# Operating System-1

Following are a few basic questions that cover the essentials of OS:

**1). Explain the concept of Reentrancy.**
**Answer:**
It is a useful, memory-saving technique for multiprogrammed timesharing systems. A Reentrant Procedure is one in which multiple users can share a single copy of a program during the same period. Reentrancy has 2 key aspects: The program code cannot modify itself, and the local data for each user process must be stored separately. Thus, the permanent part is the code, and the temporary part is the pointer back to the calling program and local variables used by that program. Each execution instance is called activation. It executes the code in the permanent part, but has its own copy of local variables/parameters. The temporary part associated with each activation is the activation record. Generally, the activation record is kept on the stack.

## Note:

A reentrant procedure can be interrupted and called by an interrupting program, and still execute correctly on returning to the procedure.

**2).Explain Belady's Anomaly.**
**Answer:**
Also called FIFO anomaly. Usually, on increasing the number of frames allocated to a process' virtual memory, the process execution is faster, because fewer page faults occur. Sometimes, the reverse happens, i.e., the execution time increases even when more frames are allocated to the process. This is Belady's Anomaly. This is true for certain page reference patterns.

**3). What is a binary semaphore? What is its use?**
**Answer:**
A binary semaphore is one, which takes only 0 and 1 as values. They are used to implement mutual exclusion and synchronize concurrent processes.

**4). What is thrashing?**
**Answer:**
It is a phenomenon in virtual memory schemes when the processor spends most of its time swapping pages, rather than executing instructions. This is due to an inordinate number of page faults.

**5. List the Coffman's conditions that lead to a deadlock.**
**Answer:**

- Mutual Exclusion: Only one process may use a critical resource at a time.
- Hold & Wait: A process may be allocated some resources while waiting for others.
- No Pre-emption: No resource can be forcible removed from a process holding it.
- Circular Wait: A closed chain of processes exist such that each process holds at least one resource needed by another process in the chain.

**6. What are short-, long- and medium-term scheduling?**
**Answer:**

## Long Term Scheduler:->

Long term scheduler determines which programs are admitted to the system for processing. It controls the degree of multiprogramming. Once admitted, a job becomes a process.

## Medium Term Scheduler:->

Medium term scheduling is part of the swapping function. This relates to processes that are in a blocked or suspended state. They are swapped out of real-memory until they are ready to execute. The swapping-in decision is based on memory-management criteria.

## Short Term Scheduler:->

Short term scheduler, also know as a dispatcher executes most frequently, and makes the finest-grained decision of which process should execute next. This scheduler is invoked whenever an event occurs. It may lead to interruption of one process by preemption.

**7. What are turnaround time and response time?**
**Answer:**
Turnaround time is the interval between the submission of a job and its completion. Response time is the interval between submission of a request, and the first response to that request.

**8. What are the typical elements of a process image?**
**Answer:**

- User data: Modifiable part of user space. May include program data, user stack area, and programs that may be modified.
- User program: The instructions to be executed.
- System Stack: Each process has one or more LIFO stacks associated with it. Used to store parameters and calling addresses for procedure and system calls.
- Process control Block (PCB): Info needed by the OS to control processes.

**9. What is the Translation Lookaside Buffer (TLB)?**
**Answer:**
In a cached system, the base addresses of the last few referenced pages is maintained in registers called the TLB that aids in faster lookup. TLB contains those page-table entries that have been most recently used. Normally, each virtual memory reference causes 2 physical memory accesses-- one to fetch appropriate page-table entry, and one to fetch the desired data. Using TLB in-between, this is reduced to just one physical memory access in cases of TLB-hit.

**10. What is the resident set and working set of a process?**
**Answer:**
Resident set is that portion of the process image that is actually in real-memory at a particular instant. Working set is that subset of resident set that is actually needed for execution. (Relate this to the variable-window size method for swapping techniques.)

**11. When is a system in safe state?**
**Answer:**
The set of dispatchable processes is in a safe state if there exists at least one temporal order in which all processes can be run to completion without resulting in a deadlock.

**12. What is cycle stealing?**

**Answer:**
We encounter cycle stealing in the context of Direct Memory Access (DMA). Either the DMA controller can use the data bus when the CPU does not need it, or it may force the CPU to temporarily suspend operation. The latter technique is called cycle stealing. Note that cycle stealing can be done only at specific break points in an instruction cycle.

**13. What is meant by arm-stickiness?**
**Answer:**
If one or a few processes have a high access rate to data on one track of a storage disk, then they may monopolize the device by repeated requests to that track. This generally happens with most common device scheduling algorithms (LIFO, SSTF, C-SCAN, etc). High-density multisurface disks are more likely to be affected by this than low density ones.

**14. What are the stipulations of C2 level security?**
**Answer:**
C2 level security provides for:

- Discretionary Access Control
- Identification and Authentication
- Auditing
- Resource reuse

**15. What is busy waiting?**
**Answer:**
The repeated execution of a loop of code while waiting for an event to occur is called busy-waiting. The CPU is not engaged in any real productive activity during this period, and the process does not progress toward completion.

**16.Explain the popular multiprocessor thread-scheduling strategies.**
**Answer:**

## Load Sharing:

Processes are not assigned to a particular processor. A global queue of threads is maintained. Each processor, when idle, selects a thread from this queue. Note that load balancing refers to a scheme where work is allocated to processors on a more permanent basis.

## Gang Scheduling:

A set of related threads is scheduled to run on a set of processors at the same time, on a 1-to-1 basis. Closely related threads / processes may be scheduled this way to reduce synchronization blocking, and minimize process switching. Group scheduling predated this strategy.

## Dedicated processor assignment:

Provides implicit scheduling defined by assignment of threads to processors. For the duration of program execution, each program is allocated a set of processors equal in number to the number of threads in the program. Processors are chosen from the available pool.

## Dynamic scheduling:

The number of thread in a program can be altered during the course of execution.

**17. When does the condition 'rendezvous' arise?**
**Answer:**
In message passing, it is the condition in which, both, the sender and receiver are blocked until the message is delivered.

**18. What is a trap and trapdoor?**
**Answer:**
Trapdoor is a secret undocumented entry point into a program used to grant access without normal methods of access authentication. A trap is a software interrupt, usually the result of an error condition.

**19. What are local and global page replacements?**
**Answer:**
Local replacement means that an incoming page is brought in only to the relevant process' address space. Global replacement policy allows any page frame from any process to be replaced. The latter is applicable to variable partitions model only.

**20. Define latency, transfer and seek time with respect to disk I/O.**
**Answer:**
Seek time is the time required to move the disk arm to the required track. Rotational delay or latency is the time it takes for the beginning of the required sector to reach the head. Sum of seek time (if any) and latency is the access time. Time taken to actually transfer a span of data is transfer time.

**21. Describe the Buddy system of memory allocation.**
**Answer:**
Free memory is maintained in linked lists, each of equal sized blocks. Any such block is of size 2^k. When some memory is required by a process, the block size of next higher order is chosen, and broken into two. Note that the two such pieces differ in address only in their kth bit. Such pieces are called buddies. When any used block is freed, the OS checks to see if its buddy is also free. If so, it is rejoined, and put into the original free-block linked-list.

**22. What is time-stamping?**
**Answer:**
It is a technique proposed by Lamport, used to order events in a distributed system without the use of clocks. This scheme is intended to order events consisting of the transmission of messages. Each system 'i' in the network maintains a counter Ci. Every time a system transmits a message, it increments its counter by 1 and attaches the time-stamp Ti to the message. When a message is received, the receiving system 'j' sets its counter Cj to 1 more than the maximum of its current value and the incoming time-stamp Ti. At each site, the ordering of messages is determined by the following rules: For messages x from site i and y from site j, x precedes y if one of the following conditions holds....(a) if Ti

**23. How are the wait/signal operations for monitor different from those for semaphores?**
**Answer:**
If a process in a monitor signal and no task is waiting on the condition variable, the signal is lost. So this allows easier program design. Whereas in semaphores, every operation affects the value of the semaphore, so the wait and signal operations should be perfectly balanced in the program.

**24. In the context of memory management, what are placement and replacement algorithms?**
**Answer:**
Placement algorithms determine where in available real-memory to load a program. Common methods are first-fit, next-fit, best-fit. Replacement algorithms are used when memory is full, and one process (or part of a process) needs to be swapped out to accommodate a new program. The replacement algorithm determines which are the partitions to be swapped out.

**25. In loading programs into memory, what is the difference between load-time dynamic linking and run-**

**time dynamic linking?**

**Answer:**

For load-time dynamic linking: Load module to be loaded is read into memory. Any reference to a target external module causes that module to be loaded and the references are updated to a relative address from the start base address of the application module.

With run-time dynamic loading: Some of the linking is postponed until actual reference during execution. Then the correct module is loaded and linked.

« Prev 12 Next »