OPERATING SYSTEM INTERVIEW QUESTIONS AND ANSWERS

Operating System Interview Question and Answers-1

1. **What is an operating system?**

   An operating system is a program that acts as an intermediary between the user and the computer hardware. The purpose of an OS is to provide a convenient environment in which user can execute programs in a convenient and efficient manner. It is a resource allocator responsible for allocating system resources and a control program which controls the operation of the computer hardware.

2. **Why paging is used?**

   Paging is solution to external fragmentation problem which is to permit the logical address space of a process to be noncontiguous, thus allowing a process to be allocating physical memory wherever the latter is available.

3. **Explain the concept of the batched operating systems?**

   In batched operating system the users gives their jobs to the operator who sorts the programs according to their requirements and executes them. This is time consuming but makes the CPU busy all the time.

4. **What is purpose of different operating systems?**

   The machine purpose workstation individual usability &resources utilization mainframe optimize utilization of hardware PC support complex games, business application Hand held PCs Easy interface & min. power consumption.

5. **What is virtual memory?**

   Virtual memory is hardware technique where the system appears to have more memory that it actually does. This is done by time-sharing, the physical memory and storage parts of the memory one disk when they are not actively being used.

6. **What is Throughput, Turnaround time, waiting time and Response time?**

   ***Throughput** : number of processes that complete their execution per time unit.*

   ***Turnaround time** : amount of time to execute a particular process.*

   ***Waiting time** : amount of time a process has been waiting in the ready queue.*

   ***Response time** : amount of time it takes from when a request was submitted until the firstresponse is produced, not output (for time-sharing environment).*

7. What are the various components of a computer system?

The hardware

The operating system

The application programs

The users.

8. What is a Real-Time System?

A real time process is a process that must respond to the event switch in a certain time period. A real time operating system is an operating system that can run real time processes successfully.

9. Explain the concept of the Distributed systems?

Distributed systems work in a network. They can share the network resources, communicate with each other.

10. What is SCSI?

SCSI - Small computer systems interface is a type of interface used for computer components such as hard drives, optical drives, scanners and tape drives. It is a competing technology to standard IDE (Integrated Drive Electronics).

11. What is a sector?

Smallest addressable portion of a disk.

12. What are the different operating systems?

Batched operating systems

Multi-programmed operating systems

Timesharing operating systems

Distributed operating systems

Real-time operating systems.

13. What is busy waiting?

The repeated execution of a loop of code while waiting for an event to occur is called busy waiting.

14. What are system calls?

System calls provide the interface between a process and the operating system. System calls for modern Microsoft windows platforms are part of the win32 API, which is available for all the compilers written for Microsoft windows.

15. What are various scheduling queues?

Job queue

Ready queue

Device queue

16. What are java threads?

Java is one of the small number of languages that support at the language level for the creation and management of threads. However, because threads are managed by the java virtual machine (JVM), not by a user-level library or kernel, it is difficult to classify Java threads as either user- or kernel-level.

17. What are types of threads?

User thread

Kernel thread

18. What is a semaphore?

It is a synchronization tool used to solve complex critical section problems. A semaphore is an integer variable that, apart from initialization, is accessed only through two standard atomic operations: Wait and Signal.

19. What is a deadlock?

Deadlock is a situation where a group of processes are all blocked and none of them can become unblocked until one of the other becomes unblocked. The simplest deadlock is two processes each of which is waiting for a message from the other.

20. What is cache memory?

Cache memory is random access memory (RAM) that a computer microprocessor can access more quickly than it can access regular RAM. As the microprocessor processes data, it looks first in the cache memory and if it finds the data there (from a previous reading of data), it does not have to do the more time-consuming reading of data.

21. What is thrashing?

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.

22. What are the states of a process?

New

Running

Waiting

Ready

Terminated

23. What is a binary semaphore?

A binary semaphore is one, which takes only 0 and 1 as values. They are used to implement mutual exclusion and synchronize concurrent processes.

24. What is a job queue?

When a process enters the system it is placed in the job queue.

25. What is a ready queue?

The processes that are residing in the main memory and are ready and waiting to execute are kept on a list called the ready queue.

26. What are turnaround time and response time?

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.

27. What are the operating system components?

Process management

Main memory management

File management

I/O system management

Secondary storage management

Networking

Protection system

Command interpreter system

28. What is mutex?

Mutex is a program object that allows multiple program threads to share the same resource, such as file access, but not simultaneously. When a program is started a mutex is created woth a unique name. After this stage, any thread that needs the resource must lock the mutex from other threads while it is using the resource. the mutex is set to unlock when the data is no longer needed or the routine is finished.

29. What is Marshalling?

The process of packaging and sending interface method parameters across thread or process boundaries.

30. What are residence monitors?

Early operating systems were called residence monitors.

31. Why thread is called as a lightweight process?

It is called light weight process to emphasize the fact that a thread is like a process but is more efficient and uses fewer resources(n hence "lighter")and they also share the address space.

32. What are operating system services?

Program execution

I/O operations

File system manipulation

Communication

Error detection

Resource allocation

Accounting

Protection

33. What is a process?

A program in execution is called a process. Or it may also be called a unit of work. A process needs some system resources as CPU time, memory, files, and i/o devices to accomplish the task. Each process is represented in the operating system by a

process control block or task control block (PCB).Processes are of two types

Operating system processes

User processes

34. What are the different job scheduling in operating systems?

Scheduling is the activity of the deciding when process will receive the resources they request.

FCFS *---> FCSFS stands for First Come First Served. In FCFS the job that has been waiting the longest is served next.*

Round Robin Scheduling*--->Round Robin scheduling is a scheduling method where each process gets a small quantity of time to run and then it is preempted and the next process gets to run. This is called time-sharing and gives the effect of all the processes running at the same time*

Shortest Job First *---> The Shortest job First scheduling algorithm is a nonpreemptive scheduling algorithm that chooses the job that will execute the shortest amount of time.*

Priority Scheduling*--->Priority scheduling is a scheduling method where at all times the highest priority process is assigned the resource.*

35. What is dual-mode operation?

In order to protect the operating systems and the system programs from the malfunctioning programs the two mode operations were evolved

System mode

User mode.

36. What is a device queue?

A list of processes waiting for a particular I/O device is called device queue.

37. What are the different types of Real-Time Scheduling?

Hard real-time systems required to complete a critical task within a guaranteed amount of time.

Soft real-time computing requires that critical processes receive priority over less fortunate ones.

38. What is starvation ?

Starvation is a resource management problem where a process does not get the resources it needs for a long time because the resources are being allocated to other processes.

39. What is a long term scheduler & short term schedulers?

Long term schedulers *are the job schedulers that select processes from the job queue and load them into memory for execution.*
The ***Short term schedulers*** *are the CPU schedulers that select a process form the ready queue and allocate the CPU to one of them.*

40. What is fragmentation?

Fragmentation occurs in a dynamic memory allocation system when many of the free blocks are too small to satisfy any request.

41. What is context switching?

Transferring the control from one process to other process requires saving the state of the old process and loading the saved state for new process. This task is known as context switching.

42. What is relative path and absolute path?

Absolute path-- Exact path from root directory.

Relative path-- Relative to the current path.

43. What are the disadvantages of context switching?

Time taken for switching from one process to other is pure over head. Because the system does no useful work while switching. So one of the solutions is to go for threading when ever possible.

44. What is the state of the processor, when a process is waiting for some event to occur?

Waiting state

45. What is the difference between Primary storage and secondary storage?

Main memory *- only large storage media that the CPU can access directly.*

Secondary storage *- extension of main memory that provides large nonvolatile*

storage capacity.

46. What is process synchronization?

A situation, where several processes access and manipulate the same data concurrently and the outcome of the execution depends on the particular order in which the access takes place, is called race condition. To guard against the race condition we need to ensure that only one process at a time can be manipulating the same data. The technique we use for this is called process synchronization.

47. What is a data register and address register?

Data registers *- can be assigned to a variety of functions by the programmer. They can be used with any machine instruction that performs operations on data.*
Address registers *- contain main memory addresses of data and instructions or they contain a portion of the address that is used in the calculation of the complete addresses.*

48. What are deadlock prevention techniques?

Mutual exclusion

Hold and wait

No preemption

Circular wait

49. What is the difference between Complier and Interpreter?

An interpreter reads one instruction at a time and carries out the actions implied by that instruction. It does not perform any translation. But a compiler translates the entire instructions.

50. What is a Safe State and what is its use in deadlock avoidance?

When a process requests an available resource, system must decide if immediate allocation leaves the system in a safe state. System is in safe state if there exists a safe sequence of all processes. Deadlock Avoidance : ensure that a system will never enter an unsafe state.

51. What is the difference between microkernel and macro kernel?

Micro-Kernel *: A micro-kernel is a minimal operating system that performs only the essential functions of an operating system. All other*

operating system functions are performed by system processes. ***Monolithic*** *: A monolithic operating system is one where all operating system code is in a single executable image and all operating system code runs in system mode.*

52. What is DRAM?

Dynamic Ram stores the data in the form of Capacitance, and Static RAM stores the data in Voltages.

53. What are the different functions of Scheduler?

Scheduler deals with the problem of deciding which of the process in the ready queue is to be allocated the CPU. Short Term Schedulers, Long Term Schedulers

54. What is a trap and trapdoor?

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.

55. What are local and global page replacements?

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.

56. What is cache-coherency?

In a multiprocessor system there exist several caches each may containing a copy of same variable A. Then a change in one cache should immediately be reflected in all other caches this process of maintaining the same value of a data in all the caches s called cache-coherency.

57. What are the benefits of multithreaded programming?

Responsiveness

Resources sharing

Economy

Utilization of multiprocessor architectures.

.

Operating System Interview Question and Answers-2

1. Explain the concept of Reentrancy?

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?

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?

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?

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.

1. **Mutual Exclusion**: Only one process may use a critical resource at a time.
2. **Hold & Wait**: A process may be allocated some resources while waiting for others.
3. **No Pre-emption**: No resource can be forcible removed from a process holding it.
4. **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?

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 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, 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?

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?

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)?

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?

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?

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?

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?

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?

C2 level security provides for:

1. Discretionary Access Control
2. Identification and Authentication
3. Auditing
4. Resource reuse

15. What is busy waiting?

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.

1. **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.
2. **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.
3. **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.
4. **Dynamic scheduling**: The number of thread in a program can be altered during the course of execution.

17. When does the condition 'rendezvous' arise?

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?

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?

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.

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.

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?

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<Tj or (b) if Ti=Tj and i<j.

23. How are the wait/signal operations for monitor different from those for semaphores?

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?

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?

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.

26. What are demand-paging and pre-paging?

With demand paging, a page is brought into memory only when a location on that page is actually referenced during execution. With pre-paging, pages other than the one demanded by a page fault are brought in. The selection of such pages is done based on common access patterns, especially for secondary memory devices.

27. Paging a memory management function, while multiprogramming a processor management function, are the two interdependent?

Yes.

28. What is page cannibalizing?

Page swapping or page replacements are called page cannibalizing.

29. What has triggered the need for multitasking in PCs?

1. Increased speed and memory capacity of microprocessors together with the support fir virtual memory and
2. Growth of client server computing

30. What are the four layers that Windows NT have in order to achieve independence?

1. Hardware abstraction layer
2. Kernel
3. Subsystems
4. System Services.

31. What is SMP?

To achieve maximum efficiency and reliability a mode of operation known as symmetric multiprocessing is used. In essence, with SMP any process or threads can be assigned to any processor.

32. What are the key object oriented concepts used by Windows NT?

Encapsulation, Object class and instance.

33. Is Windows NT a full blown object oriented operating system? Give reasons.

No Windows NT is not so, because its not implemented in object oriented language and the data structures reside within one executive component and are not represented as objects and it does not support object oriented capabilities.

34. What is a drawback of MVT?

It does not have the features like

1. ability to support multiple processors
2. virtual storage
3. source level debugging

35. What is process spawning?

When the OS at the explicit request of another process creates a process, this action is called process spawning.

36. How many jobs can be run concurrently on MVT?

15 jobs.

37. List out some reasons for process termination.

1. Normal completion
2. Time limit exceeded
3. Memory unavailable
4. Bounds violation
5. Protection error
6. Arithmetic error
7. Time overrun
8. I/O failure
9. Invalid instruction
10. Privileged instruction
11. Data misuse
12. Operator or OS intervention
13. Parent termination.

38. What are the reasons for process suspension?

1. swapping
2. interactive user request
3. timing
4. parent process request