OS-1 19UCS127

Assignment No.1

Page No.: 1 YOUVA
Date:

Q.1] Define OS and list the basic services provided by OS.

→ ① An operating system is a program that manages a computer's hardware. It also provides a basis for application programs and acts as an intermediary between the computer user and the computer hardware.

② Operating system in the one program running at all times on the computer usually called kernel.

③ basic services provided by OS :-

A] Hardware - It provides basic computing resources for system.
   ① Central Processing Unit
   ② Memory
   ③ Input/output devices

B] Operating system - It controls the hardware & coordinates its use among the various application programs for various user.

C] Application programs → It defines the ways in which the system resources are used to solve the computing problem of user.
   ② word processors, web browsers, database system, vedio games

D] Users :- People, other computer / machines.

Q. 2] Differentiate bet$^n$. single processor system & multiprocessor system.

→

| Single processor os | multiprocessor os. |
|---|---|
| ① They performs only one process at a given time. | ① These refers to the use of two / more central processing units within a single computer system. |
| ② It carries out next process in the queue only after the current process is completed. | ② These multiple CPU's are in close communication sharing the computer bus, memory & other peripheral devices. |
| ③ Os monitors the status & also sends them next executable instruction. | ③ These types of system are used when very high speed is required to process a large volume of data. |

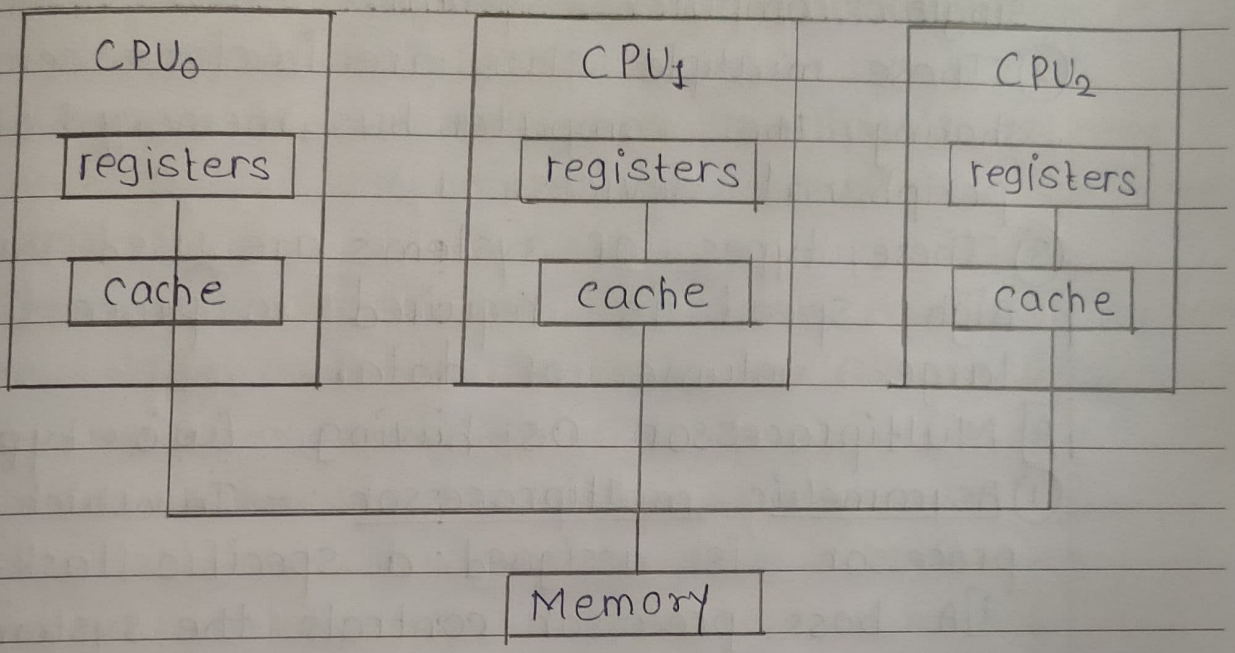
Q. 3] Explain symetric and asymetric multiprocessor system with neat diagram.

→ ① Symetric multiprocessor system :-

(A) In this, each processor performs all tasks within the operating system.

(B) SMP means that all processors are peers; no boss-worker relationship exist between processor.

(C) each processor has its own set of registers, as well as a private or local cache.
(D) However, all processors share physical memory.




Symmetric multiprocessing architecture.

(2) Asymetric multiprocessing architecture :-
(A) In this, each processor is assigned a specific task.
(B) A boss processor controls the system; the other processors either look to the boss for inst. or have predefined tasks.
(C) This scheme defines a boss-worker relationship.
(D) The boss processor schedules and allocates work to worker processor.

Q. 4] explain multiprocessor OS & their types in detail.

→ ① Multiprocessor os refers to the use of two or more central processing units within a single computer system.

② These multiple CPU's are in close communication sharing the computer bus, memory & other peripheral devices.

③ These types of systems are used when very high speed is required to process a large volume of data.

A] Multiprocessor os having two types:-

① Asymmetric multiprocessor :- In which each processor is assigned a specific task.

i] A boss processor controls the system; the other processors either look to the boss for instruction or have predefined task.

ii] This scheme defines a boss-worker relationship.

iii] The boss processor schedules & allocates work to the worker processor.

② Symmetric multiprocessing :- In which each processor performs all tasks within the os.

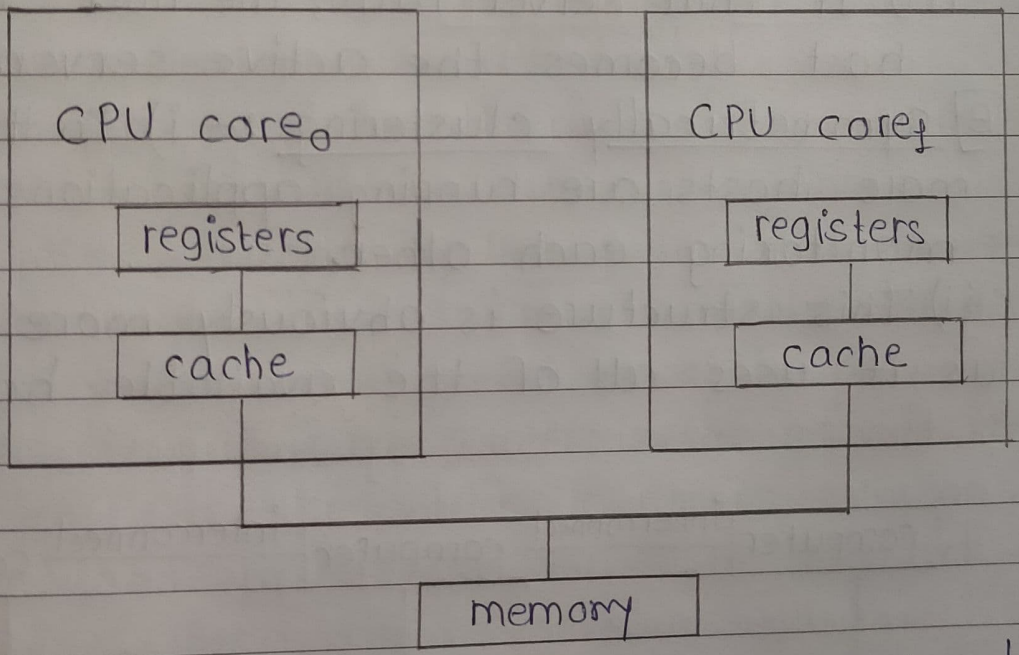
i] SMP means that all processor are peers; no boss-worker relationship exists betⁿ processor.

ii] each processor has its own set of register, as well as a private / local cache.

iii] However, all processors share physical memory.

③ A multi-core processor :- It is a single computing component with two or more independant processing units called cores, which read & execute program instructions.

① They can be more efficient than multiple chips with single cores because on-chip communication is faster than bet$^{n}$. chip communication.
② They can be more efficient the multiple chips with single cores because on-chip communication is faster than bet$^{n}$.
② In addition, one chip with multiple cores uses significanthy less powee than multiple single-core chips.
③ It is important to note that while multicore systems are multiprocessor systems, not all multiprocessor system are multicore.



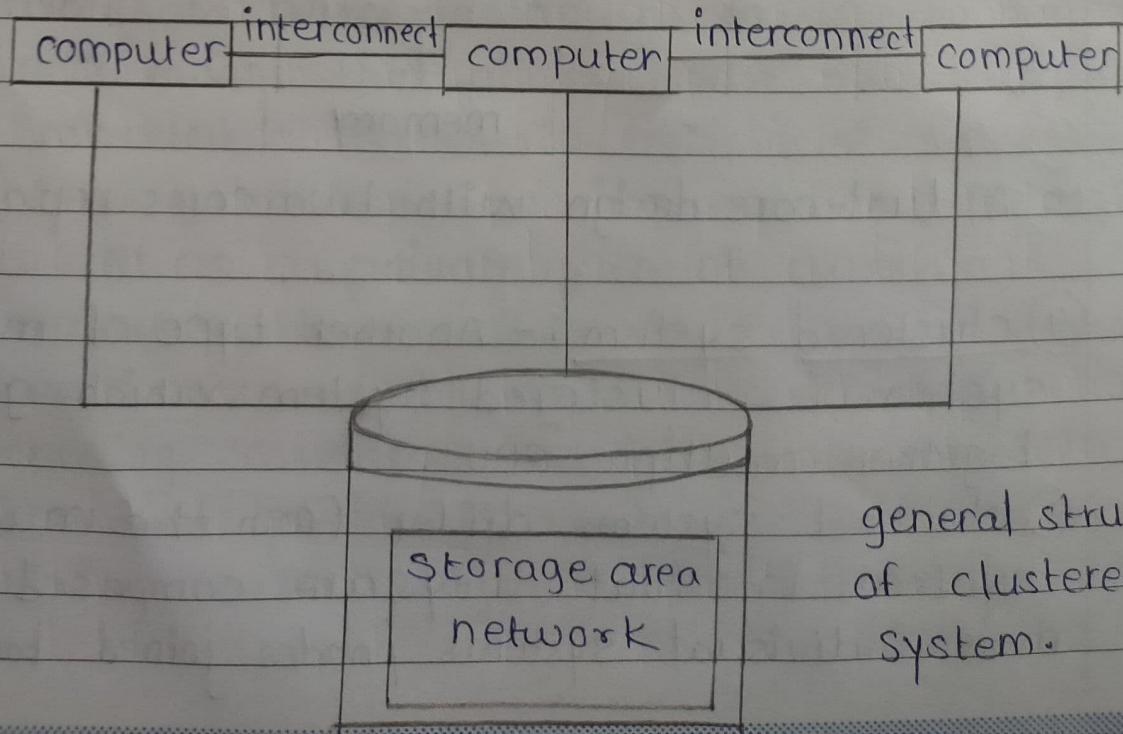

A dual-core design with two cores replaced on same chip.

④ <u>clustered system</u> :- Another type of multiprocessor system is clustered system, which gathers together multiple CPUs.
① clustered systems differ from the multiprocessor system in that they are composed of two/more individual systems/nodes joined together.

② such systems are considered loosely coupled.
③ each node may be a single processor system or multi-core system.
④ clustering is usually used to provide high-availability service that is, service will continue even if one/more systems in the cluster fail.
⑤ clustering can be structured assymetrically or symmetrically.

A] assymetrically ~~cluste~~ clustering :- i) In this, one machine is in hot-standby mode while the other is running the application.
ii) The hot-standby host machine does nothing but monitor the active server.
iii) If that server fails, the host-standby host becomes the active server.

B] symmetrically clustering :- i) In this two or more hosts are running applications & are monitoring each other.
ii) This structure is obviously more efficient, as it uses all of the available hardware.




general structure of clustered system.

Q. 5] Define following type of systems with their properties.

a] Time sharing system :-

→ Time sharing (multitasking) is logical extension of multiprogramming in which CPU switches jobs so frequently that users can interact with each job while it is running, creating interactive computing.

Properties :-

① Response time should be <1 second.
② each user has at least one program executing in memory ⇒ Process
③ If several jobs ready to run at the same time, CPU uses ⇒ CPU Scheduling
④ If processes don't fit in memory, swapping moves them in & out to run.
⑤ A time sharing system must also provide a file system that resides on a collection of disks; hence, disk managment must be provided.
⑥ Also, time-sharing system must provide mechanism for job synchronization & communication, ensure that jobs do not get stuck in deadlock, forever waiting for one another.

b] Distributed system :-

→ A distributed system is a collection of physically seperate, possibly heterogeneous, computer systems that are networked to provide users with access to the various resources that the system maintains.

Properties :-

① Access to a shared resources increases computation speed, functionality, data availability and reliability.

② Generally, systems contain a mix of two modes — for example FTP and NFS.

③ Distributed systems depend on networking for their functionality.

Q. 6] Write a note on:

a] Computing Environments :-

→ ① Traditional computing :-

A] Traditional time-sharing system are uncommon which used timer & scheduling algorithms to cycle processes rapidaly through the CPU, giving each user a share of resources.

B] The current trend is toward providing more ways to access these computing environments.

C] User processes and system processes that provide services to the user, are managed so that each frequently gets a slice of computer time.

D] Web technologies & increasing bandwidth are stretching the boundaries of traditional computing.

E] companies establish portals, which provide web accessibility to their internal servers.

② Mobile computing :-

A] Refers to computing on handled smartphones and tablet computer. These devices share the

distinguishing physical features of being portable & lightweight.

B] The memory capacity & processing speed of mobile devices, however are more limited that those of PCs. whereas a smartphones or tablet may have 64GB in storage, it is not uncommon to find 1 TB in storage on desktop computer.

c] Two operating systems currently dominate mobile computing Apple ios and Google Android. ios was designed to run on Apple iphone and ipad mobile devices. Android powers smartphones & tablet computers available from many manufactures.

(3) Distributed systems:

A] A distributed systems is a collection of physically seperated, possibly heterogeneous, computer system that are networked to provide users with access to the various resources that the system maintains.

B] Access to a shared resource increases computation speed, functionality, data availability and reliability.

c] Generally, systems contains a mix of the two modes - for example FTP and NFS.

D] Distributed system depend on networking fo theie functionality.

(4) client - Server computing:

A] Terminals connected to centralized systems are now being supplanted by PCs & mobile devices.

B] many of today's systems act as server systems to satisfy request generated by client systems. This form of specialized distributed system is called client - server system.

(5) Peer-to-peer computing :-

A] In the peer-to-peer system model, clients and servers are not distinguished from one another.

B] Instead, all nodes within the system are considered peers & each may act as either a client or a server, depending on whether it is requesting or providing a service.

(6) Virtualization :-

A] Allows operating systems to run as applications within other operating systems using emulation.

B] Here, every machine-level instruction that runs natively on the source system must be translated to equivalent function on target system, frequently resulting in several target instruction.

(7) cloud computing :-

A] Computing that delivers computing storage, & even applications as a service across a network.

B] In some ways, it's logical extension of virtualization, because it uses virtualization as a base for its functionality.

b] System calls :-

→ (1) In computing, a system call is the programmatic way in which a computer program request a service from the kernel of operating system it is executed on.

(2) A system call is a way for programs to interact with the operating system.

(3) System call provides the services of operating system to the user programs via Application program Interface (API)

(4) An example to illustrate how system call are used:

i] writing a simple program to read data from one file and copy them to another file.

ii] The first input that the program will need is the names of two files: the input file & output file.

iii] These names can be specified in many ways, depending on the OS design.

iv] Once the two file names have been obtained, the program must open the input file & create the output file.

v] Each of these operations requires another system call.

(5) Possiple error conditions for each operation can require additional system calls.

(6) When the program tries to open the input file, for ex, it may find that there is not file of that name or that the file is protected against access.

(7) These cases, the program should print a message on the console.

(8) Simple programs may take heavy use of OS. Frequently, systems execute thousands of system call per second.

(9) Typically, application developers design programs according to an application programming interface (API).

(10) API is a set of function & procedures that allow the creation of application which access the features or data of an OS, application or other services.

(11) Services provided by system calls :-
A] Process creation & managment
B] Main memory managment
C] File access, Directory & file system managment
D] Device handling (I/O)
E] Protection
F] Networking etc.

(12) Types of system call :-
A] Process control
B] File manipulation
C] Device manipulation
D] Information maintenance
E] communications
F] Protection.

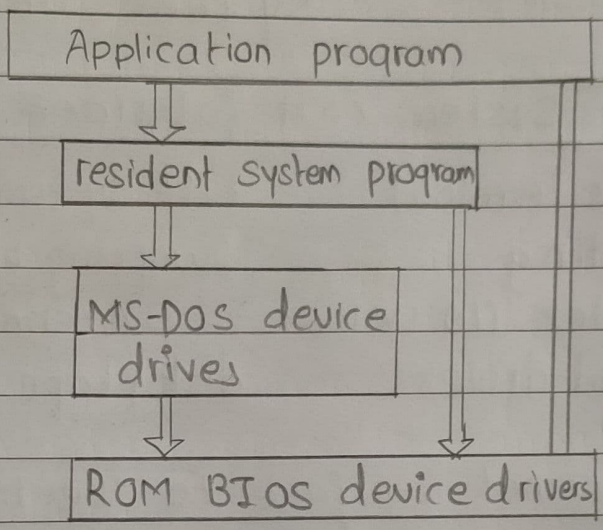
d] ~~OS structure~~ :-

c] OS structure :-

→ (1) Many operating systems do not have well-defined structures.
(2) Frequently, such systems started as small, simple & limited system & then grew beyond their original scope.
(3) MS-DOS is an example of such a system. It was originally designed & implemented by a few people who had no idea that it would become so popular.
(4) It was written to provide the most functionality in the least space, so it was not carefully divided into modules.
(5) In MS-DOS, the interfaces & levels of functionality are not well seperated. So, application program are able to access the

basic I/O routines to write directly to the display & disk drives.

(6) These leaves MS-DOS vulnerable to errant program, causing entire system crashes when user programs fail.




MS-DOS layer structure

(7) Another example of limited structuring is the original UNIX OS.

(8) Like MS-DOS, UNIX initially was limited by hardware functionality.

(9) It consists of two seperable parts: the kernel & the system programs.

(the users)

Shells and commands
compilers & interpreters
system libraries

System-call interface to kernel

kernal

| | | |
|---|---|---|
| signals terminal handling | file system | CPU scheduling |
| character I/O system | swapping block I/O system | page replacement |
| terminal drivers | disk & tape drivers | demand paging |
| | | virtual memory |

Kernel interface to the hardware

| | | |
|---|---|---|
| terminal controllers | device controller | memory controller |
| terminals | disk & tapes | physical memory |

Traditional UNIX system structure