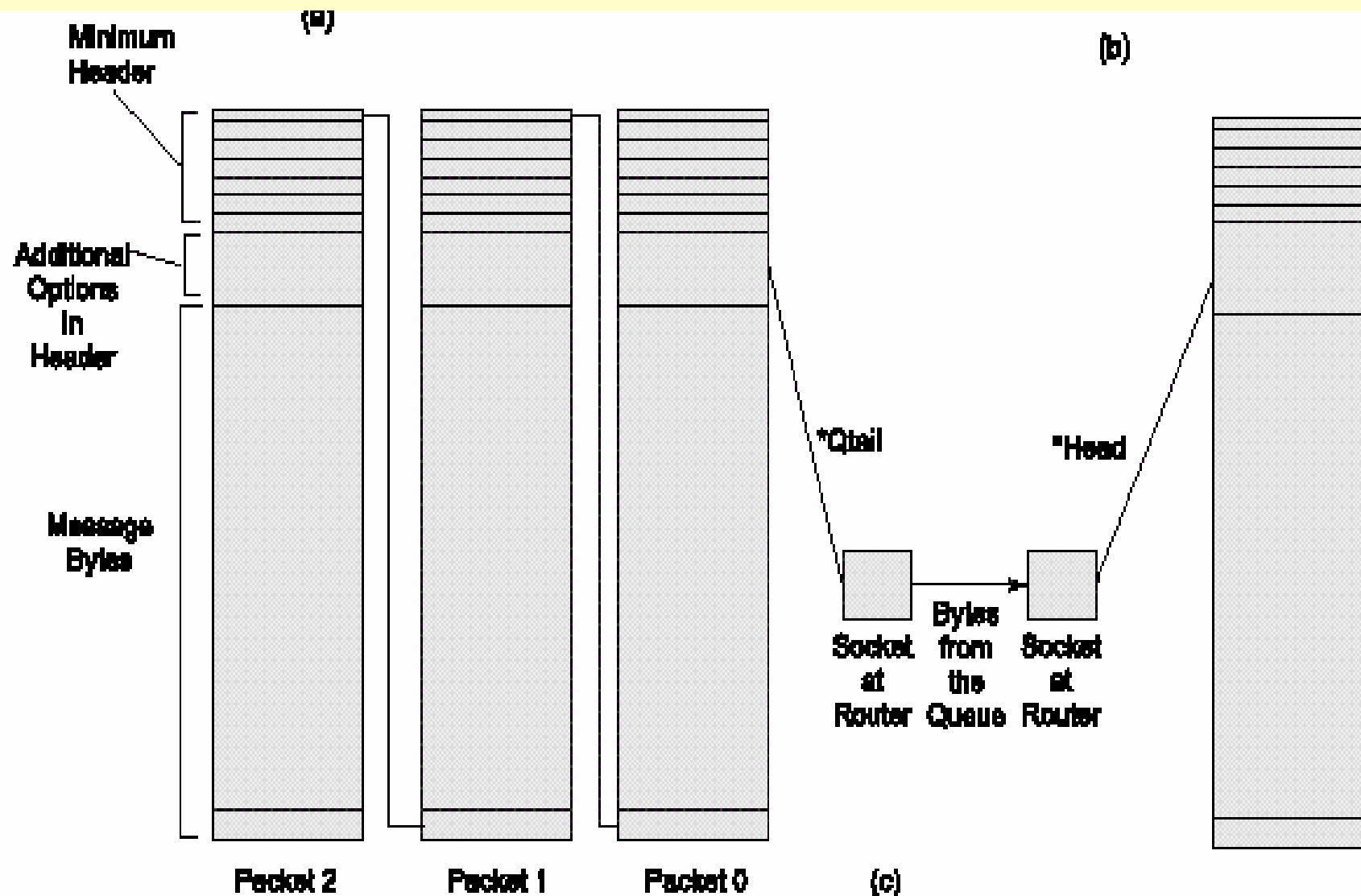
## write ( ) and read ( )

write ( )

read ( )



# Network Socket Message bytes with header as per protocol and sending data and receiving data using Device driver Functions `write ( )` and `read ( )`



# Summary

## We learnt

- Queue is an important data structure used in a program.
- The queue data structure related functions are 'constructing' a queue, 'inserting' an element into it, deleting an element from it and 'destruction' of queue.
- A queue is a first-in first-out data structure.

# We learnt

- Queue of function pointers has applications in making possible the designing of ISRs with short codes and by running the functions in the ISRs at the later stages

- Queues of bytes in a stream play a vital role in a network communication.
- Network queues have headers for length, source and destination addresses and other bytes as per the protocol used
- Virtual devices— pipes and sockets. Pipe and socket also retrieves the data as queue



# End of Lesson 3 of Chapter 5