USN | | | | | | | | | | |

Internal Assessment Test 1 – May. 2022

| Sub: | Operating System | | | | | Sub Code: | 18 EC 641 | Branch: | ECE | |
|---|---|---|---|---|---|---|---|---|---|---|
| Date: | 10-05-22 | Duration: | 90 min's | Max Marks: | 50 | Sem / Sec: | 6 – A B C D | | OBE | |

| | Answer any FIVE FULL Questions | MARKS | CO | RBT |
|---|---|---|---|---|
| 1. | Define the operating system. What are the goals of an operating system? Explain? | [10] | CO1 | L1 |
| 2. | Explain the time-sharing system with an example. | [10] | CO1 | L2 |
| 3. | With a neat diagram, explain the turnaround time of the batch processing system. | [10] | CO1 | L2 |
| 4a. | Explain the features of the real-time operating system. | [06] | CO1 | L1 |
| 4b. | Explain the measures of Efficiency, System Performance, and User Service | [04] | CO1 | L1 |
| 5. | Explain briefly, the different classes of an OS with primary concerns and key concepts. | [10] | CO1 | L2 |
| 6. | Explain the multiprogramming operating system with an example. | [10] | CO1 | L2 |
| 7. | Explain OS view of a process | [10] | CO2 | L1 |

USN | | | | | | | | | | |

Internal Assessment Test 1 – May. 2022

| Sub: | Operating System | | | | | Sub Code: | 18 EC 641 | Branch: | ECE | |
|---|---|---|---|---|---|---|---|---|---|---|
| Date: | 10-05-22 | Duration: | 90 min's | Max Marks: | 50 | Sem / Sec: | 6 – A B C D | | OBE | |

| | Answer any FIVE FULL Questions | MARKS | CO | RBT |
|---|---|---|---|---|
| 1. | Define the operating system. What are the goals of an operating system? Explain? | [10] | CO1 | L1 |
| 2. | Explain the time-sharing system with an example. | [10] | CO1 | L2 |
| 3. | With a neat diagram, explain the turnaround time of the batch processing system. | [10] | CO1 | L2 |
| 4a. | Explain the features of the real-time operating system. | [06] | CO1 | L1 |
| 4b. | Explain the measures of Efficiency, System Performance, and User Service | [04] | CO1 | L1 |
| 5. | Explain briefly, the different classes of an OS with primary concerns and key concepts. | [10] | CO1 | L2 |
| 6. | Explain the multiprogramming operating system with an example. | [10] | CO1 | L2 |
| 7. | Explain OS view of a process | [10] | CO2 | L1 |

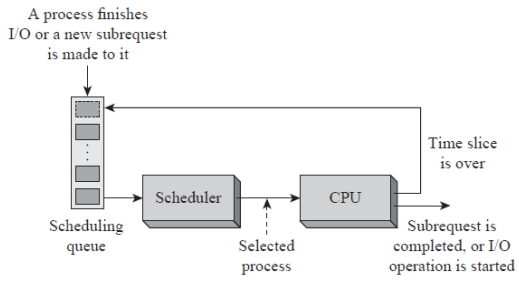
CMR
INSTITUTE OF
TECHNOLOGY

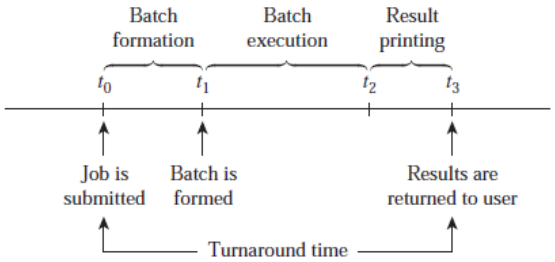
Internal Assesment Test – I MAY 2022

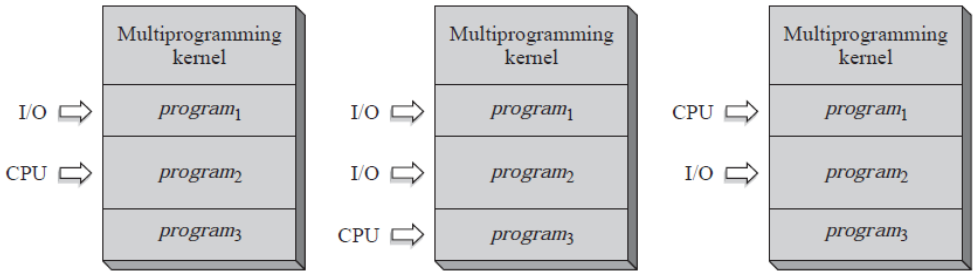
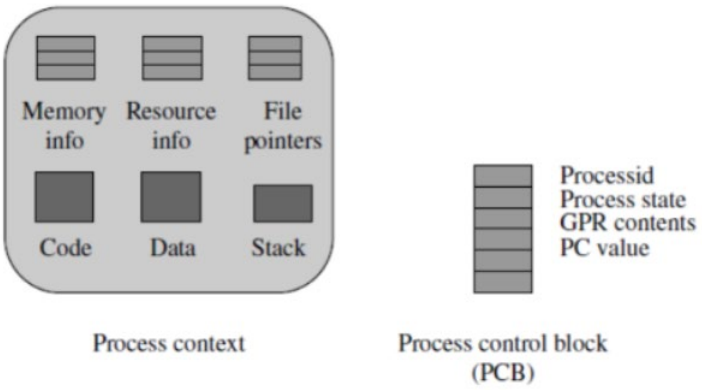
| Sub: | Operating Systems | | | | | | | Code: | 18EC641 |
|---|---|---|---|---|---|---|---|---|---|
| Date: | 10/05/2022 | Duration: | 90 mins | Max Marks: | 50 | Sem: | 6 | Branch: | ECE |

**Answer any five questions out of seven questions**

## Scheme of solutions

| Q. no. | Questions | Marks |
|---|---|---|
| 1. | What is an operating system ?<br>• An operating system (OS) is a collection of programs that achieve effective utilization of a computer system by providing<br>  • Convenient methods of using a computer<br>  • Saves users' time and boosts their productivity<br>  • Efficient use of the computer<br>• An OS has several kinds of users<br>  • The OS meets diverse requirements of different kinds of users<br>  • Each user has a different view of what an OS is, and what it does. Each of these views is called an abstract view | 2M |
| | • Two primary goals of an OS are<br>  • Efficient use of the computer's resources<br>  • To ensure cost-effectiveness of the computer<br>  • User convenience<br>  • A user should find it easy to use the computer<br>• These two goals sometimes conflict<br>  • Prompt service can be provided through exclusive use of a computer; however, efficient use requires sharing of a computer's resources among many users<br>  • An OS designer decides which of the two goals is more important under what conditions<br>  • That is why we have so many operating systems! | 8M |
| 2. | The scheduling technique used by a time-sharing kernel is called *round-robin scheduling with time-slicing*. It works as follows (see Figure 3.6): The kernel maintains a *scheduling queue* of processes that wish to use the CPU; it always schedules the process at the head of the queue. When a scheduled process completes servicing of a subrequest, or starts an I/O operation, the kernel removes it from the queue and schedules another process. Such a process would be added at the end of the queue when it receives a new subrequest, or when its I/O operation | 6M |
| |  <br>$rt = n \times (\delta + \sigma)$<br>$\eta = \frac{\delta}{\delta + \sigma}$ | 4M |

| | | |
|---|---|---|
| 3. |  | 4M |
| | A *batch* is a *sequence* of user jobs formed for processing by the operating system. A computer operator formed a batch by arranging a few user jobs in a sequence and inserting special marker cards to indicate the start and end of the batch. When the operator gave a command to initiate processing of a batch, the *batching kernel* set up the processing of the first job of the batch. At the end of the job, it initiated execution of the next job, and so on, until the end of the batch. Thus the operator had to intervene only at the start and end of a batch. | 6M |
| 4a. | **Essential Features of a Real-Time Operating System**<br>**Feature** / **Explanation**<br>Concurrency within an application: A programmer can indicate that some parts of an application should be executed concurrently with one another. The OS considers execution of each such part as a process.<br>Process priorities: A programmer can assign priorities to processes.<br>Scheduling: The OS uses priority-based or deadline-aware scheduling.<br>Domain-specific events, interrupts: A programmer can define special situations within the external system as events, associate interrupts with them, and specify event handling actions for them.<br>Predictability: Policies and overhead of the OS should be predictable.<br>Reliability: The OS ensures that an application can continue to function even when faults occur in the computer. | 6M |
| 4b. | **Measures of Efficiency, System Performance, and User Service**<br>**Aspect** / **Measure** / **Description**<br>Efficiency of use / CPU efficiency / Percent utilization of the CPU<br>Memory efficiency / Percent utilization of memory<br>System performance / Throughput / Amount of work done per unit time<br>User service / Turnaround time / Time to complete a job or a process<br>Response time / Time to implement one subrequest | 4M |
| 5. | **Key Features of Classes of Operating Systems**<br>**OS class** / **Period** / **Prime concern** / **Key concepts**<br>Batch processing / 1960s / CPU idle time / Automate transition between jobs<br>Multiprogramming / 1960s / Resource utilization / Program priorities, preemption<br>Time-sharing / 1970s / Good response time / Time slice, round-robin scheduling<br>Real time / 1980s / Meeting time constraints / Real-time scheduling<br>Distributed / 1990s / Resource sharing / Distributed control, transparency | 5x2=10M |

| | | |
|---|---|---|
| 6. |  | 4M |
| | **Feature** / **Description**<br>DMA — The CPU initiates an I/O operation when an I/O instruction is executed. The DMA implements the data transfer involved in the I/O operation without involving the CPU and raises an I/O interrupt when the data transfer completes.<br>Memory protection — A program can access only the part of memory defined by contents of the *base register* and *size register*.<br>Kernel and user modes of CPU — Certain instructions, called *privileged instructions*, can be performed only when the CPU is in the kernel mode. A program interrupt is raised if a program tries to execute a privileged instruction when the CPU is in the user mode. | 6M |
| 7. | **OS View of a process:**<br>  | 4M |
| | **Process Context** The process context consists of the following:<br>1. *Address space of the process:* The code, data, and stack components of the process (see Definition 5.1).<br>2. *Memory allocation information:* Information concerning memory areas allocated to a process. This information is used by the memory management unit (MMU) during operation of the process (see Section 2.2.2).<br>3. *Status of file processing activities:* Information about files being used, such as current positions in the files.<br>4. *Process interaction information:* Information necessary to control interaction of the process with other processes, e.g., ids of parent and child processes, and interprocess messages sent to it that have not yet been delivered to it.<br>5. *Resource information:* Information concerning resources allocated to the process.<br>6. *Miscellaneous information:* Miscellaneous information needed for operation of a process. | 6M |