CONTENTS

RCS 401 : OPERATING SYSTEMS

1 UNIT

Computer System

PART-1

Introduction : Operating System and Function, Classification of Operating System : Batch, Interactive, Time-Sharing, Real Time System, Multi-processor Systems, Multi-user Systems, Multi-process Systems, Multi-threaded System.

CONCEPT OUTLINE : PART-1

- An operating system is a program that acts an intermediary between a user of a computer and the computer hardware.
- **Function of operating system :**
  i. Memory management
  ii. Processor management
  iii. Device management
  iv. File management
- The feature of a batch system is the lack of interaction between the user and the job while that job is executing.
- A multi-processor is a computer system with two or more central processing units, with each one sharing the common main memory as well as peripherals.

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 1.1. What is operating system ? Write the major functions of an operating system.

Answer

1. An operating system acts as an intermediary between the user of a computer and computer hardware.
2. An operating system is software that manages the computer hardware.
3. The hardware must provide appropriate mechanisms to ensure the correct operation of the computer system and to prevent user programs from interfering with the proper operation of the system.

Functions of an operating system :

The operating systems perform the following functions :

1. **Booting**
   i. Uses diagnostic routines to test system for equipment failure.
   ii. Copies BIOS (Basic Input System) programs from ROM chips to main memory.
   iii. Loads operating system into computer's main memory.
2. **Formatting :**
   Formats (initializes) diskettes so they can store data and programs.
3. **Managing computer resources :**
   i. Keeps track of locations in main memory where programs and data are stored (memory management).
   ii. Moves data and programs back and forth between main memory and secondary storage (swapping) via partitioning, using foreground and background areas, and using buffers and queues.
4. **Managing files :**
   i. Copies files/programs from one disk to another.
   ii. Backup files/programs.
   iii. Erases (deletes) files/programs.
   iv. Renames files.
5. **Managing tasks :**
   May be able to perform multi-tasking, multi-programming, time-sharing or multi-processing.

Que 1.2. **Describe the classification of operating system.**

OR

Write down the different types of operating system.

AKTU 2016-17, Marks 10

Answer

Various types of operating system are as follows :

1. ***Batch system :***
   a. This type of operating system was used in the earlier age.
   b. To speedup processing, jobs with similar needs were batched together and were run through the computer as a group.
   c. The definitive feature of a batch system is the lack of interaction between the user and the job while that job is executing.
   d. In this execution environment, the CPU is often idle.

Multi-programming system :

a. In this type of operating system, more than one program will reside into main memory.
b. The operating system picks and begins to execute one of the jobs in the memory.

c. Eventually, the job may have to wait for some task, the operating system simply switches to another job and executes it.
d. When the first job finishes, waiting job gets the CPU back.
e. As long as there is always some job to execute, the CPU will never be idle.

3. **Time-sharing system :**
   a. A time-shared operating system allows the many users to share the computer simultaneously.
   b. A time-shared operating system uses CPU scheduling and multi-programming to provide each user with a small portion of a time-shared computer.
4. **Real time operating system :**
   a. Real time operating system is a special purpose operating system, used when there are rigid time requirements on the operation of a processor or the flow of data.

Que 1.3. **What is real time operating system ? What is the difference between hard real time and soft real time operating system ?**

AKTU 2013-14, Marks 05

Answer

1. Real time operating system is a special purpose operating system, used when there are rigid time requirements on the operation of a processer or the flow of data.
2. Real time operating system has well-defined, fixed time constraints otherwise system will fail.
3. For example, scientific experiments, medical imaging systems, industrial control systems, weapon systems, robots, home-appliances controllers, air traffic control system etc.

Difference between hard real time and soft real time operating system :

S. No.	Characteristic	Hard real time	Soft real time
1.	Response time	Hard required	Soft desired
2.	Peak-load performance	Predictable	Degraded
3.	Control of pace	Environment	Computer
4.	Safety	Often critical	Non-critical
5.	Size of data files	Small/medium	Large
6.	Redundancy type	Active	Checkpoint-recovery
7.	Data integrity	Short-term	Long-term
8.	Error detection	Autonomous	User assisted

Que 1.4. **Discuss essential properties of time-sharing operating system, real time operating system and distributed operating system.**

AKTU 2012-13, Marks 05

Answer

Properties of time-sharing operating system :

1. Uses CPU scheduling and multi-programming to provide economical interactive use of a system.
2. The CPU switches rapidly from one user to another *i.e.*, the CPU is shared between a number of interactive users.
3. Instead of having a job defined by spooled card images, each program reads its next control instructions from the terminal and output is normally printed immediately on the screen.

Properties of real time operating system :

1. Real time systems are usually dedicated, embedded systems.
2. They typically read from and react to sensor data.
3. The system must guarantee response to events within fixed periods of time to ensure correct performance.

Properties of distributed operating system :

1. Distributes computation among several physical processors.
2. The processors do not share memory or a clock, instead, each processor has its own local memory.
3. They communicate with each other through various communication lines.

Que 1.5. **Explain the following terms :**

i. **Multi-programming system**
ii. **Multi-processor system**

Answer

i. **Multi-programming system :** Refer Q. 1.2, Page 1–3B, Unit-1.

ii. **Multi-processor system :**

Multi-processor systems have more than one processor in close communication. They share the computer bus, system and input-output devices and sometimes memory.

In multi-processing system, it is possible for two processes to run in parallel.

Multi-processor systems are of two types : symmetric multi-processing and asymmetric multi-processing.

a. In symmetric multi-processing, each processor runs identical copy of the operating system and they communicate with one another as needed. All the CPU shared the common memory.

b. In asymmetric multi-processing, each processor is assigned a specific task. It uses master-slave relationship.

Que 1.6. **Explain the following terms clearly :**

i. **Multi-programming**

ii. **Multi-threading**

AKTU 2011-12, Marks 05

Answer

i. **Multi-programming :** Refer Q. 1.2, Page 1–3B, Unit-1.

ii. **Multi-threading :**

1. Multi-threading extends the idea of multi-tasking into applications, so we can sub-divide specific operations within a single application into individual threads.
2. Each of the threads can run in parallel.
3. The OS divides processing time not only among different applications, but also among each thread within an application.
4. In a multi-threaded program, an example application might be divided into four threads : a user interface thread, a data acquisition thread, network communication, and a logging thread.
5. We can prioritize each of these, so that they operate independently.
6. Thus, in multi-threaded applications, multiple tasks can progress in parallel with other applications that are running on the system.

Que 1.7. **Write down the difference between multi-processing and multi-programming operating system.**

AKTU 2013-14, Marks 05

Answer

S. No.	Multi-processing	Multi-programming
1.	Multi-processing refers to processing of multiple processes at same time by multiple CPUs.	Multi-programming keeps several programs in main memory at the same time and execute them concurrently utilizing single CPU
2.	It utilizes multiple CPUs.	It utilizes single CPU.
3.	It permits parallel processing.	Context switching takes place.

4.	Less time taken to process the jobs.	More time taken to process the jobs.
5.	It facilitates much efficient utilization of devices of the computer system.	Less efficient than multi-processing.
6.	Usually more expensive.	Such systems are less expensive.

Que 1.8. **Describe the differences between symmetric and asymmetric multi-processing.** **AKTU 2014-15, Marks 05**

Answer

S.No.	Basis	Symmetric multi-processing	Asymmetric multi-processing
1.	Architecture	All processor in symmetric multi-processing has the same architecture.	All processor in asymmetric multi-processing may have same or different architecture.
2.	Communication	All processors communicate with another processor by a shared memory.	Processors need not to communicate as they are controlled by the master processor.
3.	Failure	If a processor fails, the computing capacity of the system reduces.	If a master processor fails, a slave is turned to the master processor to continue the execution. If a slave processor fails, its task is switched to other processors.
4.	Ease	Symmetric multi-processor is complex.	Asymmetric multi-processor is simple.

Que 1.9. **What is spooling ?**

OR

What is spooling ? What are the advantage of spooling over buffering ?

Answer

1. SPOOLING is acronym for Simultaneous Peripheral Operations Online.
2. Spooling refers to putting jobs in a buffer, a special area in memory or on a disk where a device can access them when it is ready.

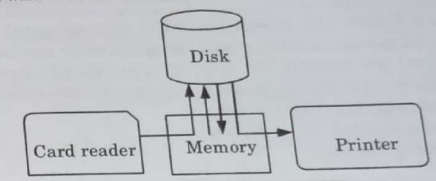


Fig. 1.9.1. Print spooling.

3. Spooling is useful because device access data at different rates.
4. The buffer provides a waiting station where data can rest while the slower device catches up.
5. The most common spooling application is print spooling.
6. In print spooling, documents are loaded into a buffer and then the printer pulls them off the buffer at its own rate.
7. Spooling is also used for processing data at remote sites. The CPU sends the data via communications path to a remote printer.

Advantages of spooling over buffering :

1. The spooling operation uses a disk as a very large buffer.
2. Spooling is capable of overlapping I/O operation for one job with processor operations for another job.

Que 1.10. **Differentiate between batch processing system and multi-programming system with example.**

Answer

S. No.	Batch processing system	Multi-programming system
1.	In batch systems, the instructions, data and some control information are submitted to the computer operator in the form of a job.	Multi-programming system allow concurrent execution of multiple programs, and hence multi-programming operating system require more sophisticated scheduling algorithms.

2.	Only one program is in execution at a time, any time-critical device management is not required, which simplifies the I/O management.	Multi-programming operating system allow sharing of I/O devices among multiple users, more sophisticated I/O management is required.
3.	Since files are also accessed in a serial manner, a concurrency control mechanism for file access is not required, making file management also a very simple matter in a batch operating system.	File management in multi-programming operating system must provide advanced protection, and concurrency control methods.
4.	The users are not allowed to interact with the computer system.	The programs should be scheduled in such a way that the CPU remains busy for maximum amount of time.
5.	**Example :** Execution of batch files such as autoexec.bat by the computer.	**Example :** Interleaved execution of two or more different and independent programs by the same computer.

PART-2

Operating System Structure, Layered Structure, System Components, Operating System, Services Re-entrant Kernels, Monolithic and Micro-kernel System.

CONCEPT OUTLINE : PART-2

- **Structure of OS :**
  - i. Simple structure
  - ii. Layered approach
  - iii. Micro-kernels
  - iv. Modules
- **System components :**
  - i. Memory management
  - ii. Processor management
  - iii. Device management
  - iv. File management

- **Operating system service :**
  - i. Program execution
  - ii. I/O operation
  - iii. File system manipulation
  - iv. Error detection
  - v. Protection and security

Questions-Answers

Long Answer Type and Medium Answer Type Questions

Que 1.11. **What is an operating system ? Discuss the operating system structure.**

OR

Explain layered structure of an operating system. Also explain advantages and disadvantages of the layered approach to system design. **AKTU 2014-15, Marks 05**

Answer

Operating system : Refer Q. 1.1, Page 1–2B, Unit-1.

Structure of OS :

1. **Simple structure :** MS-DOS written to provide the most functionality in the least space :
   a. Not divided into modules.
   b. Although MS-DOS has some structure, its interfaces and levels of functionality are not well separated.

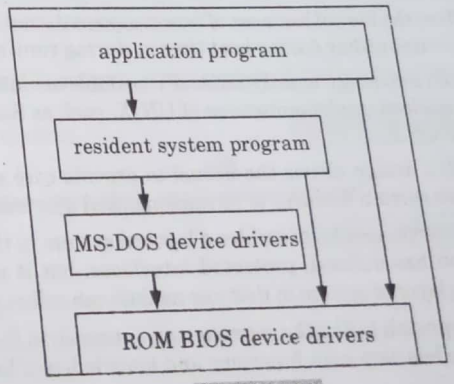


Fig. 1.11.1.

2. **Layered approach :**
   a. The operating system is divided into a number of layers (levels), each built on top of lower layers. The bottom layer (layer 0), is the hardware; the highest (layer *N*) is the user interface.
   b. With modularity, layers are selected such that each uses functions (operations) and services of only lower-level layers.

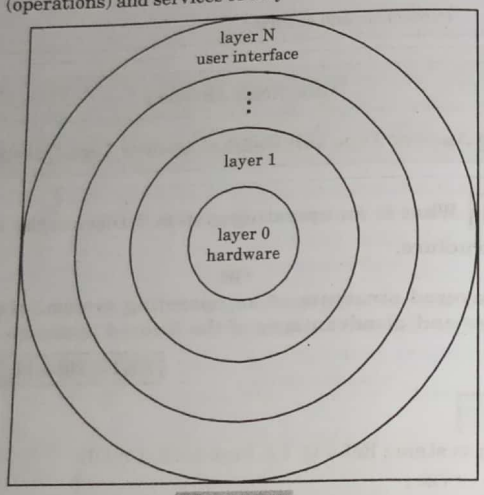


Fig. 1.11.2.

3. **Micro-kernels :**
   a. Moves as much from the kernel into "user" space communication takes place between user modules using message passing.
4. **Modules :**
   a. Here the kernel has a set of core components and links in additional services either during boot time or during runtime.
   b. Such a strategy uses dynamically loadable modules and is common in modern implementations of UNIX, such as Solaris, Linux, and Mac OS X.
   c. Such a design allows the kernel to provide core services yet also allows certain features to be implemented dynamically.
   d. The overall result resembles a layered system in that each kernel section has defined, protected interfaces; but it is more flexible than a layered system in that any module can call any other module.
   e. The approach is like the microkernel approach in that the primary module has only core functions and knowledge of how to load and communicate with other modules; but it is more efficient, because modules do not need to invoke message passing in order to communicate.

Advantages :

1. The main advantage of the layered approach is simplicity of construction and debugging.
2. The layers are selected so that each uses functions (operations) and services of only lower-level layers.
3. This approach simplifies debugging and system verification.
4. The first layer can be debugged without any concern for the rest of the system, because, it uses only the basic hardware to implement its functions.
5. Once the first layer is debugged, its correct functioning can be assumed while the second layer is debugged, and so on.
6. If an error is found during the debugging of a particular layer, the error must be on that layer, because the layers below it are already debugged.
7. Thus, the design and implementation of the system is simplified.

Disadvantages :

1. The major difficulty with the layered approach involves appropriately defining the various layers. Because a layer can use only lower level layers, careful planning is necessary.
2. A final problem with layered implementations is that they tend to be less efficient than other types. At each layer, the parameters may be modified, data may need to be passed, and so on. Each layer adds overhead to the system call; the net result is a system call that takes longer than one on a non-layered system.
3. These limitations have caused a small backlash against layering. Fewer layers with more functionality are being designed, providing most of the advantages of modularized code while avoiding the difficult problems of layer definition and interaction.

Que 1.12. What is an operating system ? Define the components of an operating system. **AKTU 2013-14, Marks 05**

OR

Discuss various operating system components. **AKTU 2014-15, Marks 05**

Answer

Operating system : Refer Q. 1.1, Page 1–2B, Unit-1.

Components of an operating system are :

1. **Memory management :**

   An operating system does the following activities for memor management :

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

   b. In multi-programming, the OS decides which process will ge memory when and how much.

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

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

2. **Processor management :**

   An operating system does the following activities for processor management :

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

   b. Allocates the processor (CPU) to a process.

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

3. **Device management :**

   An operating system manages device communication via their respective drivers. It does the following activities for device management :

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

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

   c. Allocates the device in the efficient way.

   d. De-allocates devices.

4. **File management :**

   An operating system does the following activities for file management :

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

   b. Decides who gets the resources.

   c. Allocates the resources.

   d. De-allocates the resources.

ue 1.13. Define the services provided by the operating system.

AKTU 2013-14, Marks 05

OR

Explain the services provided by operating system.

AKTU 2016-17, Marks 10

Answer

Some of the services provided by operating system are as follows :

1. **Program execution :**

   a. The purpose of computer system is to allow the users to execute programs in an efficient manner.

   b. The operating system provides an environment where the user can conveniently run these programs.

   c. To run a program, the program is required to be loaded into the RAM first and then to assign CPU time for its execution.

2. **I/O operations :**

   a. Each program requires an input and after processing the input submitted by user it produces output.

   b. This involves the use of I/O devices.

   c. The I/O service cannot be provided by user-level programs and it must be provided by the operating system.

3. **File system manipulation :**

   a. While working on the computer, generally a user is required to manipulate various types of files like opening a file, saving a file and deleting a file from the storage disk.

   b. This is an important task that is also performed by the operating system.

4. **Communication :**

   a. Operating system performs the communication among various types of processes in the form of shared memory.

   b. Communications may be implemented via shared memory, in which two or more processes read and write to a shared section of memory, or message passing, in which packets of information in predefined formats are moved between processes by the operating system.

5. **Error detection :**

   a. The main function of operating system is to detect the errors like bad sectors on hard disk, memory overflow and errors related to I/O devices.

   b. After detecting the errors, operating system takes an appropriate action for consistent computing.

6. **Resource allocation :**
   a. In the multitasking environment, when multiple jobs are runn at a time, it is the responsibility of an operating system to alloc the required resources (like CPU, main memory, tape drive secondary storage etc.) to each process for its better utilization.
8. **Protection and security :**
   a. Protection involves ensuring that all access to system resources controlled.
   b. Such security starts with requiring each user to authenticate hi or her to the system, usually by means of a password, to gain acce to system resources.

Que 1.14. What is kernel ? Describe various operations performe by kernel ? **AKTU 2016-17, Marks 10**

Answer

1. Kernel is the main component of most computer operating systems.
2. It provides an interface between applications and actual data processing at the hardware level.
3. Kernel is considered as the heart of an operating system.
4. Kernel provides the lowest-level abstraction layer for the resources (especially processors and I/O devices) that application software must control to perform its function.

The various types of operations performed by the kernel are :

1. It controls the state of the process that is it checks whether the process is running or process is waiting for the request of the user.
2. It provides the memory for the processes those are running on the system that is kernel runs the allocation and de-allocation of process.
3. The kernel also maintains a time table for all the processes those are running that is the kernel also prepare the schedule time which will provide the time to various process of the CPU.
4. The kernel puts the waiting and suspended jobs into the different memory area.
5. When a kernel determines that the logical memory does not fit to store the programs, then it uses the concept of the physical memory which will store the programs into temporary manner. Therefore, physical memory of the system can be used as temporary memory.

Que 1.15. Describe re-entrant kernels.

Answer

1. A re-entrant kernel is one where many processes or threads can execute in the same kernel program concurrently without affecting one another.
2. In non-reentrant kernels, a process does not modify kernel programs, but modify global kernel data.
3. If a process is executing operating system programs, no other processes may be allowed to execute the programs, nor may system start another kernel path execution when the kernel access the global data.
4. An interrupts from I/O devices may not be handled immediately by the operating system.
5. Normally the operating system is composed of re-entrant and non-reentrant functions. The re-entrant functions may modify local data, but they do not modify any global data.
6. Concurrent executions of re-entrant functions do not affect the behaviour of one another. The operating system makes sure that non-reentrant functions are executed mutually exclusively by the kernel paths so that the function executions do not modify global data at the same time.

Que 1.16. Differentiate between the following :

i. Shell and kernel

ii. Monolithic kernel and micro-kernel

AKTU 2011-12, Marks 05

OR

What are the differences between shell and kernel ?

AKTU 2014-15, Marks 05

Answer

i. Shell and kernel :

S. No.	Shell	Kernel
1.	When a user logs in, the login program checks the username and password, and then starts another program called the shell.	The kernel is the hub of the operating system, it allocates time and memory to program and handles the file storage a communications in response system calls.
2.	Shell gives interface between user and kernel.	Kernel gives the hardware interaction with user.
3.	The shell is the GUI or interface to the kernel.	The kernel is the part of t operating system that int with the hardware.

ii. Monolithic kernel and micro-kernel :

S. No.	Monolithic kernel	Micro-kernel
1.	Kernel size is large.	Kernel size is small.
2.	OS is complex to design.	OS is easy to design, implement and install.
3.	All the operating system services are included in the kernel.	Kernel provides only IPC and low level device management services.
4.	Request may be serviced faster.	Request may be serviced slower than monolithic kernel.
5.	No message passing and no context switching are required while the kernel is performing the job.	Micro-kernel requires message passing and context switching.

Que 1.17. What do you understand by system call ? How is a system call made ? How is a system call handled by the system ? Choose suitable examples for explanation. **AKTU 2011-12, Marks 05**

Answer

System calls :

1. System calls provide an interface to the services made available by an operating system.
2. These calls are generally available as routines written in C and C++, although certain low-level tasks (for example, task where hardware must be accessed directly), may need to be written using assembly language instructions.

System call is made :

1. An operating system makes system calls available, let's first use an example to illustrate how system calls are used : writing a simple program to read data from one file and copy them to another file.
2. The first input that the program will need is the names of the two files : the input file and the output file.
3. One approach is for the program to ask the user for the names of the two files.
4. In an interactive system, this approach will require a sequence of system calls, first to write a prompting message on the screen and then to read from the keyboard the characters that define the two files.
5. Once the two file names are obtained, the program must open the input file and create the output file. Each of these operations requires another system call.

When the program tries to open the input file, it may find that there is no file of that name or that the file is protected against access.

7. In these cases, the program should print a message on the console and then terminate abnormally.
8. If the input file exists, then we must create a new output file.
9. We may find that there is already an output file with the same name. This situation may cause the program to abort, or we may delete the existing file and create a new one.
10. On input, the program may find that the end of the file has been reached or that there was a hardware failure in the read (such as a parity error.)
11. Finally, after the entire file is copied, the program may close both files, write a message to the console or window, and finally terminate normally.
12. Typically, application developers design programs according to an application programming interface (API).
13. The API specifies a set of functions that are available to an application programmer, including the parameter that are passed to each function and the return values the programmers can expect.
14. The functions that make up an API typically invoke the actual system calls on behalf of the application programmer.

System call handled by system :

1. The runtime support system for most programming languages provides a system call interface that serves as the link to system calls made available by the operating system.
2. The system call interface intercepts function calls in the API and invokes the necessary system calls within the operating system.
3. Typically, a number is associated with each system call, and the system call interface maintains a table indexed according to these numbers.
4. The system call interface then invokes the intended system calls in the operating system kernel and returns the status of the system call and any return values.
5. The caller need know nothing about how the system calls is implemented or what it does during execution.
6. Rather, it need only obey the API and understand what the operating system will do as a result of the execution of that system call.
7. Thus, most of the details of the operating system interface are hidden from the programmer by the API and are managed by the runtime support library.

Que 1.18. What do you understand by system call ? Enumerate five system calls used in process management.

AKTU 2013-14, Marks 05

OR

What do you understand by system call ? List and explain three system calls used for process management.

AKTU 2012-13, Marks 05

Answer

System call : Refer Q. 1.17, Page 1–17B, Unit-1.

System calls used for process management :

fork() : To create a new process
exec() : To execute a new program in a process
wait() : To wait until a created process completes its execution
exit() : To exit from a process execution
getpid() : To get a process identifier of the current process
getppid() : To get parent process identifier
brk() : To increase/decrease the data segment size of a process

Que 1.19. What are the advantages of the layered approach to the design of operating system ? **AKTU 2012-13, Marks 05**

1. The main advantage of the layered approach is simplicity of construction and debugging.
2. The layers are selected so that each uses functions (operations) and services of only lower-level layers.
3. This approach simplifies debugging and system verification.
4. The first layer can be debugged without any concern for the rest of the system, because, it uses only the basic hardware to implement its functions.
5. Once the first layer is debugged, its correct functioning can be assumed while the second layer is debugged, and so on.
6. If an error is found during the debugging of a particular layer, the error must be on that layer, because the layers below it are already debugged.
7. Thus, the design and implementation of the system is simplified.

Que 1.20. What is the reason behind dual mode operation of processors ? **AKTU 2011-12, Marks 05**

OR

Whether it is possible to construct a secure operating system without having dual mode of operation in the system ? Give arguments in favour of your answer. **AKTU 2012-13, Marks 05**

Answer

1. In dual mode operation, two separate modes are used for working of operating system. These modes are user mode and monitor mode.
2. For indicating mode of the system, mode bit is used in the computer hardware. The mode bit is 0 for monitor and 1 for user.
3. With the mode bit, we are able to distinguish between a task that is executed in user mode or monitor mode.
4. This feature helps to the operating system in many ways :
   a. At the booting time, the hardware starts in the monitor mode, then operating system is loaded. The hardware switches from user mode to monitor mode when interrupts occur. When the operating system gains control of the system, it is in monitor mode.
   b. The dual mode operation provides the protection to the operating system from unauthorized users.

   c. The privileged instructions are executed only in the monitor mode. The computer hardware is not allowed for executing the privilege instructions in other mode, *i.e.* user mode.
   d. These provide greater protection for the operating system. Fig. 1.20.1 shows the dual mode protecting.

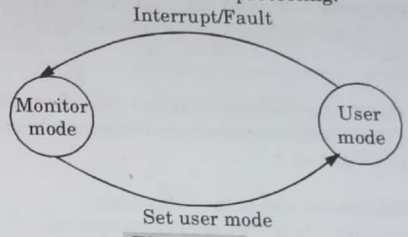


Fig. 1.20.1.

Que 1.21. Differentiate between process switch and mode switch. **AKTU 2012-13, Marks 2.5**

Answer

S. No.	Process switch	Mode switch
1.	Process switching occurs when the processor switches from one process to another.	Mode switching is the switching of a process between kernel mode and user mode. A process can execute in either of these two modes.
2.	When process switching occurs, the kernel saves the context of old process in its PCB and loads the saved context of new process scheduled to run.	Mode switching can be done using system call. This is a special instruction that sets the system's state to kernel mode.
3.	Example : If an interrupt occurs, the system needs to save the current context of running process on CPU, so that it can restore the saved context when processing of interrupt is over, *i.e.*, process switching is useful for suspending and resuming of a process.	Example : If a user process needs to access things controlled by kernel, it is necessary to perform mode switching.

VERY IMPORTANT QUESTIONS

Following questions are very important. These questions may be asked in your SESSIONALS as well as UNIVERSITY EXAMINATION.

Q. 1. Write down the different types of an operating system and also write functions of an operating system.

Ans. Refer Q. 1.2 and Q. 1.1.

Q. 2. Differentiate between the following :

i. Hard real time and soft real time system.

ii. Multi-processing and multi-programming.

Ans.

i. Refer Q. 1.3.

ii. Refer Q. 1.7.

Q. 3. Explain layered structure of an operating system.

Ans. Refer Q. 1.11.

Q. 4. Define the services provided by the operating system.

Ans. Refer Q. 1.13.

Q. 5. Differentiate between the following :

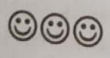
i. Shell and kernel

ii. Monolithic kernel and micro-kernel

Ans. Refer Q. 1.16.

Q. 6. How is a system call handled by the system ? Enumerate the five system calls used in process management.

Ans. Refer Q. 1.17 and Q. 1.18.



2 UNIT

Concurrent Processes