Quizlet

Operating Systems - Chpt 6

1.	**6.1 Give examples of reusable and consumable resources.**	Examples of reusable resources are processors, I/O channels, main and secondary memory, devices, and data structures such as files, databases, and semaphores. Examples of consumable resources are interrupts, signals, messages, and information in I/O buffers.F
2.	**6.2 What are the three conditions that must be present for deadlock to be possible?**	Mutual exclusion. Only one process may use a resource at a time. Hold and wait. A process may hold allocated resources while awaiting assignment of others. No preemption. No resource can be forcibly removed from a process holding it
3.	**6.3 What are the four conditions that create deadlock?**	The above three conditions, plus: Circular wait. A closed chain of processes exists, such that each process holds at least one resource needed by the next process in the chain.s
4.	**6.4 How can the hold-and-wait condition be prevented?**	The hold-and-wait condition can be prevented by requiring that a process request all of its required resources at one time, and blocking the process until all requests can be granted simultaneously
5.	**6.5 List two ways in which the no-preemption condition can be prevented.**	First, if a process holding certain resources is denied a further request, that process must release its original resources and, if necessary, request them again together with the additional resource. Alternatively, if a process requests a resource that is currently held by another process, the operating system may preempt the second process and require it to release its resources.
6.	**6.6 How can the circular wait condition be prevented?**	The circular-wait condition can be prevented by defining a linear ordering of resource types. If a process has been allocated resources of type R, then it may subsequently request only those resources of types following R in the ordering.
7.	**6.7 What is the difference among deadlock avoidance, detection, and prevention?**	Deadlock prevention constrains resource requests to prevent at least one of the four conditions of deadlock; this is either done indirectly, by preventing one of the three necessary policy conditions (mutual exclusion, hold and wait, no preemption), or directly, by preventing circular wait. Deadlock avoidance allows the three necessary conditions, but makes judicious choices to assure that the deadlock point is never reached. Deadlock detection, requested resources are granted to processes whenever possible.; periodically, the operating system performs an algorithm that allows it to detect the circular wait condition.