**CO4**

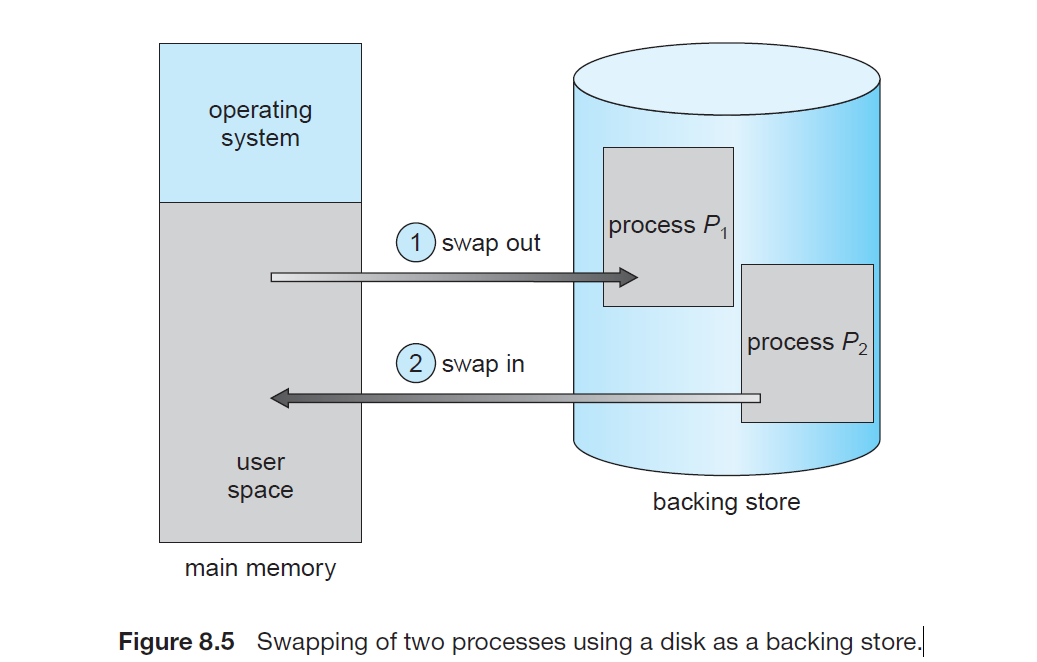
**MEMORY MANAGEMENT**

**SWAPPING:**

A process must be in memory to be executed. Swapping is a memory management technique that a process can be **swapped** temporarily out of memory to a **backing store** and then brought back into memory for continued execution. It is used to improve main memory utilization. In secondary memory, the place where the swapped-out process is stored is called swap space. Swapping is done so that other processes get memory for their execution.

**Standard Swapping:**

Standard swapping involves moving processes between main memory and a backing store. The backing store is commonly a fast disk. It must be large enough to accommodate copies of all memory images for all users, and it must provide direct access to these memory images. The system maintains a ready queue consisting of all processes whose memory images are on the backing store or in memory and are ready to run. Whenever the CPU scheduler decides to execute a process, it calls the dispatcher. The dispatcher checks to see whether the next process in the queue is in memory. If it is not, and if there is no free memory region, the dispatcher swaps out a process currently in memory and swaps in the desired process. It then reloads registers and transfers control to the selected process.



The above diagram shows swapping of two processes where the disk is used as a Backing store. There are two steps involved in swapping:

**a) Swap-in:** A swap-in process in which a process moves from secondary storage / hard disk to main memory (RAM).

**b) Swap-out:** Swap out takes a process out of the main memory and places it in secondary memory.

In the above diagram, suppose there is a multiprogramming environment with a round-robin scheduling algorithm; whenever the time quantum expires then the memory manager starts to swap out those processes that are just finished and swap another process into the memory that has been freed. And in the meantime, the CPU scheduler allocates the time slice to some other processes in the memory.

In priority-based scheduling algorithm. If any higher-priority process arrives and wants service, then the memory manager swaps out lower priority processes and then load the higher priority processes and then execute them. When the process with higher priority finishes, then the process with lower priority swapped back in and continues its execution. This variant is sometimes known as roll in and roll out.

Due to the swapping technique performance usually gets affected, but it also helps in running multiple and big processes in parallel. **The swapping** process is also known as a technique for **memory compaction.**

Standard swapping is not used in modern operating systems. It requires too much swapping time and provides too little execution time to be a reasonable memory-management solution. Modified versions of swapping, however, are found on many systems, including UNIX, Linux, and Windows. Another variation involves swapping portions of processes—rather than entire processes—to decrease swap time. Typically, these modified forms of swapping work in conjunction with virtual memory.