## 

**DEPARTMENT OF COMPUTER SCIENCE & TECHNOLOGY**

**Subject Name:** Operating System  **Subject Code: CSH206B-T**

**Topic:** Introduction to Operating System

**Tutorial: 2**

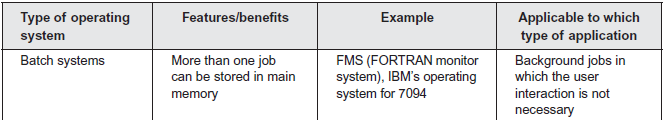
**Objective: To gain familiarity with introductory concepts of OS**

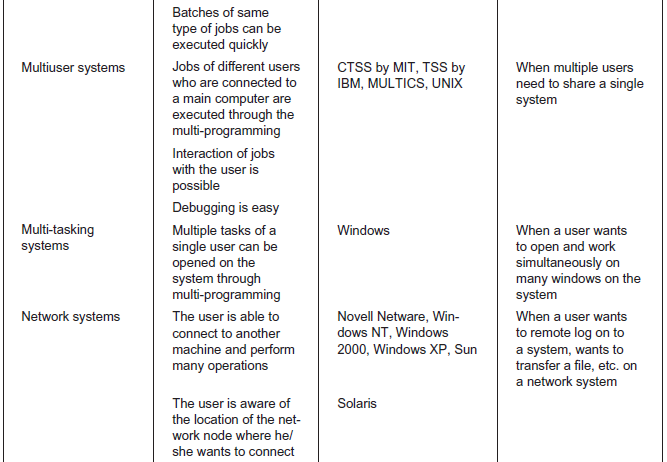
**Course Outcome : C01 : Learn architecture of OS**

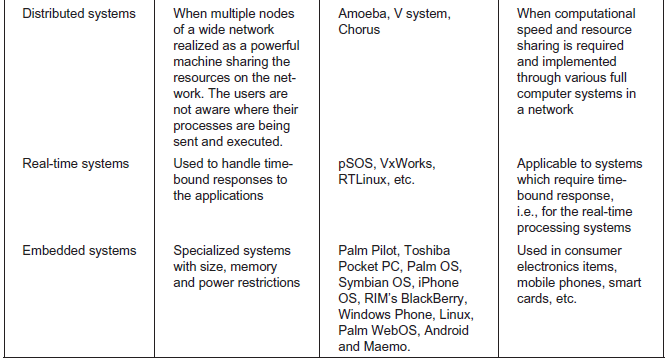
**Bloom’s Taxonomy : BT1 : Knowledge**

Q1. Based on the features of major types of Operating System, Complete the Following: -

|  |  |  |  |
| --- | --- | --- | --- |
| Type of OS | Features/Benefits | Example | Applicable to which type of application |







Q2. Examine the following conditions and find appropriate operating system for them.

1. In a LAN, users want to share some costly resources like laser printers: N/w OS
2. Multiple users on a system want quick response on their terminals: Time Sharing
3. Railway reservation system: Real Time
4. Washing machine: Embedded

Q3. Differentiate between the following: -

1. Network and Distributed Operating System.

|  |  |  |
| --- | --- | --- |
| **S.NO** | **NETWORK OPERATING SYSTEM** | **DISTRIBUTED OPERATING SYSTEM** |
| 1. | Network Operating System’s main objective is to provide the local services to remote client. | Distributed Operating System’s main objective is to manage the hardware resources. |
| 2. | In Network Operating System, Communication takes place on the basis of files. | In Distributed Operating System, Communication takes place on the basis of messages and share memory. |
| 3. | Network Operating System is more scalable than Distributed Operating System. | Distributed Operating System is less scalable than Network Operating System. |
| 4. | In Network Operating System, fault tolerance is less. | While in Distributed Operating System, fault tolerance is high. |
| 5. | Rate of autonomy in Network Operating System is high. | While The rate of autonomy in Distributed Operating System is less. |
| 6. | Ease of implementation in Network Operating System is also high. | While in Distributed Operating System Ease of implementation is less. |
| 7. | In Network Operating System, All nodes can have different operating system. | While in Distributed Operating System, All nodes have same operating system. |

1. System Program and Application Programs.

|  |  |  |
| --- | --- | --- |
| **S.NO** | **SYSTEM SOFTWARE** | **APPLICATION SOFTWARE** |
| 1. | System Software maintain the system resources and give the path for application software to run. | Application software is built for specific tasks. |
| 2. | Low level languages are used to write the system software. | While high level languages are used to write the application software. |
| 3. | Its a general purpose software. | While its a specific purpose software. |
| 4. | Without system software, system can’t run. | While without application software system always runs. |
| 5. | System software runs when system is turned on and stop when system is turned off. | While application software runs as per the user’s request. |
| 6. | Example of system software are operating system, etc. | Example of application software are Photoshop, VLC player etc. |

1. Multiprogramming, Multiuser and Multitasking OSs.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Characteristic** | **Multiprogramming** | **Multiprocessing** | **Multithreading** | **Multitasking** |
| 1 | What it is: | The concurrent residency of more than one program in the main memory is called as multiprogramming. | The availability of more than one processor per system, which can execute several set of instructions in parallel is called as multiprocessing. | A process is divided into several different sub-processes called as threads, which has its own path of execution. This concept is called as multithreading. | The execution of more than one task simultaneously is called as multitasking. |
| 2 | Number of CPU: | One | More than one | Can be one or more than one | One |
| 3 | Job processing time: | More time is taken to process the jobs. | Less time is taken for job processing. | Moderate amount of time is taken for job processing. | Moderate amount of time. |
| 4 | Number of process being executed: | One process is executed at a time. | More than one process can be executed at a time | Various components of the same process are being executed at a time. | One by one job is being executed at a time. |
| 5 | Economical: | It is economical. | Is less economical. | Is economical. | It is economical. |
| 6 | Number of users: | One at a time. | Can be one or more than one. | Usually one. | More than one. |
| 7 | Throughput: | Throughput is less. | Throughput is maximum. | Moderate. | Throughput is moderate. |
| 8 | Efficiency: | Less | Maximum | Moderate | Moderate |
| 9 | Categories: | No further divisions | Symmetric & Asymmetric. | No further divisions. | Single User & Multiuser. |

1. Real Time and Embedded OSs.

Embedded system describes a system that contains one or more software programmable devices but which is not itself a general purpose computer. Such a system typically has a fixed, single application rather than end-user selected and loaded software (which would make it general purpose).

However "embedded" covers a wide spectrum of systems and is not always easy to define; for example if you were writing the UMTS code for a smartphone, you might reasonably be regarded as an embedded developer, if you were writing Flappy Angry Birds 2.0 for that same phone however, you would not - so a smartphone may be both an embedded system and general purpose computer - depending on your view point. Similarly a hand-held games console's system software is embedded; the games themselves are not I would say.

A real-time system describes a system with deterministic low latency response to input events. An embedded system may be "real-time, or it might not. I would normally use the term "real-time embedded system" to be clear.