Unit 2: Functionalities Of Generalize Operating System

2

Unit Structure

2.1 LEARNING OBJECTIVE

After studying this chapter, students should be able to understand:

- Process management by the operating system
- Process scheduling and process life cycle
- Process Control Block (PCB)
- Concept of Threading
- Concept of Multitasking
- Memory Management
- Security objective and problems
- User management
- File management

2.2 INTRODUCTION

In the previous chapter we have discussed about the basics of operating system. The main functions of operating system are process management, memory management, security mechanism, user management, file management and device management.

In this chapter we will learn about all the functions performed by OS in detailed. We start with process management. Process refers to any activity done by an OS. As modern operating system provides multi-tasking environment, operating system need to manage number of processes that exist at the same time. OS needs to manage inter process communication, creation of process and status of process from creation to destruction. Operating system implements the multitasking concept by implementing the threading mechanism. The multi-tasking operating system needs to handle memory management. The multiple processes exist in the memory, each process demands memory dynamically. So operating system needs to make partition the memory to be allocated and to be reallocated. This memory partition needs to manage by the operating system.

The user of operating system stores the data in to the secondary device, generally in hard disk. This data is in the form of files and stored in hard disk. Without any proper mechanism it is too difficult to retrieve it from secondary device.

File system is the convenient mechanism to store data file in proper format. Computer system is accessed by the user. Each user has specific privilege to access set of resources and files. Operating system needs to manage its user to provide privilege for the resources. At the end we will discuss about the security issues of the computer system. We will also discuss the protection mechanism against the security threats.

2.3 PROCESSES MANAGEMENT

2.3.1 WHAT IS PROCESS

Process is the instance of the program that needs to execute. The program is the passive / static entity whereas process is the active / dynamic entity. When program is loaded in to main memory along with data it becomes, process. Before moving into the detail of process management we need to clear some terminologies like program, job, task and process.

2.3.2 TERMINOLOGY

Basically CPU needs set of instructions to be executed. This set of instructions is sometimes called program or job or task or process. When looked for operating system perspective each of these terms has specific meaning. The program, job, task and process are not the same for the operating system. Let's see the meaning of each of these terms and then see how a program is converted into process.

- **Program :**

  A program is a set of instruction in the form of modules. The tem "Program" is used from the time of batch processing system. A code written in any programming (for e.g. C language) is called program. It contains only a set of instructions.

- **Job :**

  In the batch processing system, the magnetic tapes need to be loaded and unloaded to do specific set of activities such as compiling, linking

and loading. The term “job” is used to perform the said activities required to execut the programs.

- **Task :**

The term “task” is used when a single processor executes concurrent program. It means a single user works on multiple programs simultaneously. For example while using a window operating system, the user opens media player, works on excels and check mails on browser simultaneously. It is also called multi-tasking. Figure-8 shows three tasks being performed on an OS.

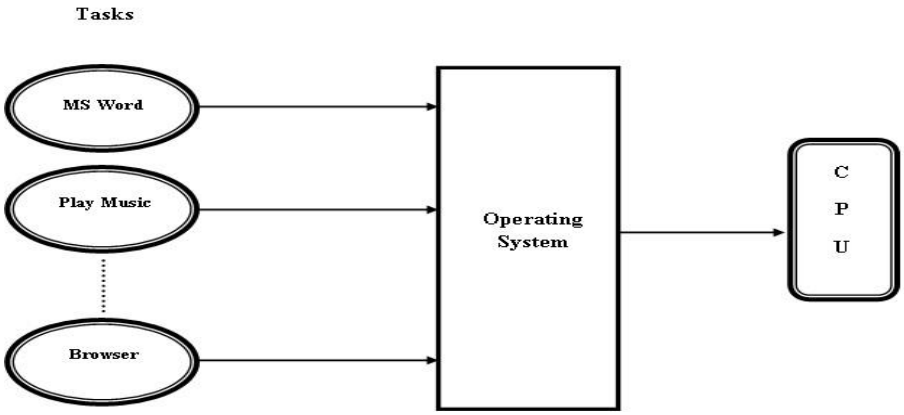

Figure-8 Multi-Tasking

- **Process :**

The Process is different from “Program” and “Job”. When a program that is ready to execute, is loaded in to memory, it is called process. A program, ready to execute, is able to compete for the resources. The resources such as CPU time, memory, I/O devices and so on.

When a process gets CPU time, it requires several other things to complete the activity, this all things make process environment. The process environment has Program counter (PC), code section, data section, CPU registers and stacks. Figure-9 shows a glimpse of process environment.

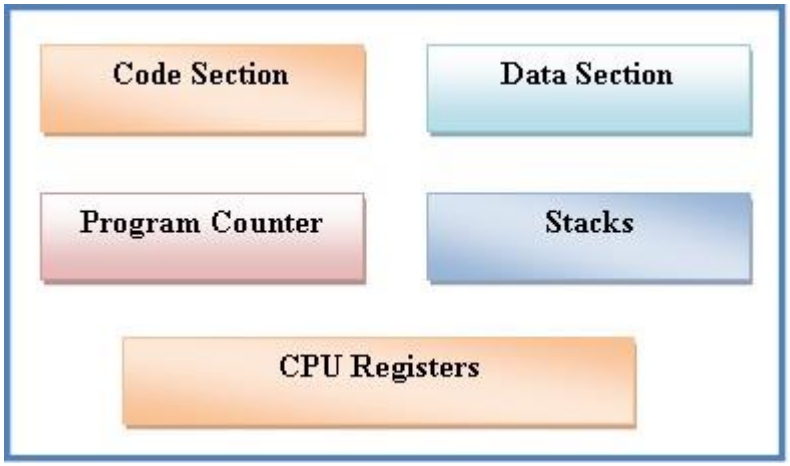

Figure-9 Process Environment

The Program counter is used for moving to next instruction when performing sequential execution. Data section stores the value of global and local variable of program. Code section stores the instruction of program along with program counter. CPU register and stacks are used to store data.

The Table-2 below shows the difference between program and process.

PROGRAM	PROCESS
It is a collection of instructions(Code Section) only	It is a collection of instruction (code section), data section, stacks, CPU register and Program Counter (PC).
It is Passive or Static entity	It is Active or Dynamic entity
It never compete of resources	It compete for resource.

Table-2 Program v/s Process

2.3.3 PROCESS SCHEDULING

After understanding the basic difference between program and process, now it's time to know how program gets converted in to process.

A program ready to execute is called job. When the job is ready to execute it is appended in the queue, call job queue. The job queue is placed in the hard disk. Then particular job is selected from the job queue and loaded in to the main memory through job scheduling. As soon as job is loaded in to the main memory it becomes a process and is stored in process queue. All the process placed in queue is called

ready queue and they all are in waiting for their turn for CPU allocation. All the processes in ready queue are competing for CPU and other system resources. The process from ready queue is selected for the CPU execution and process is transferred from the main memory to CPU. The activity of selecting next process for execution is called process scheduling. It also called CPU scheduling. The scheduling functions are performed by scheduler. Figure-10 shows how the process scheduling is done.

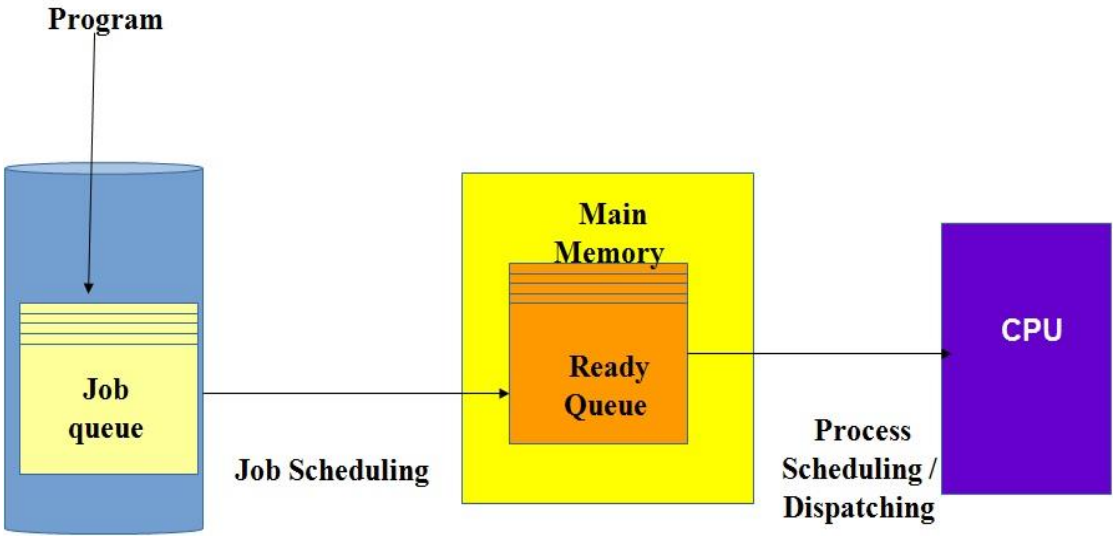

Figure-10 Process Scheduling

The scheduler use scheduling algorithm to select process from the ready queue and transfers it in to CPU. Various kind of scheduling algorithms are used by modern operating system depending upon their process management scheme. Although most widely used scheduling algorithms for the generalize operating systems are as follow:

- First Come First Served (FCFS)
- Priority Scheduling
- Shortest Process Next (SPN)
- Shortest Remaining Time Next (SRT)
- Round Robin

2.3.4 PROCESS LIFE CYCLE

As we know that the process is active / dynamic entity, it changes its status periodically. It passes through various states from its creation to destruction.

The process life cycle is divided into five states that are as follow:

- **New:**

  Whenever new job is appended into job queue, the process at that instance is in “new” state. At this time process is still on secondary memory (Hard disk).

- **Ready:**

  When job is selected from the job queue and transferred into main memory in ready queue, the process is in its “ready” state. At this point the process is ready to be executed.

- **Running:**

  Scheduler selects particular process from the ready queue and dispatches it to CPU for execution, at this time process is in its running state. The process in running state competes for the system resources. In this state CPU executes instruction of the process. The process may not remain in the CPU till it executes all instructions. Whenever time quantum gets completed, CPU takes away the running process and chooses next ready process for execution.

- **Blocked:**

  During the execution of the process, it may encounter an instruction where it needs to wait for I/O devices or some other event. In this case CPU will take away the process and CPU is given to other ready state process. Therefore a process that is waiting for the I/O device and taken away from the CPU is in its blocked state.

- **Terminated:**

  Once the process has executed all its instruction and is about to complete it in a “terminated” state.

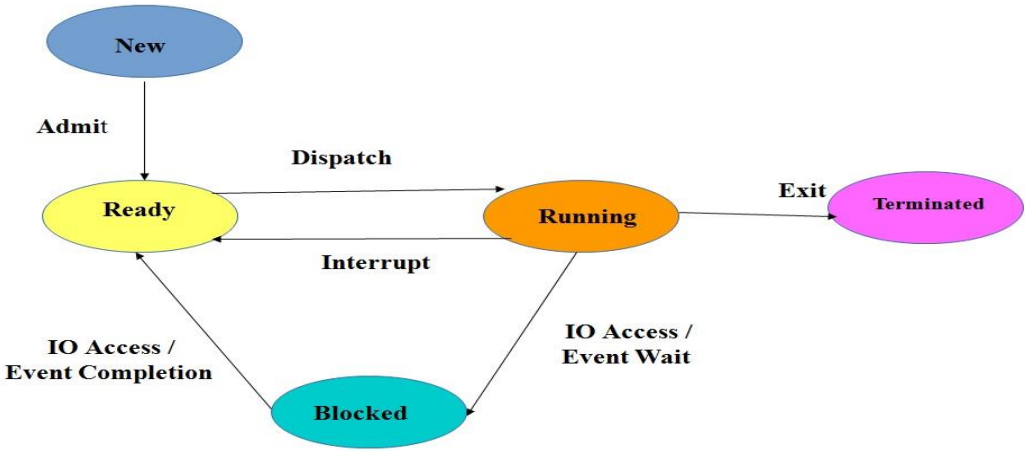

Figure-11 Process Life Cycle

Figure-11 shows the states and event execution during the process life cycle. The process changes its state whenever an event occurs. The event causes state transition. There are four events, which help in state transition of the process. The events are:

- **Admit (New State To Ready State)**

  The event takes place when process transfers from secondary memory to main memory.

- **Dispatch (Ready State To Running State)**

  This event transfers the process to the CPU for execution.

- **Interrupt (Running State To Ready State)**

  The running process may be pre-empted from the execution by some other process. It may happen due to several reasons. In such a case running process needs to transfer from running state to ready sate.

- **IO or Event Wait (Running State To Blocked State)**

  The running process reaches an instruction that needs requires it to wait for the I/O devices or event. At this time, the wait event transfers the process from the running state to blocked state.

- **IO or Event Wait Completion (Blocked State To Ready State)**

  When the blocked process's wait is over, It is assumed to have accessed the I/O device or served by event which it was waiting has been over. At

this time wait completion event transfers process from the blocked state to ready state.

- **Exit (Running State To Terminated State)**
  When running process completes execution of all instructions, the exit event changes its state from running to terminated.

2.3.5 PROCESS COTROL BLOCK (PCB)

As already discussed the process environment consists of code section, data section, CPU registers and program counter. But these are not sufficient to manage the process. The operating system requires some additional attributes to control and manage processes. All these attributes stored in a stack, known as Process Control Block (PCB). It also called Process Descriptor.

Each process has its own PCB. That means when new process is created its PCB is created and when process is terminated the PCB is terminated. The PCB is responsible to manage and control its process during the process life cycle. It stores all the relevant data which is needed by the process during the life time. Figure-12 shows the stack of Process Control Block.

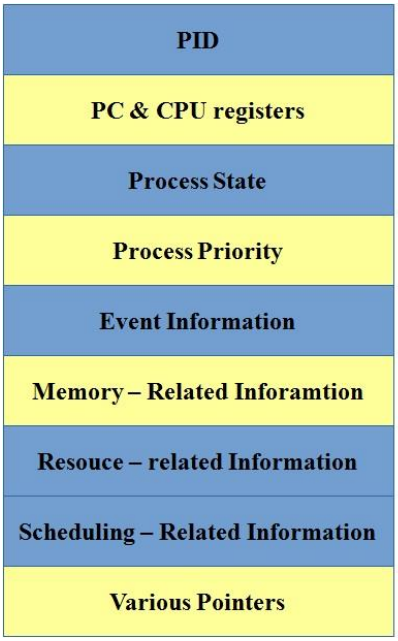

Figure-12 Process Control Block

The PCB stores the following details related to its process:

- **Process ID:** Process ID (PID) is the unique number to identify the process.
- **Program Counter:** It shows the address value at which next instruction of the process will be executed by the processor.
- **Registers:** It is used to store the data during the execution of process. Various kinds of registers are there like data registers, address registers, and control / status registers. They are used to store data during process execution.
- **State:** It indicates the current state of the process, and is used in process scheduling.
- **Priority**

  Each process has its priority number, based on it scheduler gives preference to a process. High priority process will execute first.
- **Event Information**

  Process life cycle passes from one state to other state using event. The event related information is stored here.
- **Memory Related Information**

  Memory uses various kinds of registers and tables during life cycle. The information related to memory which is used by process is mentioned here.
- **Resources Related Information**

  The process may need to access the system resources. These resources related information is listed here.
- **Scheduling Related Information**

  The process executes based on scheduling algorithm. All the scheduling related information like waiting time of process, execution time of process etc. is stored here.

- **Pointers**

  The parent process pointer, child process pointer, pointer to process's instructions and pointer to process's data are stored here.

Till this point we must have understood process creation and management, life cycle of process and process control block. These are the fundamental concepts of process management. Now we need to go in details of working of operating system.

2.4 THREADING

While learning the basic concept of process management, we should know that the CPU executes one process at a time. It means CPU is single tasking but the switching between the processes is so fast that we feel that it's multitasking. While switching between the processes the os requires storing the last values of attributes in PCB of the existing process and restoring the values of the attributes from PCB for an entering process. This entire sequence is called context switching. In the process management the context switching time is proportional to the frequency of processes arrived in scheduling. High context switching time incurs in terms of the system's performance.

In the evolution of the operating system, the need of operating systems which can respond very fast and has higher efficiency has increased. Both the requirement is fulfilled by the concept called threading. It is also called multi-threading concept.

2.4.1 WHAT IS THREAD

The thread is the path of execution within the process. A process can contain multiple threads. Thread has same properties as the process has. The idea of parallel computing is achieved by multi-threading concept. The process is divided into multiple threads. For example multiple tabs in browsers can be different threads. Thread is called light weight process.

The threads of same process can share some information of process like code section, data section and status of system resources used. As result, the time required for the context switching is less and performance of operating system

increases becomes fast. The main advantage of thread is that it minimizes the context switching time.

To manage the thread it requires the same attributes as process requires. All the required attributes are stored in the thread related structures, called Thread Control Block (TCB). The names of attributes are TID, PC, Registers, State, Priority, Event Information, Scheduling Information and TCB pointers. Figure-13 shows three threads within the process.

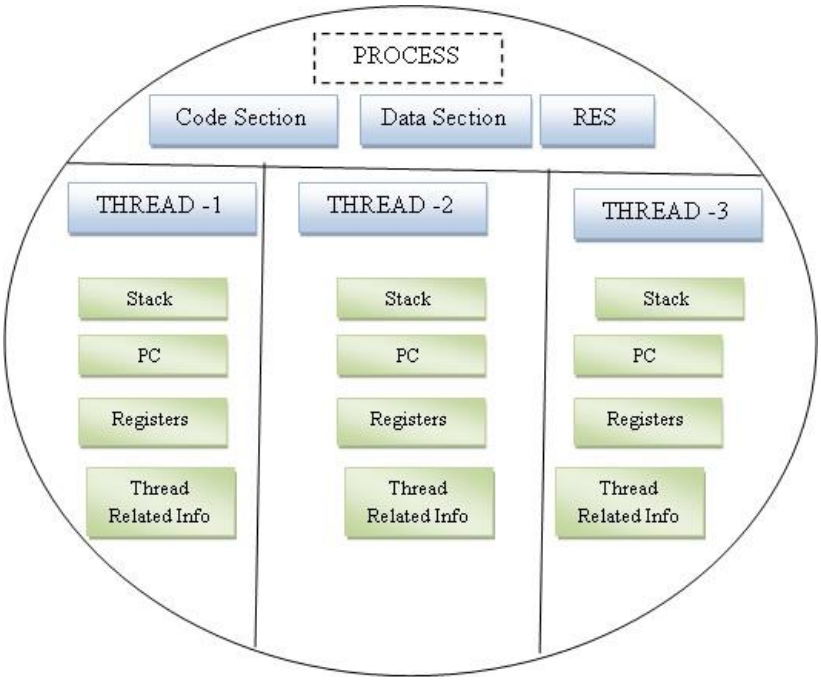

Figure-13 Process Having Three Threads

Threads are implemented in two ways:

- **User Thread**

  The user thread is managed by the application in the user space of operating system. It is created and managed by thread library which is provided by the operating system. The thread library contains code for creating and destroying threads, for passing messages and data between threads, for scheduling thread execution and for saving and restoring thread contexts.

- **Kernel Thread**

  The kernel thread is managed by the kernel of the operating system in the kernel space. Kernel threads are supported directly by the operating

system. The Kernel performs thread creation, scheduling and management in Kernel space.

2.4.2 PROCESS VS THREAD

The Table-3 shows the difference between Process and Thread.

Process	Thread
Process requires more resources than thread.	Thread requires fewer resources than process.
Process switching needs interaction with operating system.	Thread switching does not need to interact with operating system.
Multiple processes cannot share data and code section.	Threads can share data and code section.
In multiple processes each process operates independently from the others.	One thread can read, write or change another thread's data.
If process is in block state, it will not execute until the process is unblocked.	If one thread is blocked, a second thread of the same process can run.

Table-3 Process v/s Thread

2.5 MULTITASKING

The multitasking concept is somewhat different from multi-threading. We have already discussed about the multi-tasking operating system in previous unit. The multi-tasking is implemented at the process level whereas multi-threading is implemented at the thread level. Modern operating system achieves multitasking by using multithreading.

2.6 MEMORY MANAGEMENT

Multi-tasking operating system raises the issue known as memory management. Process management to work properly needs support from memory management. Each executing process needs memory. So it is essential for the operating system to keep track of memory to be allocated to process and a memory

which is free. Memory management needs to decide which process will get memory at what time and which memory will be freed by the processes.

2.6.1 STATIC AND DYNAMIC MEMORY ALLOCATION

Memory is the resource of the computer system, which needs to be divided and allocated to ready process. This allocation needs to perform in such a manager that memory and CPU can be utilized efficiently.

Memory allocation is performed by two methods:

- **Static Allocation**

  The allocation is done before the execution of a process is called static allocation. It happens when the location of the process in the memory is known at compile time, the compiler generates an absolute code for the process.

- **Dynamic Allocation**

  If the memory allocation is done during the process execution it is called dynamic allocation.

2.6.2 LOGICAL AND PHYSICAL ADDRESSES

All the modern operating system works based on dynamic memory allocation. Let us learn how dynamic memory is allocated by the operating system to running process. In the dynamic allocation, the place where memory for the process will be allocated is not known at the compile time and load time. The CPU at compile time generates some addresses, known as logical addresses. These logical addresses need to be converted into absolute addresses at the time of execution of the process. The absolute addresses are known as physical addresses. The dynamic memory management of the operating system performs the conversion of logical addresses into physical addresses. Through the dynamic memory allocation technique, operating system allocates memory to process as and when it is required at run time. It is the best way to manage memory dynamically.

2.6.3 SWAPPING

In modern operating system, due to multitasking at times no memory is available for execution of new processes. In this case the best solution is to take out

a process from the memory, and allocate the available memory to new process. This process is called swapping.

Here new issues are raised as to which process should be taken out from memory and where it will be placed? Well we can take out a process which has low priority or is blocked or stands last in queue. The process will be placed it in secondary memory (Hard disk).

The action of taking out a process from memory is called swap-out, and the process is known as a swapped – out process. The action of bringing back the swapped-out process into memory is called swap-in. the entire action is called swapping. A separate space in the hard disk, known as swap space is allocated by Os to perform the swapping activity. Swapping is the technique which is used by operating system to manage memory efficiently.

2.7 SECURITY

Security is the most important parameter for the modern operating system. In this section we will discuss about the objective of security and protection mechanism adopted by the operating system.

2.7.1 SECURITY OBJECTIVES

The basic security concepts that need to be taken care by the operating system are as mentioned:

- **Confidentiality**

  It refers to protection against disclosure of information to any unauthorized entity.

- **Authentication**

  It refers to validity of a message, it means the information received by the receiver has originated from a specific known sources.

- **Integrity**

  It refers to protection against the altered message in transmission.

- **Authorization**

  It refers to the process, through which a requester is allowed to perform operation.

- **Availability**

  It refers to the information and communication services that are available for user to 24*7.

2.7.2 SECURITY PROBLEMS

The lack of security could be extensive and can cause loss of information, corruption of data, privacy violations and so on. The main security problems are categories in three parts as under:

- **Unauthorized Disclosure**

  It is a threat to confidentiality of a system. In communication, an unauthorized node receives the sensitive data that are supposed to have been transferred between two authorized parties. This activity is known as interception.

  An unauthorized entity may infer the information, by observing the traffic in a network, or from a database, and disclose the same. It is known as inference.

- **Deception**

  It is a threat to the integrity. An unauthorized entity may pose to be an authorized one and attempts to gain access of the system. This attack is known as interception. A file or database may be altered or some data may be replaced with false data. This attack is known as falsification. A user may deny that he has sent or received data. This attack is known as repudiation.

- **Disruption**

  It is a threat to the objective of availability. For instance, a system receives so much data that the communication system is not able to handle them, thereby, interrupting the operation on the system. This attack is known as denial – of – service (DoS)

2.7.3 PROTECTION MECHANISMS

There are various kinds of resources available in the computer system. Operating system need to implement effective protection mechanisms to protect the resources. The protection mechanisms are based on two principles:

- **Need to Know Principle**

  It state that unless a user has a specific reason to access a piece of information, the permission to access is denied.

- **Principle of Least Privilege**

  It states that a process or user must be able to access information or resources that are necessary for its legitimate purpose.

Various kinds of protection models are used to implement the protection mechanisms. The techniques are Access Control Model, Access Matrix Model, Access Control Lists (ACL), Capability List (C-lists) and so on.

2.8 USER MANAGEMENT

The operating system needs to determine the identity of the user at the time of logging. This process is called authentication. An authentication has two steps:

- **Identification**

  It refers to a unique identifier allocated to the user for authentication.

- **Verification**

  Here, the verification of user is performed, that is, conformation which binds the user and the identifier.

Operating system authenticates the user when they login which is based on either password, PIN, biometrics identity, token, smartcard, OTP, key or patterns.

2.9 FILE MANAGEMENT

Operating system uses File System to manage the files in the computer system. File system is the convenient way to store and retrieve the data. Files are used to

store related information and are mapped to the hard disks or other storage media by the operating system.

The information stored in a file is in bits, bytes, lines or records. File system has two views: Logical View and System View. The user only sees the logical view of the file. The work of system views is to map the logical file to the secondary storage.

The following are the primary elements of a file system:

- **File Management:** It manages how the files are stored, referenced, shared and secured.
- **File Allocation:** It provides the methods to allocate files on the disk space.
- **File Access Methods:** It provides the methods to access stored files.

The Files can be accessed by the any of the following technique:

- **Sequential File Access:** By this technique file is accessed in the sequential manner, it means it can access in the same order as it is stored.
- **Indexed Sequential File Access:** It is a combination of sequential and random access mechanism. The key field is maintained in sequential manner, based on key filed the pointer of main file moves to read write operations.
- **Indexed File Access:** In this type of access it is not necessary to store data in a file sequentially and to search the record by the key field. Here multiple indexes are there for each attribute, by which the user can, search a record in the main file.
- **Direct File Access:** Direct access is meant for a random structure of secondary storage.

2.10 LET US SUM UP

In this unit we learnt about the concepts of process management, Threads, Multitasking, Memory Management, Security, User Management and File management.

Let’s quickly review the main points of the unit.

- Process is the instance of the program that is different from program.
- Process is dynamic entity and program is static entity.
- Process has environment that consist of PC, Code Section, Data Section, and CPU Registers.
- Process scheduler selects process for the next execution.
- Process life cycle shows the states of process and events that passes through.
- Process related data is stored in Process Control Block (PCB)
- Thread is the path of execution within the process. A process can contain multiple threads.
- Thread related data is stored in Thread Control Block (TCB)
- Memory can be allocated in two ways: static allocation and dynamic allocation.
- Each modern operating system need to focus on security objectives and its problems.
- The modern operating systems are also responsible to manage users and files.

2.11 CHECK YOUR PROGRESS

Give the answer of the following MCQ.

1. Process is _____ of the program.
2. A _____ is a set of instructions in the form of modules.
3. Compiling, linking and loading activities are done by _____.
4. PC stand for _____.
5. Program is _____ entity.
6. Process is _____ entity.
7. When job is loaded into the main memory, it becomes _____.
8. A process selected for next execution is called _____.
9. FCFS stands for _____.
10. SPN stand for _____.
11. A process is transferred from New state to Ready state by _____ event.

12. A process is transferred from Running state to Ready state by _____event.
13. PCB stands for _____.
14. TCB stands for_____.
15. The memory allocation done before the execution of a process is called _____ allocation.
16. _____ refer to protection against disclosure of information to any unauthorized entity.
17. The ACL stands for _____.
18. _____ refers to protection against the altered message in transmission.

2.12 CHECK YOUR PROGRESS: POSSIBLE ANSWERS

1.	Instance	2.	Program	3.	Job
4.	Program Counter	5.	Static	6.	Dynamic
7.	Process	8.	Process Scheduling	9.	First Come First Server
10.	Shortest Process Next	11.	Admit	12.	Interrupt
13	Process Control Block	14.	Thread Control Block	15.	Static
16.	Confidentiality	17.	Access Control List	18.	Integrity

2.13 FURTHER READING

- Naresh Chauhan (2014), Principals of Operating System, Oxford.
- “Introduction to OS”, https://nptel.ac.in/courses/106106144/2.
- “Process and Thread Management”, https://www.youtube.com/watch?v=OrM7nZcxXZU.
- “memory management in operating system”, https://www.youtube.com/watch?v=W0068fRJTGQ.

2.14 ASSIGNMENTS

Write answers of the following Questions.

1. What is process? How is process created?
2. Explain Process life cycle with diagram.
3. Write a short note on Process control block.
4. What is Threading? Explain how it is different from process.
5. How memory management is done the operating system.
6. List the security objectives need to be taken care by operating system.