

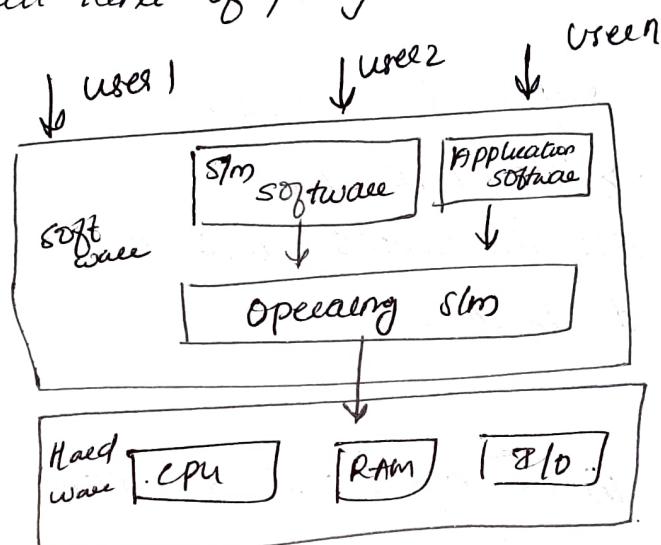
10/11/2020

① With neat diagram, explain the functions of the OS?

OS is an interface between a computer user and computer hardware. An OS is a software which performs all basic tasks like file management, memory management, process management, handling I/O & O/P and controlling peripheral devices.

Popular OS: Linux, Windows OS, VMS, OS/400, AIX, z/OS etc.

It is an interface b/w user and hardware and controls the execution of all kind of programs.



Functions

- 1. Memory management
 - 2. Processor management
 - 3. Device management
 - 4. File management
 - 5. Security
 - 6. Control over S/M performance.
 - 7. Job accounting
 - 8. Error detecting aids
 - 9. Co-ordination b/w other software & user.
-
- 1. Memory management.
 - Management of primary/main memory
 - Keeps track of 1^o memory, that is what part of it are in use by whom, what part are not in use.
 - In multiprogramming, OS decides which process will get memory, and how much.
 - Allocates memory when a process requests it to do so.

- Deallocates my when process no longer need or it terminates
- ② processor management.

In multiprogramming, OS decides which process gets the processor when and for how much time. This is called process scheduling.

- Keep track of processor and status of process. - The program responsible for this task called Clock controller
- Allocates the processor (cpu) to a process
- Deallocates processor when no need / terminates.

③ Device management

manage devices via respective drivers.

- keep track all device → I/O controller
- Decides which process gets the devices and when & for how much time
- Allocates device and deallocates

④ File Management.

File s/m organised in to directories for easy navigation & usage. These directories contains files and directories.

- keep track of organization, location, uses, status etc. → file s/m
- Decides who get resource
- allocates / deallocates resources.

⑤ Security - By means of password & other similar technique, it prevent unauthorised access.

⑥ control over s/m performance - Recording delays b/w requests for service and response from s/m.

⑦ Job accounting - keep tracking of time & resources used by various jobs and users.

⑧ Error detecting aids - production of dumps, traces, error msgs and other debugging & error detecting aids

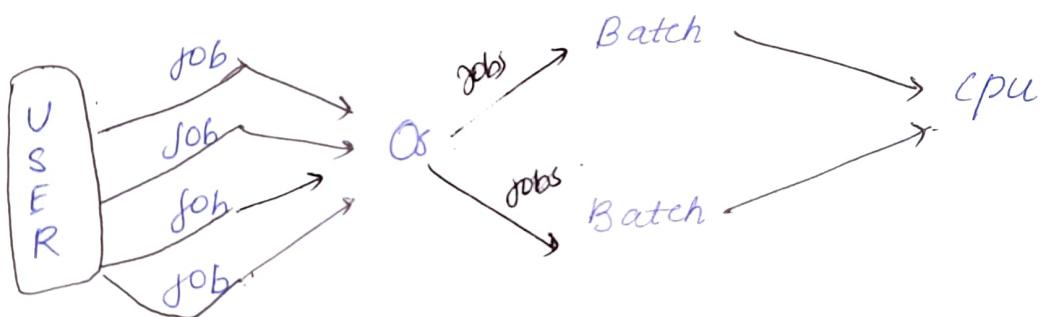
⑨ coordination b/w other software & user - co-ordination & assignment of compilers, interpreters, assemblers & other software to various users of computer s/m.

Q) With Neat diagram, explain concept of batch processing?

Batch processing works as an OS. Batch processing S/m means to grab all types of programs and data in the batch form, then proceed to process. Main use of batch processing is to decrease the set up time, where same job given to CPU.

It is implemented in hard disk and card readers as well. The jobs are saved on hard disk to make their execution as batch form. These jobs divides in to groups and finally precede same job to similar batch. Now all batched jobs are ready to execution one by one without wasting time and due to this system enhance the system utilization while decrease turnaround time.

executed in FCFS base



when job complete, then only file and result for job get copied as a output spool for further printing otherwise processing.

Advantages of Batch processing OS.

- All jobs are performed in repeating form without user permission.
- can be feed data without using extra hardware components.
- Batch S/m capable to work in Offline mode.
- less time to execute all job.
- shareable in nature, so can be share to couple of user.
- idle time is low
- huge repeated tasks can handle smoothly.

④ List and explain briefly the classification of software application.

Application software is dealing with user input and helps the user to complete the task. It resides above ~~application~~ ^{SLM} software. It is programmed for single as well as complex task. It is either installed or access online. It can be single program or group of program.

Types of application program

① presentation software: It is the programs which shows information in form of slides. we can add text, graphic, video and images to slides to make them more informative.

It has:

text editor,

inserting graphics, text, video, and other multimedia slide show to display the information.

It helps the presenter to present their ideas with ease and visual information ~~key~~ easy to understand.

Eg: Microsoft ppt.

2. Spreadsheet software: It is used to perform manipulations and calculations. In spreadsheet software data is stored in information intersection row and column. The intersections called cell. It has many formula and functions to perform calculation like arithmetic operations, logical etc. It provides charts, graphs to display data graphically.

3. Database software: Collection of data related to any application can be done. The data regarding user is stored. The application data is accessed from the database and after manipulation it gets back stored in the database.

Database management systems (DBMS) software tool used for storing, modifying, creating and searching for information within a data base. MS access, SQL server and Oracle is the example.

④. multimedia software: multimedia is a combination of graphics, audio and multimedia software used in editing of video, audio and text.

⑤. simulation software: It is an imitation of real world and environment. The simulation creates a physical environment of real world to represent the simulae behavior and key nature of the selected topic.

Simulator is technology for education, engineering, testing and learning, video games and for scientific modelling of natural sims. Area of technology flight, economics, automobile, Robotics, digital lifecycle, space shuttle etc.

⑥ word processing software: manipulate, format the text, to create memos, letters, faxes and documents. Processing software is used to format and beautify the text. It provides a list of features.

Eg: Microsoft word, Lotus word pro etc

11/11/2020

⑤ List and explain the difference b/w single task vs multitask vs multi~~tasking~~ applications?

① Single tasking :- 1 program at a time. functions like printing a document, downloading images etc. can be performed only one at a time. Examples include MS DOS, Palm OS etc.

Adv

less space in memory

disadv : only single task at a time

② Multi-tasking :- multiple jobs executed by CPU simultaneously by switching b/w them. User may interact with each program while it is running. The activities are

→ The user gives inst. to operating system or to program directly, & receives an immediate response.

→ OS can handle multiple operations/executes multiple programs at a time.

→ also called time sharing

→ It developed to provide interactive use of a computer system at reasonable cost.

→ process executes, it typically executes for only short time, before it either finishes or need to perform I/O, at that time, the CPU can be utilized by another process.

→ one CPU shared between many programs.

③ multiprogramming application

It is an OS that permits several users to use the programs that are concurrently running on a single network server. The single network server is termed as terminal server.

Adv

- highly productive as it performs multiple tasks at a time
- enables several programs concurrently.
- cooperative multi-tasking, 32 bit versions of both windows NT and used preemptive multitasking.
- MAS OS prior to OSX used to support cooperative multitasking.

⑥ What are device drivers. Explain briefly its function.

A device driver or hardware driver is a group of files that enable one or more hardware device to communicate with OS. Without driver, the computer would not able to send and receive data correctly to hardware.
 SP kind of software program controls a specific hardware. The need of driver is the hardware devices that are unknown by OS or the features unknown for OS. The devices which need drivers: are

card reader

controller

modem

motherboard chipset

network card

printer

scanner

sound card

tape drive

USB device

video card

Device driver

device

hardware

For device like keyboard, monitor, disc drives CPU etc.. do not need drivers.

If appropriate driver is not installed, the device may not fn properly. With some devices, the device may work, but all its features may not work. Eg mouse. (The additional button works only if we install driver).

In device manager, if error ~~exists~~ or a driver config can occur and ~~this~~ ^{if this} is ~~not~~ encountered with devices, the computer / hardware manufacturer release driver update to fix the problem.

The decree only make the hardware installed in the computer function properly.

⑦ what is a kernel. Explain the type of kernel briefly?

A kernel is a computer program that is the heart and core of an operating system. Since OS has control over S/m, Kernel also has control over S/m. It is most important part of a system.

When S/m starts, the kernel is first program that loaded after boot loader because kernel has to handle the rest of the thing of S/m of OS. The kernel remains only until the OS is shut down.

The kernel responsible for low level task such as disk management, memory management, task management etc. It is an interface between user and hardware component of systems. When process makes a request to kernel, it is called system call.

Kernel provides ^{with} a protected kernel space which is a separate area of memory and area is not accessible by other application program. So the code of kernel is loaded in to this protected kernel space. Apart from this, the memory used by other application is called the user space. The communication b/w these 2 space is bit slower.

Functions

① Access computer resources → like CPU, I/O & other resource bridge b/w user & resource.

② Resource management

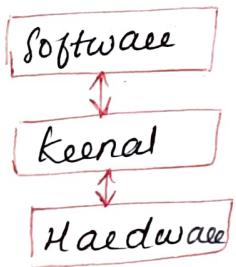
③ memory management → allocation & deallocation

④ Device management

Types of kernel

Generally there are 5 types.

1. Monolithic Kernel :- The user and kernel services are implemented in same memory space. The size of kernel increases thus the size of OS also increase. The kernel & user are in same space, so the process execution will be faster.



Adv :- Scheduling, memory management, file management all through system call only.

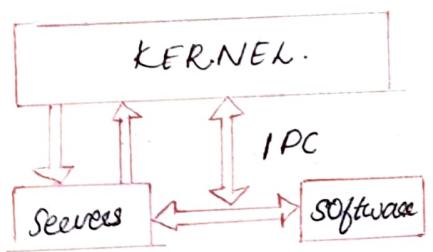
Disadv :- Fast execution

Disadv :- If any service fails, system fails

→ If need to add any service, the entire OS needs to modify.

2. Microkernel

In microkernel the user service and kernel service are implemented in different spaces. The kernel space and user space are separate here, so it reduces size of kernel & this in turn reduce the size of OS.



IPC - Interprocess communication

Here we are using different spaces, the communication between services are done with help of message passing and this in turn reduce speed of execution.

Adv :- Services can be easily added

Disadv :- Communication between services increase overall execution time.

3. Hybrid kernel

Combination of both monolithic & microkernel.

make use of the speed of monolithic & modularity.

8 microkernel.

It is the microkernel that have some non essential code in kernel space for the code to run more quickly than it would be in user space. So some services such as network stack / file system are run in kernel space to reduce the performance overhead, but still it runs kernel code as services in user space.

4. Nano Kernel

The whole code of kernel is small. i.e. the code executing in the privileged mode of the hardware is very small. It is used to describe a kernel that supports a ns clock resolution.

5. Exokernel.

Developed by the MIT parallel and distributed OS group. Here, the resource protection is separated from the management and this in turn result in allowing user to perform application-specific customization.

It will impose few abstractions as possible and by doing so, the abstraction should be used only when needed. So no fore abstraction will be there in Exokernel and this feature make different from monolithic kernel & microkernel. The drawback is the design is complex.

③ With neat diagram explain the type concept of monolithic & microkernel?

Monolithic

Monolithic kernel is an OS architecture where the entire OS is working in kernel space. The user services & kernel services are implemented under same address space. This increase the size of kernel further increase the size of OS.

Monolithic code

The are able to dynamically load executable modules at run time. It manages system resource b/w application and hardware of 8lm.

It provides CPU scheduling, memory management & other OS fn through 8lm call. Kernel & user space in same address space, thus executes fast operation.

If one service fails, entire 8lm is crashed. If new service to be added, the entire OS should be modified.

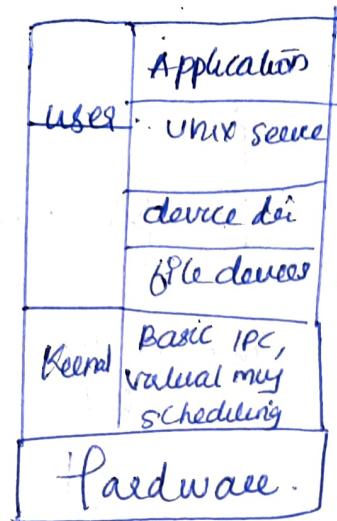
Eg: Linux, free BSD, Open BSD, Solaris, OS-9, AIX, HP-UX, MS windows (95, 98, Me), Open VMS, XTS-400 etc..

User space	Applications Libraries
Kernel space	File systems Inter process comm. I/O & Device management Fundamental process management
	Hardware

Micro kernel

The core functionality is isolated from system services and device drivers.

The user service and kernel services are implemented in different address space. The user services are kept in user address space and kernel services are kept in kernel address space also reduce the size of kernel & size of operating sys as well.



It provides minimum service of process & memory management. The communication b/w client / application and services running in user address space is established through message passing, reducing the speed of execution ~~mechanism~~. The OS remains unaffected as user services and kernel services are isolated so if any user service fails it does not affect kernel services. It is easily extendable i.e. if any new services are to be added they are added to user address space and hence requires no modification in kernel space. It is also portable, secure and reliable.

microkernel architecture - Only the most important services are inside kernel and rest of OS are present inside S/m application program. It is responsible for most important services of OS, they are

- inter process communication
- memory management
- CPU scheduling

Advantages

- small and isolated hence it can run better
- expansion of the S/m is easier, it is simply added in the S/m application w/o disturbing the kernel.

Eg. Eclipse IDE

Q9. What is virtualization? How virtualization helps? Give the command to check the support virtualization by a system.

Virtualization is a technology that helps us to install different OS on a hardware. They are completely separated and independent from each other.

Virtualization hides the physical characteristics of computing resources from their users, their applications or end users. This includes making a single physical resource (such as a server, an OS, an application or a storage device) appear to function as multiple virtual resources. It can also include multiple physical devices to single resources.

It can be used →

- ① Increase IT agility, flexibility and scalability
- ② Create workload mobility, increase performance.
- ③ boost efficiency
- ④ availability of resources and automated operation.

use task manager to check CPU virtualization, open task manager $CTRL + SHIFT + ESC \rightarrow$ if support virtualization, can be found Other details: Under performance tab, enabled or disabled.

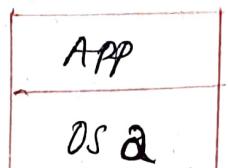
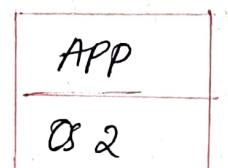
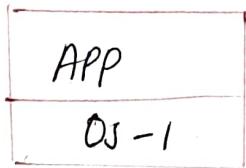
Q10. With a neat diagram, Explain the concept of Base metal & hosted virtualization.

A hypervisor is a thin software layer that intercepts or calls to the hardware. It is called as virtual machine monitor (VMM). It creates a virtual platform on the host computer on top of which multiple guest OS are executed and monitored.

- Hypervisor
- ① Nature: ~~of~~ ^{on} Base metal Hypervisor
 - ② Hosted Hypervisor

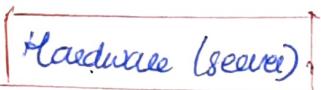
Base metal Hypervisor: are software systems that run directly on the host's hardware to control the hardware and to manage the Guest OS. The guest OS run a separate level above the hypervisor. All of them have virtual machine manager.

Oracle VM, Microsoft Hyper-V, VMware ESX and Xen.



Hypervisor type 1

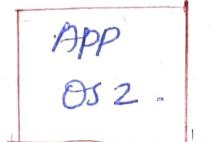
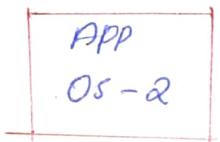
Eg: Oracle VM, Microsoft Hyper-V, VMware ESX and Xen.



Hosted Hypervisor

It is used to run in traditional OS. adds a distinct software layer on top of the hosting OS. While the guest OS becomes 3rd software layer above hardware.

Eg: Oracle VM VirtualBox, VMware Server and VM Workstation, KVM, QEMU and parallels.



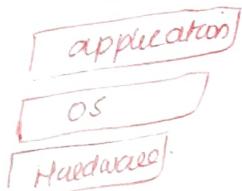
Hypervisor type 2, Eg: VMware server & Workstation

OS installed on Hardware

Hardware (server).

⑩ List the types of virtualisation and explain briefly?

O. Server virtualisation: Virtualising server infrastructure where you do not have to use any more physical server for different purpose.



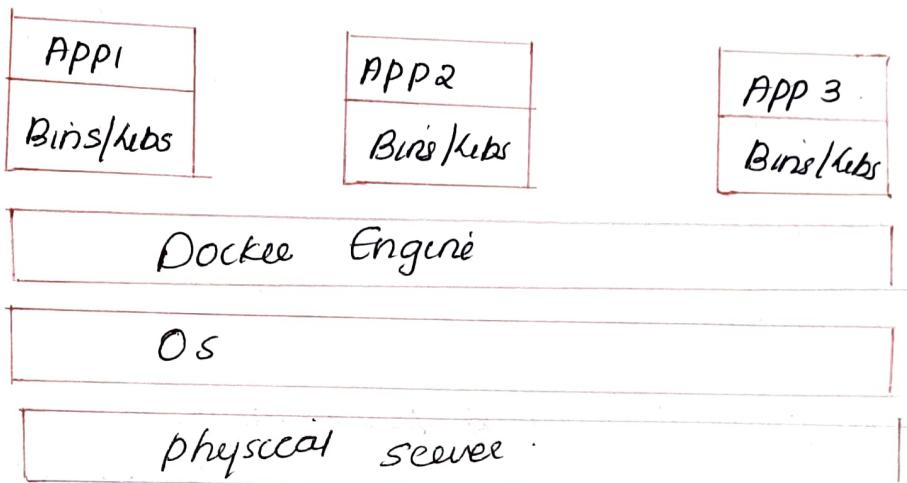
VMware ESX

Hardware

② Client & Desktop virtualization : On uses solo, virtualize their desktop. Change their desktop with thin client and by utilizing the data center resources.

③ Service and application virtualization :

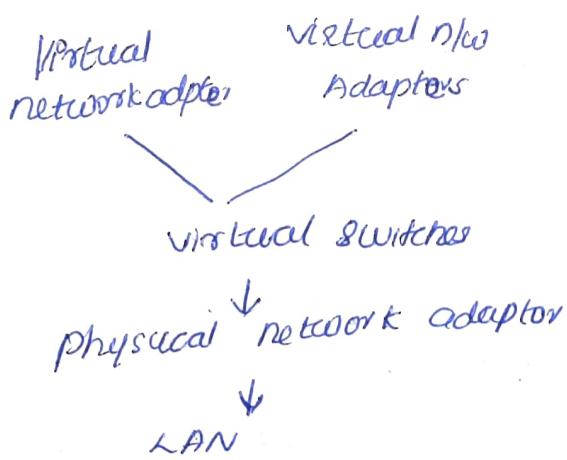
Isolates applications from underlying Os and from other applications in order to increase compatibility and manageability.
Eg: Docker can be used.



④ Network virtualization

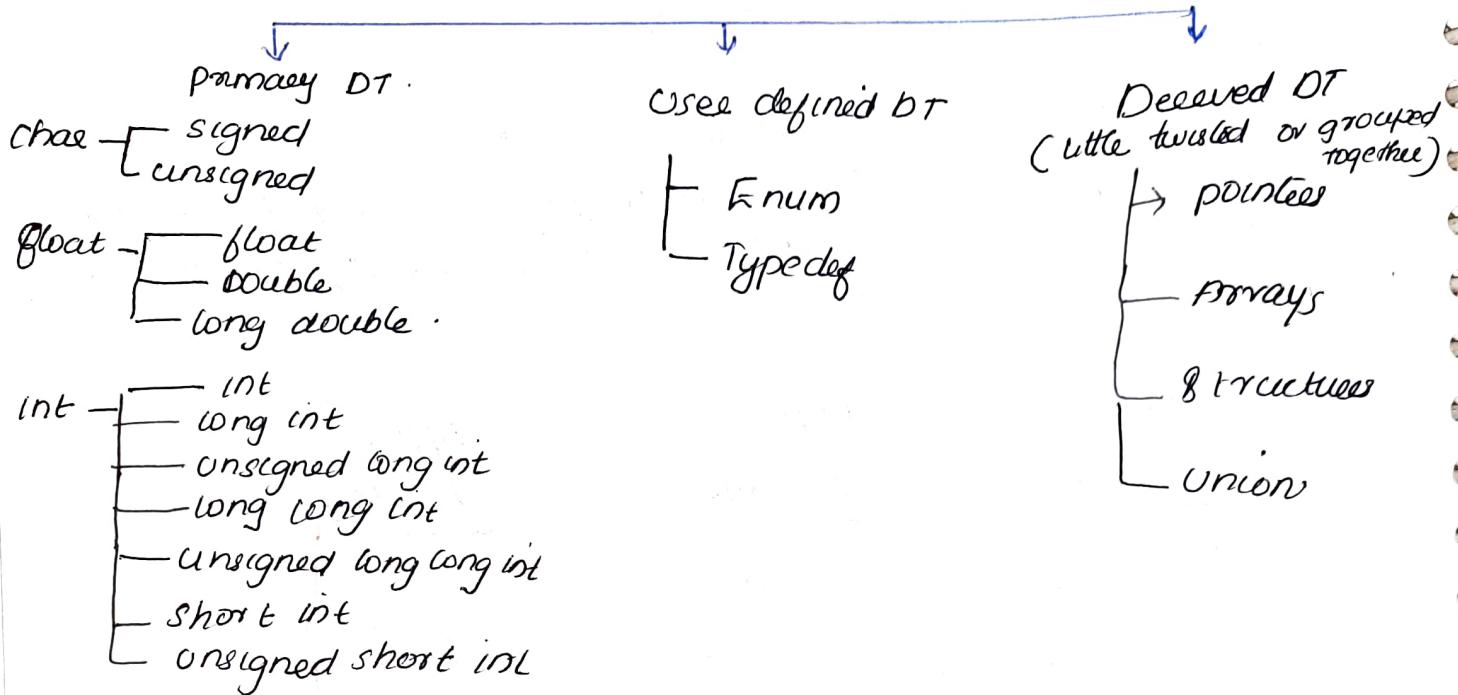
used if you are going to virtualize your server. It helps to create multiple switches, Vlans, NAT-ing, etc..

APP1 APP2
OS OS



⑤ storage virtualization → when have big storage & it helps you to create, delete allocated storage to different hardware.

(12) What are the data types supported in C language. Write a C pgm to check the datatype



double - double precision floating point value.

void - valueless special purpose ~~symbol~~ with meaning closely related

array - collection of data, all same type.

structure - C is collection of items of different types.

union - storing different data type in same memory location. one value at a time.

pointer : value of address of another variable.

Program

```
#include <stdio.h>
#include <conio.h>

int main ()
{
    clrscr();
    printf (" short int is %d \n", sizeof(short int));
    printf (" int is %d \n", sizeof(int));
    printf (" int * is %d bytes \n", sizeof(int *));
    printf (" long int is %d \n", sizeof(long int));
    printf (" long int * is %d \n", sizeof(long int *));
    printf (" signed int is %d \n", sizeof(signed int));
    printf (" unsigned int is %d \n", sizeof(unsigned int));
```

```

printf(" float is %2d bytes", size of (float));
printf(" float* is %2d bytes", size of (float*));
printf(" double is %2d bytes", size of (double));
printf(" double* is %2d bytes", size of (double*));
printf("\n");
printf(" char is %2d bytes", size of (char));
printf(" signed char is %2d bytes", size of (signed char));
return 0;
}

```

(13) What is padding of byte. Explain in how padding works in struct?

padding is a term define to filling a field with pad character.
Eg if name contains 8 characters, REVU becomes REVU0000.

Structural padding :-

many processors expect memory for variables to be aligned based on the size of variable. A char of 1 byte can be allocated anywhere in memory like 5000 or 5001. And if an integer, structural padding automatically done in all members are byte aligned and it will be 0x5000 or 0x5001.

Eg: Struct name {
 char ch;
 int num;
 double temp;
}

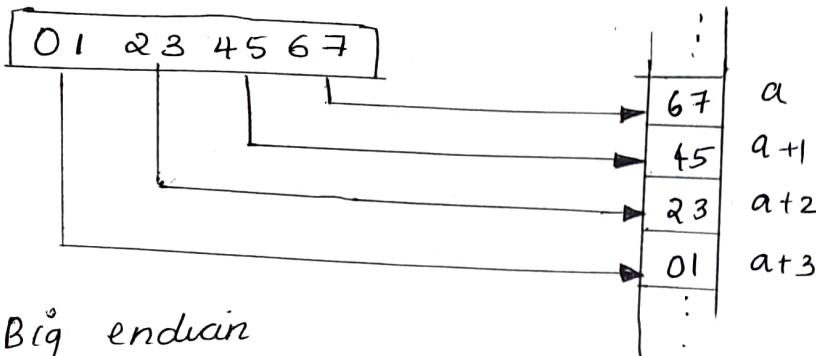
Index	1	2	3	4	5	6	7
	✓	✗	✗	✗	✓	✓	✓
char							
integer							

compiler used 3 wasted bytes to pad the structure so that all other memory are byte aligned. Now the size of structure = 8 bytes.

(14) What is endianness? Explain the type of endianness with neat & Endian is the order or sequence of bytes of word of digits of data in computer memory.

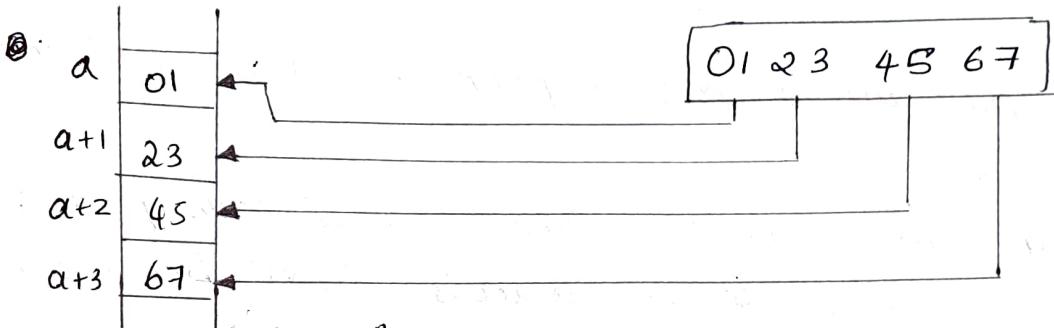
(12) Little endian

Eg: $0x01234567$



The last octet will stored first.

Big endian



The first byte will store first.

- (15) List the features of Windows, Mac and Linux OS compare and contrast?

The base of comparison	Windows	Mac	Linux
Basic difference & history.	First release in 1985. supposed to be GUI on top of MS-DOS. huge success in and led to windows transition.	From apple from 1984. began as a GUI. In 2005 the design & structure changed to intel x86 based architecture.	Finnish university released in 1991. NNU develops later. Integrated into Linux. It is open to consumer.
	Follows a directory structure to store files. has logical drives and cabinet files drives. It also has folders. folders like documents, photos, music, video. It has extension as .txt, .jpg etc.	In MAC hardisk lots of directories. The root directory of MAC may encounter when they visit their own MAC book.	Developed with different code base. It stores data in the form of a tree.

Also provides recycle bin where all deleted files can be stored.

Recycle bin can be configured to ↑ in size.

The file /bin and directories are in /Application, /bin, /sbin, /tmp, etc..

There is a single file and all your devices are mounted over this file.

Registry Windows registry is the main database used to store all settings.

Responsible to store all user information with its password and device related info.

The registry also has editor which allows you to view all keys and values or even delete if necessary.

Stores all application settings in a series of .plist files which have the various preference folder.

The plist file contains all properties in either plain text or binary format. Stored at /Library/preferences folder.

not has any specific registry of its own. All applications stored in on program basis under the different users in same binary format of files being stored. There is no centralized storage and no periodic cleaning.

interface Until windows 8 not interchangeable. Windows xp had some improvement but not much.

has facility to bridge virtual network interface. This can be done by going to /etc preferences & managing the interface.

easy to switch interface. There are GNOME and KDE desktops which help in catering to needs. They help in focusing on different aspects.

command terminal Black command prompt used to execute commands also called as windows command processor.

used to execute commands and different batch files also used for administrators to solve all windows issues.

Provides console as terminal application, command line, prompt & terminal.

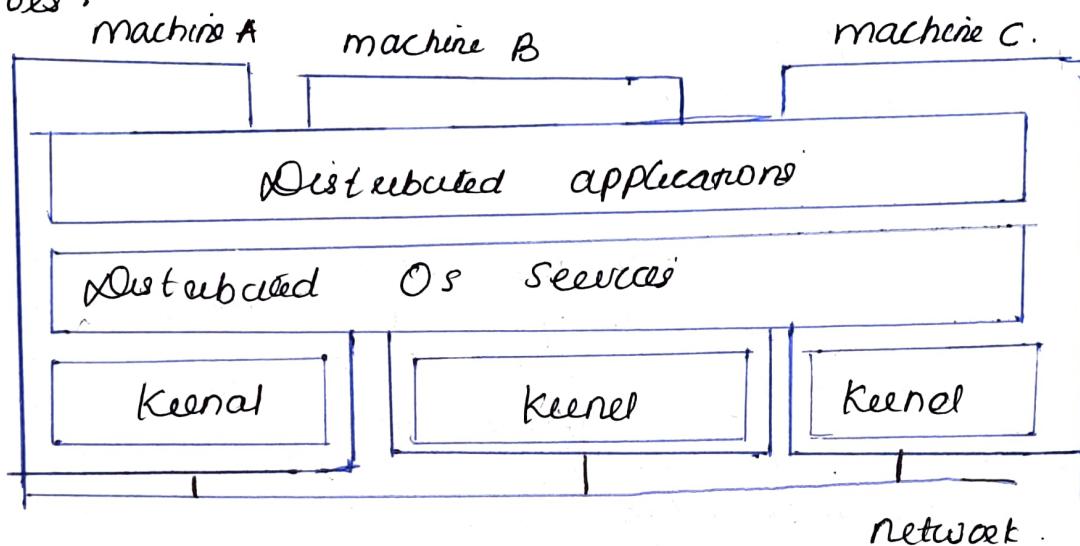
command line used to type command. prompt will provide you with some information and our command.

it will provide our

terminal at application → /bin or application → utilities. In addition also a shell prompt most common shell used in bash. It describes how the terminal will behave and look when it runs.

16) What is distributed OS. Explain in detail?

Multiple central processors are used by Distributed S/W to serve multiple real-time applications and multiple users. Accordingly data processing jobs are distributed among the processors.



processors communicate with each other through various communication lines (like high speed buses or telephone lines). These are known as loosely coupled systems or distributed systems. Processors in this system may vary in size and function. They are referred as site nodes, computers and so on.

Advantages

- With resource sharing facility, a user at one site may be able to use the resources available at another.
- speed up the exchange of data with one another via electronic mail
- Failure of one site in a distributed system doesn't affect the others, the remaining sites can potentially continue operating
- Better service to the customer
- Reduction of the load on the host computer
- Reduction of delay in data processing

(17) What is the difference between gets vs scanf vs fgets. Justify why gets is dangerous to use?

In

scanf (const char* format, ...)

scanf is a fn that read data with specified format from a given string stream source, originated from c program

* → optional starting asterisk indicate that the data is to be read from the stream but ignored.

width → maximum number of characters to be read.

modifiers → data pointed by corresponding additional argument.

type → character specify the type of data.

gets :-

char * gets (char * str) reads a line from stream and stores it into the string pointed to by str. It stops when either the new line character is read or when the EOF is reached whichever comes first.

```
Eg: int main() {  
    char str[50]  
    printf("Enter a string");  
    gets(str);
```

fgets

char * fgets (char * str, int n, FILE * stream).

str → pointer to an array of chars when stream read is stored

n → maximum num of characters to be read

stream → This is the pointer to FILE object that identifies the stream where characters are read from.

e.g.
fget(str, 60, fp);

~~get~~

gets() is dangerous because it is possible for the user to crash the program by typing too much into the prompt. It can't detect the end of available memory, so if you allocate an amount of memory too small for the purpose, it can cause a seg fault and crash.

Ques 18 With a neat example explain the handling of strings in C language?

String is 1D array of characters terminated by null character.

char greet[6] = {'H', 'e', 'l', 'l', 'o', '\0'};

Or else can be initialised as

char greeting[] = "Hello";

Index	0	1	2	3	4	5
-------	---	---	---	---	---	---

Variable	H	e	l	l	o	\0
----------	---	---	---	---	---	----

Address	0x2345	0x2346	0x2347	0x2348	0x2349	0x2350
---------	--------	--------	--------	--------	--------	--------

Strcat: This fn will append a copy of source string to the end of destination string.

The terminating character at the end of dest is replaced by the first character of source.

Eg: #include <iostream.h>.

```
int main()
{
    char dest[50] = "This is an ";
    char sou[50] = "example";
}
```

```
strcat(dest, sou);
```

```
printf("%s", dest);
```

```
return 0;
```

```
}
```

O/p: This is an example.

2. `strchr()` - C string is header file required. Returns a pointer to the last occurrence of a character in a string.

3. ~~strcpy~~ → copy a string

Assigning the values to the string

`char c[] = "abcd"`

`char c[5] = "abcd";`

`char c[] = {'a', 'b', 'c', 'd', '\0'}`

`strcpy` can be used to copy from one string.

2. Reading from the user.

1. `scanf()`.

It include `<stdio.h>`

`int main()`

{ `char name[20];`

`printf("Enter name");`

`scanf("%s", name);`

`printf("Your name is %s", name);`

return 0;

}

It will read a single word.

② `fgets()` & `put`.

To read a line of text.

Eg:

`int main()`

{ `char name[30];`

```
printf("Enter name");
fgets(name, size of (name), stdin);
puts(name)
Prints entire name.
```

Passing strings to fn

String can be passed to fn in similar way of arrays.

```
#include <stdio.h>
void display(char str[]);
int main()
{
    char str[50];
    printf("Enter string");
    fgets(str, size of (str), stdin);
    display(str);
    return 0
}.
```

```
void display (char str[])
{
    printf("String o/p");
    puts(str);
}
```

(19) Why length of string is +1 in fgets.

fgets() reads in atmost one less than size of character from stream and stores them in to the buffer pointed to s. Reading stops after an EOF or a new line.

A terminating null byte (as long) is stored after the last character in the buffer so it adds 1n after your 4 letters returning string length + 1

Q) what are the string functions in C. Give examples?

① strlen.

Size_t strlen(const char *str).

int main()

```
{ char str1[20] = "Beating";
  printf("Length of string str1: %d", strlen(str1));
  return 0;
}
```

Hence strlen(str) - returns 8

and size of (str) → returns 20 - array size is 20.

② strnlen.

Size_t strnlen(const *str, size_t maxlen).

int main()

```
{ printf("Length of str1 when maxlen is 30: %d", strnlen(str1, 30));
  printf("Length of str when maxlen is 5: %d", strnlen(str, 5));
}
```

O/P → length at 30 → 8 (prev point)
length of 5 → 5 program

③ strcmp

compare 2 strings.

strcmp(s1, s2).

④ strncmp
defined
compares the number of letters in 2 strings

`strcmp(s1, s2, 8)`

Compare 8 letters.

⑤ `strcat`

`strcat(s1, s2)`

concatenate s_1 and s_2 ,

⑥ `strncat`

`strcat(s1, s2, 3)`

concatenate the 3 words from s_2 to s_1 .

⑦ `strcpy`

`strcpy`.

copy the content of one string to another.

`strcpy(s1, s2)`.

⑧ `strchr`.

`strchr(my, 't');`

`int main()`

{ `my[20] = "nevarthy"`

~~printf("%s",~~ `strchr`

~~printf("%s",~~ `strchr(my, 't')`);

return 0;

}

O/P: `ithy`

⑨ `strchr` → this will point w/o the character.

⑩ `strstc`.

It will search of string term instead of single char.

`strstc(s1, "love")`

16 giving revathy venugopal at input, then the
string will be
∴ venugopal

```
#include <string.h>
```

```
int main()
```

```
{
```

```
char str[70] = "This is sample program";  
printf (" O/P is %s ", strstr(str, 'sam'));  
return 0;
```

```
}.
```

O/P is Sample program.