

1. Limitations of the conventional pipes:	<p>(1) They cannot be used to communicate between unrelated processes (i.e., processes not sharing the same ancestor in which the pipe was created)</p> <p>(2) They are not permanent. They vanish when the relevant processes terminate.</p> <p>To address this problem, Unix systems introduced FIFO or named pipes (because a FIFO is a special file with a file name.)</p>	6. Shared Memory	<p>a shared memory is a piece of physical memory in the kernel. Multiple processes can each map a piece of their own virtual address space to that same physical memory, thus sharing it with each other. A shared memory provides a simple and fast method for multiple processes to communicate and exchange data with each other. However, the access to the shared memory often must be controlled to ensure the integrity of the data. The access control is often implemented with the use of one or more semaphores.</p>
2. Message Queue:	a linked list of messages, each with a priority number. Multiple processes may access (post and retrieve) messages to/from the message queue at any order, not necessarily in the first-in-first-out order.	7. What do pipes do?	<p>A pipe provides a one-way communication channel between related processes.</p> <ul style="list-style-type: none"> • A pipe can be used only to communicate between processes sharing a common ancestry in which the pipe was created. • A pipe is uni-directional from the write end to the read end, although more than one process can read from (or write to) the same pipe • A pipe is implemented with a FIFO buffer in the memory. A read is blocked if the pipe is empty. A write is blocked if the pipe is full.
3. Message Queue, Semaphores and Shared Memory are what and originated where?	<p>IPCs (interprocess communications.)</p> <p>Originated from Systems V</p>		
4. Processes do not normally share memory space, so how do they communicate with each other?	<p>Unix systems provide several inter-process communication mechanisms (IPCs) to suit different applications:</p> <p>BSD & SV pipes</p> <ul style="list-style-type: none"> -FIFOS -Stream pipes -named stream pipes <p>SV</p> <ul style="list-style-type: none"> -message queues -semaphores -shared memory 		
5. Semaphores:	semaphores provide controlled access to shared resources, such as a shared file, or a shared memory. They are often used to implement mutual exclusions and synchronizations.		