Jaypee Institute of Information Technology
Department of Computer Science and Engineering
Course Name: Operating System and System Programming
Course Code: 15B11CI412

Tutorial-2

1.The services and functions provided by an operating system can be divided into two main categories. Briefly describe the two categories and discuss how they differ.

2. What are the five major activities of an operating system in regard to file management?

3. Describe three general methods for passing parameters to the operating system.

4. Describe the actions taken by a kernel to context-switch between processes.

5. Explain the characteristics of suspended process.

6. Explain whether following transitions between process states are possible or not. If possible, give the example.

1. Running-----Ready
2. Running-----Waiting
3. Waiting------Running
4. Running------ Terminated

7. Why is it important for the scheduler to distinguish I/O-bound programs from CPU-bound programs?

8. Discuss how the following pairs of scheduling criteria conflict in certain settings.

a. CPU utilization and response time

b. Average turnaround time and maximum waiting time

c. I/O device utilization and CPU utilization

9. Explain the different factor on which computation speed up of an application through concurrency and parallelism would depend.

10. An OS contain two process P1 and P2, with P2 having a higher priority than P1. Let P2 be blocked on an I/O operation and let P1 be running. What action take place when the I/O completion event occurs for the I/O operation of P2.

OSSP

Atishay Jain
2110 3285

Tutorial-2

Date.....

Q1 The services and functions can be provided by an OS can be divided into two main categories -

1. System Services → These manage hardware, memory, processors, devices, and security at a lower level. They include memory management, process scheduling, device management

2. User Services → These offer higher-level functionalities for applications and user interactions. They include file and data management, process communication, user interface services, utilities, and security mechanisms.

Q2. The five major activities of an operating system in regard to file management are :

1. File creating and deletion → OS creates and allocates space for new files, and removes files no longer needed.
2. The creation and deletion of directories.
3. The support of primitives for manipulating files and directories.
4. The mapping of files onto secondary storage.
5. The backup of files on stable (non volatile) storage media.

Date.....

Q3. There are three main methods to pass the parameters required for a system call-

1. Pass the parameters in registers (insufficient when there are more parameters than registers).
2. Store the parameters in a block, or table, in memory, and pass the address of block as a parameter in a register. (used by Linux and Solaris).
3. Push the parameters onto a stack; to be popped off by the OS. (they do not limit the no. or length of parameters passed).

Q4. Actions taken by a kernel to context switch between processes are-

1. The OS must save the PC and user stack pointer of the currently executing proces, in response to a clock interrupt and transfers control to the kernel clock interrupt handler.
2. Saving the rest of the registers, as well as other machine state, such as the state of the floating point registers, in the process PCB is done by the clock interrupt handler.
3. The scheduller to determine the next process to execute is invoked the OS.
4. Then the state of the next process from its PCB is retrieved by OS and restores the registers.

Date.....

Q5. The characteristics of a suspended process are →

1. Paused execution → A suspended process is temporarily halted, not actively running its instructions.
2. Memory Preserved → The process's memory and data are saved, allowing it to resume from where it left off.
3. Event dependency → It's waiting for a specific event, like user input or resource availability, to continue.
4. Resource conservation → While suspended, it doesn't consume CPU time, conserving system resources for other tasks.

Q6.

1) Running ... Ready → A process moves from "running" to the "ready" state when it is temporily paused to allow other processes to execute.

eg → A user is playing video game (running). It receives a notification that a download is complete. The game process moves from running to ready, allowing computer to allocate resources to the download.

2) Running ... Waiting → A process transition when it needs to wait for a specific event.

eg → A user is typing a document & wants to print it. The doc. processing software moves to waiting as it awaits the user to specify printing options.

Date.....

3. Waiting → Running → It is possible when the event it was waiting for occurs, allowing it to resume.
eg→ A file transfer program is waiting for connection & when it is established, the program transitions.

4. Running → Terminated → It is possible when it completes its task or is forcefully ended due to an error or user action.
eg→ A calculator program is open & the user press close button.

Q7. I/O Bound programs have the property of performing only a small amount of computation before performing I/O. Such programs typically don't use up their CPU quantum. CPU-bound programs use their entire quantum without performing any blocking I/O operations. Consequently, one could make better use of the computer's resources by giving higher priority to I/O bound programs and allow them to execute.

Q8.
a) These criteria conflict because high CPU utilization is often achieved by keeping the CPU busy with tasks. This can lead to lower response times for tasks that are already in the queue waiting. If CPU is less utilised, response time may improve but overall system efficiency might decrease.

Date.....

b) These conflict as reducing the maximum waiting time might involve giving it higher priority, leading to longer average turnaround time. Conversely, focussing on minimising average turnaround time might result in longer waiting time for process.

c) These conflict due to resource contention. I/O device utilization requires frequent interaction with I/O devices, leading to frequent context switches, affecting CPU utilization. conversely, high CPU utilization might mean the CPU is busy with computation tasks, causing I/O devices to remain idle, leading to lower I/O device utilization.

Q9.

Concurrency →

It is the ability of an application to manage multiple tasks or processes concurrently, without necessarily running them simultaneously. Speedup through concurrency depends on →

1) Task nature and Dependency
2) Task granularity
3) Resource Management
4) Scheduling
5) Synchronization Mechanisms.

Parallelism → It involves executing multiple tasks or inst. simultaneously on multiple processors or cores. Speedup dependson

1. Task decomposition
2. Task Dependency
3. Load Balancing
4. Communication Overhead.
5. Scalability
6. Architecture & Hardware
7. Data Access Patterns.

Spiral

Date.....

Q10. When the I/O completion event occurs for process P2 with higher priority, the OS's interrupt handler will mark P2 as ready to run, perform a context switch, and allow P2 to resume execution. This ensures that the higher priority process is given the CPU as soon as it becomes available after the I/O operation.