Memory Management

Memory management refers to management of Primary Memory or Main Memory. Main memory is a large array of words or bytes where each word or byte has its own address.

Main memory provides a fast storage that can be accessed directly by the CPU. For a program to be executed, it must in the main memory. An Operating System does the following activities for memory management –

Keeps tracks of primary memory, i.e., what part of it are in use by whom, what part are not in use.

In multiprogramming, the OS decides which process will get memory when and how much.

Allocates the memory when a process requests it to do so.

De-allocates the memory when a process no longer needs it or has been terminated.

Processor Management

In multiprogramming environment, the OS decides which process gets the processor when and for how much time. This function is called process scheduling. An Operating System does the following activities for processor management –

Keeps tracks of processor and status of process. The program responsible for this task is known as traffic controller.

Allocates the processor (CPU) to a process.

De-allocates processor when a process is no longer required.

Device Management

An Operating System manages device communication via their respective drivers. It does the following activities for device management –

Keeps tracks of all devices. Program responsible for this task is known as the I/O controller.

Decides which process gets the device when and for how much time.

Allocates the device in the efficient way.

De-allocates devices.

File Management

A file system is normally organized into directories for easy navigation and usage. These directories may contain files and other directions.

An Operating System does the following activities for file management –

Keeps track of information, location, uses, status etc. The collective facilities are often known as file system.

Decides who gets the resources.

Allocates the resources.

A program is a piece of code which may be a single line or millions of lines. A computer program is usually written by a computer programmer in a programming language. For example, here is a simple program written in C programming language −

```
#include <stdio.h>

int main() {
   printf("Hello, World! \n");
   return 0;
}
```

A computer program is a collection of instructions that performs a specific task when executed by a computer. When we compare a program with a process, we can conclude that a process is a dynamic instance of a computer program.

A part of a computer program that performs a well-defined task is known as an algorithm. A collection of computer programs, libraries and related data are referred to as a software.

Process Life Cycle

When a process executes, it passes through different states. These stages may differ in different operating systems, and the names of these states are also not standardized.

In general, a process can have one of the following five states at a time.

S.N. State & Description

1

Start

This is the initial state when a process is first started/created.

2

Ready

The process is waiting to be assigned to a processor. Ready processes are waiting to have the processor allocated to them by the operating system so that they can run. Process may come into this state after Start state or while running it by but interrupted by the scheduler to assign CPU to some other process.

3

Running

Once the process has been assigned to a processor by the OS scheduler, the process state is set to running and the processor executes its instructions.

4
Waiting

Process moves into the waiting state if it needs to wait for a resource, such as waiting for user input, or waiting for a file to become available.

5
Terminated or Exit

Once the process finishes its execution, or it is terminated by the operating system, it is moved to the terminated state where it waits to be removed from main memory.

Process States
Process Control Block (PCB)
A Process Control Block is a data structure maintained by the Operating System for every process. The PCB is identified by an integer process ID (PID). A PCB keeps all the information needed to keep track of a process as listed below in the table –

S.N. Information & Description
1
Process State

The current state of the process i.e., whether it is ready, running, waiting, or whatever.

2
Process privileges

This is required to allow/disallow access to system resources.

3
Process ID

Unique identification for each of the process in the operating system.

4
Pointer

A pointer to parent process.

5
Program Counter

Program Counter is a pointer to the address of the next instruction to be executed for this process.

6
CPU registers

Various CPU registers where process need to be stored for execution for running state.

7
CPU Scheduling Information

Process priority and other scheduling information which is required to schedule the process.

8
Memory management information

This includes the information of page table, memory limits, Segment table depending on memory used by the operating system.

9
Accounting information

This includes the amount of CPU used for process execution, time limits, execution ID etc.

10
IO status information

This includes a list of I/O devices allocated to the process.