DRONACHARYA College of Engineering

LABORATORY MANUAL

B.Tech. Semester- IV

OPERATING SYSTEM LAB
Subject code: LC-CSE-212G

Prepared by:

Dr Manoj Kumar

Checked by:

Dr Ritu Pahwa

Approved by:

Name : Prof. (Dr.) Isha Malhotra

Sign.: …………………….

Sign.: …………………..

Sign.: …………………..

DEPARTMENT OF CSE(AI&ML)
DRONACHARYA COLLEGE OF ENGINEERING
KHENTAWAS, FARRUKH NAGAR, GURUGRAM (HARYANA)

Table of Contents

1. Vision and Mission of the Institute

2. Vision and Mission of the Department

3. Programme Educational Objectives (PEOs)

4. Programme Outcomes (POs)

5. Programme Specific Outcomes (PSOs)

6. University Syllabus

7. Course Outcomes (COs)

8. CO- PO and CO-PSO mapping

9. Course Overview

10. List of Experiments

11. DOs and DON'Ts

12. General Safety Precautions

13. Guidelines for students for report preparation

14. Lab assessment criteria

15. Lab Experiments

Vision and Mission of the Institute

Vision:

"To impart Quality Education, to give an enviable growth to seekers of learning, to groom them as World Class Engineers and managers competent to match the expending expectations of the Corporate World has been ever enlarging vision extending to new horizons of Dronacharya College of Engineering"

Mission:

M1: To prepare students for full and ethical participation in a diverse society and encourage lifelong learning by following the principle of 'Shiksha evam Sahayata' i.e., Education & Help.

M2: To impart high-quality education, knowledge and technology through rigorous academic programs, cutting-edge research, & Industry collaborations, with a focus on producing engineers& managers who are socially responsible, globally aware, & equipped to address complex challenges.

M3: Educate students in the best practices of the field as well as integrate the latest research into the academics.

M4: Provide quality learning experiences through effective classroom practices, innovative teaching practices and opportunities for meaningful interactions between students and faculty.

M5: To devise and implement programmes of education in technology that are relevant to the changing needs of society, in terms of breadth of diversity and depth of specialization.

Vision and Mission of the Department

Vision:

To cultivate skills and make proficient engineers cum trainers in the domain of Artificial Intelligence & Machine Learning for exceptional contributions to the society.

Mission:

M1: To impart intense training and learning to generate knowledge through the state-of-the-art concepts and technologies in Artificial Intelligence and Machine Learning.

M2: To establish centres of excellence by collaborating with the leading industries to exhilarate innovative research and development in AIML and its allied technology.

M3: To inculcate regenerative self-learning abilities, team spirit, and professional ethics among the students for noble cause.

Programme Educational Objectives (PEOs)

PEO1- ANALYTICAL SKILLS:

Using a solid foundation in mathematical, scientific, engineering, and current computing principles, formulate, analyse, and resolve engineering issues in real-world domain.

PEO2- TECHNICAL SKILLS:

Apply artificial intelligence theory and concepts to analyse the requirements, realise technical specifications, and design engineering solutions.

PEO3- SOFT SKILLS:

Through inter-disciplinary projects and a variety of professional activities, demonstrate technical proficiency, AI competency, and foster collaborative learning and a sense of teamwork.

PEO4- PROFESSIONAL ETHICS:

Excel as socially responsible engineers or entrepreneurs with high moral and ethical standards, competence, and soft skills that will enable them to contribute to societal demands and achieve sustainable advancement in emerging computer technologies.

PROGRAM OUTCOMES (POs)

PO1: Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2: Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3: Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7: Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9: Individual and teamwork: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

P11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

P12: Life-long learning: Recognize the need for and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES (PSOs)

PSO1: **Fundamentals and critical knowledge of the Computer System:** Apply the knowledge gained pertaining to build, asses, and analyze the software and hardware aspects of the program to solve real world business problems.

PSO2: **Comprehensive and applicative knowledge of Software Development:** Ability to evaluate and apply knowledge of data engineering, methodologies, and able to plan, develop, test, analyze, and manage required aspects in heterogenous platforms individually or in team work.

PSO3: **Applications in Computing Domain:** Ability to acquire computational knowledge and project development abilities using novel tools and methodologies to tackle challenges in the fields related to Deep Learning, Machine learning, Artificial Intelligence.

PSO4: **Applications in Innovations and Research:** Capacity to direct a team or firm that develops products and to use the knowledge learned to recognise actual research issues

Operating System Lab (LC-CSE-212G)

University Syllabus

Course code	LC-CSE-212G				
Category	Lab Course				
Course title	Operating System Lab				
Scheme and Credits	L	T	P	Credits	Semester = 4
	0	0	4	2	
Classwork	25 Marks				
Exam	25 Marks				
Total	50 Marks				
Duration of Exam	03 Hours				

1. Introduction to UNIX File System.

2. File and Directory Related Commands in UNIX.

3. Essential UNIX Commands for working in UNIX environment.

4. I/O Redirection and Piping

5. Introduction to VI Editors.

6. Introduction of Processes in UNIX

7. Communication in UNIX and AWK.

8. Introduction of the concept of Shell Scripting.

9. Decision and Iterative Statements in Shell Scripting.

10. Writing the Shall Scripts for unknown problems

Course Outcomes (COs)

Upon successful completion of the course, the students will be able to:

CO1: Understand the structure and architectural components of UNIX Operating System to analyze and design the problem. Moreover, students would be able to know the Basic Introduction of UNIX Operating System.

CO2: Basic Introduction of UNIX Commands that are used for operating the UNIX.

CO3: Introduction of Shell Scripting and VI Editor.so that the students get familiar with writing the UNIX scripts in UNIX editor.

CO4: Students will establish themselves as effective professionals by solving real problems with UNIX Shell Scripting knowledge and with attention to teamwork, critical thinking and problem-solving skills by Writing Shell Scrips of unknown problems

CO-PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1		1	1						1	1		
CO2		1	1						1	1		
CO3	2	2	2	2					2	2	2	2
CO4	3	3	3	3	3				3	3	3	3

CO-PSO Mapping

CO	PSO1	PSO2	PSO3	PSO4
CO1	1	1		
CO2	2	2		
CO3	2	2	3	2
CO4	3	3	3	3

*3-HIGH
*2-MEDIUM
*1-LOW

Course Overview

Operating System Lab is a course that teaches students about the fundamentals of operating systems. The lab sessions are designed to give students hands-on experience with operating system concepts, such as process management, memory management, file systems, and inter-process communication.

The topics covered in Operating System Lab include:

* Process management

* Memory management

* File systems

* Inter-process communication

* Scheduling

* Virtual memory

* Distributed operating systems

List of Experiments mapped with COs

Program No	List of Programs
1.	Study of BASIC UNIX COMMANDS
2.	Study of BASIC DOS COMMANDS
3.	PROGRAM TO FIND THE LARGEST AMONG THREE NUMBERS.
4.	PROGRAM TO FIND THE ADDITION OF TWO NUMBER'S
5.	OGRAM TO PRINT THE NUMBERS FROM 5 TO 1
6.	OGRAM TO PRINT THE CURRENT DATE.
7.	OGRAM TO DISPLAY THE BIODATA.
8.	OGRAM FOR SYSTEM CALLS OF UNIX OPERATING SYSTEMS (OPENDIR, READDIR, CLOSEDIR)
9.	OGRAM FOR SYSTEM CALLS OF UNIX OPERATING SYSTEM
10.	WRITE A C PROGRAM FOR IMPLEMENTATION OF FCFS AND SJF SCHEDULING ALGORITHMS.

DOs and DON'Ts

DOs

1. Login-on with your username and password.

2. Log off the Computer every time when you leave the Lab.

3. Arrange your chair properly when you are leaving the lab.

4. Put your bags in the designated area.

5. Ask permission to print.

DON'Ts

1. Do not share your username and password.

2. Do not remove or disconnect cables or hardware parts.

3. Do not personalize the computer setting.

4. Do not run programs that continue to execute after you log off.

5. Do not download or install any programs, games or music on computer in Lab.

6. Personal Internet use chat room for Instant Messaging (IM) and Sites is strictly prohibited.

7. No Internet gaming activities allowed.

8. Tea, Coffee, Water & Eatables are not allowed in the Computer Lab.

General Safety Precautions

Precautions (In case of Injury or Electric Shock)

1. To break the victim with live electric source, use an insulator such as fire wood or plastic to break the contact. Do not touch the victim with bare hands to avoid the risk of electrifying yourself.

2. Unplug the risk of faulty equipment. If main circuit breaker is accessible, turn the circuit off.

3. If the victim is unconscious, start resuscitation immediately, use your hands to press the chest in and out to continue breathing function. Use mouth-to-mouth resuscitation if necessary.

4. Immediately call medical emergency and security. Remember! Time is critical; be best.

Precautions (In case of Fire)

1. Turn the equipment off. If power switch is not immediately accessible, take plug off.

2. If fire continues, try to curb the fire, if possible, by using the fire extinguisher or by covering it with a heavy cloth if possible, isolate the burning equipment from the other surrounding equipment.

3. Sound the fire alarm by activating the nearest alarm switch located in the hallway.

4. Call security and emergency department immediately:

Emergency : 201 (Reception)
Security: 231 (Gate No.1)

Guidelines to students for report preparation

All students are required to maintain a record of the experiments conducted by them. Guidelines for its preparation are as follows: -

1) All files must contain a title page followed by an index page. ***The files will not be signed by the faculty without an entry in the index page.***

2) Student's Name, roll number and date of conduction of experiment must be written on all pages.

3) For each experiment, the record must contain the following

     (i) Aim/Objective of the experiment

     (ii) Pre-experiment work (as given by the faculty)

     (iii) Lab assignment questions and their solutions

     (iv) Test Cases (if applicable to the course)

     (v) Results/ output

Note:

1. Students must bring their lab record along with them whenever they come for the lab.

2. Students must ensure that their lab record is regularly evaluated.

Lab Assessment Criteria

An estimated 10 lab classes are conducted in a semester for each lab course. These lab classes are assessed continuously. Each lab experiment is evaluated based on 5 assessment criteria as shown in following table. Assessed performance in each experiment is used to compute CO attainment as well as internal marks in the lab course.

Grading Criteria	Exemplary (4)	Competent (3)	Needs Improvement (2)	Poor (1)
AC1: **Pre-Lab written work (this may be assessed through viva)**	Complete procedure with underlined concept is properly written	Underlined concept is written but procedure is incomplete	Not able to write concept and procedure	Underlined concept is not clearly understood
AC2: **Program Writing/ Modeling**	Unable to understand the reason for errors/ bugs even after they are explicitly pointed out	Assigned problem is properly analyzed, correct solution designed, appropriate language constructs/ tools are applied	Assigned problem is properly analyzed & correct solution designed	Assigned problem is properly analyzed
AC3: **Identification & Removal of errors/ bugs**	Able to identify errors/ bugs and remove them	Able to identify errors/ bugs and remove them with little bit of guidance	Is dependent totally on someone for identification of errors/ bugs and their removal	Unable to understand the reason for errors/ bugs even after they are explicitly pointed out
AC4: **Execution & Demonstration**	All variants of input /output are tested, Solution is well demonstrated and implemented concept is clearly explained	All variants of input /output are not tested, However, solution is well demonstrated and implemented concept is clearly explained	Only few variants of input /output are tested, Solution is well demonstrated but implemented concept is not clearly explained	Solution is not well demonstrated and implemented concept is not clearly explained
AC5: **Lab Record Assessment**	All assigned problems are well recorded with objective, design constructs and solution along with Performance analysis using all variants of input and output	More than 70 % of the assigned problems are well recorded with objective, design contracts and solution along with Performance analysis is done with all variants of input and output	Less than 70 % of the assigned problems are well recorded with objective, design contracts and solution along with Performance analysis is done with all variants of input and output	

LAB EXPERIMENTS

PROGRAM NO. 1

Study of BASIC UNIX COMMANDS

Date command:

Unix maintains a system clock. As for now you can simply display the current date with the date command, which shows the

date and time for the nearest second as shown:

$ date: Thur oct 4 11:23:52 IST 1999

$ date +%m : This command shows current month(in digits) [eg. 08]

$ date +%h : This command shows current month(in words) [eg. Aug]

Cal command:

This command is used for printing the calendar of any particular month or the entire year.

$ cal 2000: This command will show the calender of 2000.

$ Cal 09 2007- This command will display the calender of September,2007

The head command:

The head command is used to display the top few records of the file. The syntax of the

command is as follows:

$ head -5 filename

This command will display the top 5 lines of filename.

The tail command:

The tail command displays the end few records of the file. If no line count is given, the tail command displays the last ten lines of the file.

$ tail –3 emp.lst

This command will display the last 3 lines of emp.lst .

To create a file:

$ cat >seema

is a good girl

To save a file:

ctrl+d

To display a file:

$ cat seema

is a good girl

ls:

 This command is used to list all the files or directories.

$ ls – x

This command will display all the files column wise.

$ ls –a

This command will display all the hidden files

To copy one file into another:

$ cp dog jam

 it will copy the content of dog file into jam file.

To rename a file:

 $ mv   fish whale

it will rename a file name fish to file name whale.

To remove a file:

$ rm whale

it remove the file whale.

To show the current directory in which you are working:

$ pwd

To clear the screen:

$ tput clear

To make a directory:

$ mkdir school

This command will create a directory name school.

$ mkdir apple grapes banana

This command will create all the above directories in one go.

To remove a directory:

$ rmdir school

To use the calculator:

$ bc

12+5

This command will display the result 17.

12+5;12-5;12*5

This command will display the result 17 7 60.

Scale=2

14/3

here Scale is used to display the result upto 2 decimal places.

To come out from the UNIX:

$ exit

PROGRAM NO-2

Study of BASIC DOS COMMANDS

1.   File commands.

(a) Copy

C:\>copy source-path_name destination-path_name

This command will copy source file into destination file.

(b) Delete

C:\>del abc

This command will delete the abc file.

(c) Rename

C:\>ren abc xyz

This command will rename the file name abc to file name xyz.

(d) Type

C:\>type xyz

This command will display the contents of file xyz.

2.   Directory command

(a) DIR

This command is used to list the files under a directory.

C:\>Dir

This command will show all directories

Dir\w :- width wise

Dir\p :- page file

(b) CHDIR

This command changes working directory to directory you specify.

C:\> cd e:

This command changes C directory to E.

E:\>

C:\>cd..

This command make the parent directory the root directory

C:\> cd\

This command makes the root directory as working directory.

(c) MKDIR (MD)

This command is used to create a sub-directory.

C:\> md java

This command will create a directory java.

3. MISCELLANEOUS COMMANDS

(a) DATE

C:\> Date

To display current date and enter the new date if you want to change

(b)Time

C:\>Time

To display the time currently and enter new time.

(c) VERSION

C:\>ver

This command displays the version of MS- DOS you are working upon.

(d) CLEAR SCREEN

C:\>cls

This command clears the screen.

(e) PATH

C:\>path c:\>java\bin   //This command will set the above path.

PROGRAM NO-3

PROGRAM TO FIND THE LARGEST AMONG THREE NUMBERS.

echo largest among three numbers

echo Enter number1

read a

echo Enter number2

read b

echo Enter number3

read c

if [$a -gt $b] && [$a -gt $c]

then

echo $a is greater

elif [$b -gt $a] && [$b -gt $c]

then echo $b is big

else echo $c is big

fi

~

~

"large" 33L, 268C   12,1   All

OUTPUT

[user9@localhost user9]$ sh large

largest among three numbers

Enter number1

34

Enter number2

73

Enter number3

44

73 is big

[user9@localhost user9]$

PROGRAM NO-4

PROGRAM TO FIND THE ADDITION OF TWO NUMBER'S

echo addition of two numbers

echo Enter num1

read a

echo Enter num2

read b

c=`expr $a + $b`

echo addition is $c

~

~

~

~

~

"add" 7L, 113C   6,1   All

OUTPUT

[user9@localhost user9]$ sh add

addition of two numbers

Enter num1

4

Enter num2

5

addition is 9

[user9@localhost user9]$

PROGRAM NO-5

PROGRAM TO PRINT TNE NUMBERS FROM 5 TO 1

i=5

echo sequence is

while [$i != 0]

do

  echo $i

  i=`expr $i - 1`

done

~

~

~

"print" 8L, 80C   8,1   All

OUTPUT

[user9@localhost user9]$ sh print

sequence is

5

4

3

2

1

[user9@localhost user9]$

PROGRAM NO-6

PROGRAM TO PRINT THE CURRENT DATE.

echo current date is `date`

echo user is `who am i`

echo current dir `pwd`

~

~

~

~

"date" 4L, 76C   3,1   All

OUTPUT

[user9@localhost user9]$ sh date

current date is Wed Nov 14 11:33:58 IST 2007

user is localhost.localdomain!user9 pts/3 Nov 14 11:24 (192.168.1.49)

current dir /home/user9

[user9@localhost user9]$

PROGRAM NO-7

PROGRAM TO DISPLAY THE BIODATA.

echo "Enter Your name"

read a

echo "Enter your age"

read b

echo "Enter your father's name"

read c

echo "Enter your mother's name"

read d

echo "Enter your address"

read e

echo your name is $a

echo your age is $b

echo your father's name is $c

echo your mother's name is $d

echo your address is $e

~

~

~

"biodata" 15L, 297C   15,1   All

OUTPUT

biodata" 15L, 297C written

[user9@localhost user9]$ sh biodata

Enter Your name

Seema

Enter your age

20

Enter your father's name

Mr.Satyapal yadav

Enter your mother's name

Mrs.Sushila yadav

Enter your address

Delhi

your name is Seema

your age is 20

your fathers name is Mr.Satyapal yadav

echo your mothers name is Mrs. Sushila yadav

your address is Delhi

[user9@localhost user9]$

PROGRAM NO-8

PROGRAM FOR SYSTEM CALLS OF UNIX OPERATING SYSTEMS (OPENDIR, READDIR, CLOSEDIR)

```
#include<stdio.h>

#include<dirent.h>

struct dirent *dptr;

int main(int argc, char *argv[])

{

char buff[100]; DIR

*dirp;

printf("\n\n ENTER DIRECTORY NAME"); scanf("%s", buff); if((dirp=opendir(buff))==NULL)

{

printf("The given directory does not exist");

exit(1); }

while(dptr=readdir(dirp))

{

printf("%s\n",dptr->d_name);

}

closedir(dirp);

}
```

OUTPUT

```
ENTER DIRECTORY NAME
file.txt
The given directory does not exist

...Program finished with exit code 0
Press ENTER to exit console.
```

PROGRAM NO-9

PROGRAM FOR SYSTEM CALLS OF UNIX OPERATING SYSTEM (fork, getpid, exit)

```c
#include<stdio.h>
#include<unistd.h>
#include<stdlib.h>
#include<sys/types.h>
int main() {  int pid, pid1, pid2; pid = fork();
if (pid == -1) {
printf("ERROR IN PROCESS CREATION \n");
exit(1);  }  if
(pid != 0) {
pid1 = getpid();
printf("\n the parent process ID is %d\n", pid1);
} else {    pid2
= getpid();
printf("\n the child process ID is %d\n", pid2);
}
return 0;
}
```

OUTPUT:

```
the parent process ID is 3974

the child process ID is 3978


...Program finished with exit code 0
Press ENTER to exit console.
```

PROGRAM NO-10

To write a C program for implementation of FCFS scheduling algorithm.

```c
#include<stdio.h>
#include<stdlib.h>
struct fcfs
{ int
pid; int
btime;
int
wtime;
int ttime;
} p[10];int main() { int i,n;
 int
 towtwtime=0,totttime=0;
 printf("\n  fcfs
 scheduling...\n");
 printf("enter the no of
 process"); scanf("%d",&n);
 for(i=0;i<n;i++)
 {
 p[i].pid=1;
 printf("\n burst time of the process");
 scanf("%d",&p[i].btime);
 }
p[0].wtime=0;
p[0].ttime=p[0].btim
e;
totttime+=p[i].ttime;
for(i=0;i<n;i++)
{
p[i].wtime=p[i-1].wtime+p[i-
1].btim
p[i].ttime=p[i].wtime+p[i].btime;
totttime+=p[i].ttime;
towtwtime+=p[i].wtime;
}
 for(i=0;i<n;i++)
 {{
 printf("\n waiting time for process"); printf("\n turn around time for process"); printf("\n"); }}
 printf("\n total waiting time :%d", totwtime ); printf("\n average waiting
 time:%f",(float)totwtime/n); printf("\n total turn around time
 :%d",totttime); printf("\n average turn around time:
 :%f",(float)totttime/n); }
```

OUTPUT

```
 fcfs scheduling...
enter the no of process2

 burst time of the process2

 burst time of the process1

 waiting time for process
 turn around time for process

 waiting time for process
 turn around time for process

 total waiting time :2
 average waiting time :1.000000
 total turn around time :5
 average turn around time: :2.500000

...Program finished with exit code 0
Press ENTER to exit console.
```

Viva Voce Questions with Answers

1) What is an operating system?

The operating system is a software program that facilitates computer hardware to communicate and operate with the computer software. It is the most important part of a computer system without it computer is just like a box.

2) What is the main purpose of an operating system?

There are two main purposes of an operating system:

It is designed to make sure that a computer system performs well by managing its computational activities.

It provides an environment for the development and execution of programs.

3) What are the different operating systems?

- o Batched operating systems
- o Distributed operating systems
- o Timesharing operating systems
- o Multi-programmed operating systems
- o Real-time operating systems

4) What is a socket?

A socket is used to make connection between two applications. Endpoints of the connection are called socket.

5) What is a real-time system?

Real-time system is used in the case when rigid-time requirements have been placed on the operation of a processor. It contains a well defined and fixed time constraints.

6) What is kernel?

Kernel is the core and most important part of a computer operating system which provides basic services for all parts of the OS.

7) What is monolithic kernel?

A monolithic kernel is a kernel which includes all operating system code is in single executable image.

8) What do you mean by a process?

An executing program is known as process. There are two types of processes:

- o   Operating System Processes
- o   User Processes

9) What are the different states of a process?

A list of different states of process:

- o   New Process
- o   Running Process
- o   Waiting Process
- o   Ready Process
- o   Terminated Process

10) What is the difference between micro kernel and macro kernel?

Micro kernel: micro kernel is the kernel which runs minimal performance affecting services for operating system. In micro kernel operating system all other operations are performed by processor.

Macro Kernel: Macro Kernel is a combination of micro and monolithic kernel.

11) What is the concept of reentrancy?

It is a very useful memory saving technique that is used for multi-programmed time sharing systems. It provides functionality that multiple users can share a single copy of program during the same period.

It has two key aspects:

- o   The program code cannot modify itself.
- o   The local data for each user process must be stored separately.

12) What is the difference between process and program?

A program while running or executing is known as a process.

13) What is the use of paging in operating system?

Paging is used to solve the external fragmentation problem in operating system. This technique ensures that the data you need is available as quickly as possible.

14) What is the concept of demand paging?

Demand paging specifies that if an area of memory is not currently being used, it is swapped to disk to make room for an application's need.

15) What is the advantage of a multiprocessor system?

As many as processors are increased, you will get the considerable increment in throughput. It is cost effective also because they can share resources. So, the overall reliability increases.

16) What is virtual memory?

Virtual memory is a very useful memory management technique which enables processes to execute outside of memory. This technique is especially used when an executing program cannot fit in the physical memory.

17) What is thrashing?

Thrashing is a phenomenon in virtual memory scheme when the processor spends most of its time in swapping pages, rather than executing instructions.

18) What are the four necessary and sufficient conditions behind the deadlock?

These are the 4 conditions:

1) **Mutual Exclusion Condition:** It specifies that the resources involved are non-sharable.

2) **Hold and Wait Condition:** It specifies that there must be a process that is holding a resource already allocated to it while waiting for additional resource that are currently being held by other processes.

3) **No-Preemptive Condition:** Resources cannot be taken away while they are being used by processes.

4) **Circular Wait Condition:** It is an explanation of the second condition. It specifies that the processes in the system form a circular list or a chain where each process in the chain is waiting for a resource held by next process in the chain.

19) What is a thread?

A thread is a basic unit of CPU utilization. It consists of a thread ID, program counter, register set and a stack.

20) What is FCFS?

FCFS stands for First Come, First Served. It is a type of scheduling algorithm. In this scheme, if a process requests the CPU first, it is allocated to the CPU first. Its implementation is managed by a FIFO queue.

21) What is SMP?

SMP stands for Symmetric MultiProcessing. It is the most common type of multiple processor system. In SMP, each processor runs an identical copy of the operating system, and these copies communicate with one another when required.

22) What is RAID? What are the different RAID levels?

RAID stands for Redundant Array of Independent Disks. It is used to store the same data redundantly to improve the overall performance.

Following are the different RAID levels:

RAID 0 - Stripped Disk Array without fault tolerance

RAID 1 - Mirroring and duplexing

RAID 2 - Memory-style error-correcting codes

RAID 3 - Bit-interleaved Parity

RAID 4 - Block-interleaved Parity

RAID 5 - Block-interleaved distributed Parity

RAID 6 - P+Q Redundancy

23) What is deadlock? Explain.

Deadlock is a specific situation or condition where two processes are waiting for each other to complete so that they can start. But this situation causes hang for both of them.

24) Which are the necessary conditions to achieve a deadlock?

There are 4 necessary conditions to achieve a deadlock:

- o **Mutual Exclusion:** At least one resource must be held in a non-sharable mode. If any other process requests this resource, then that process must wait for the resource to be released.

- o **Hold and Wait:** A process must be simultaneously holding at least one resource and waiting for at least one resource that is currently being held by some other process.

- o **No preemption:** Once a process is holding a resource (i.e. once its request has been granted), then that resource cannot be taken away from that process until the process voluntarily releases it.

- o **Circular Wait:** A set of processes { P0, P1, P2, . . ., PN } must exist such that every P[i] is waiting for P[(i + 1) % (N + 1)].

25) What is Banker's algorithm?

Banker's algorithm is used to avoid deadlock. It is the one of deadlock-avoidance method. It is named as Banker's algorithm on the banking system where bank never allocates available cash in such a manner that it can no longer satisfy the requirements of all of its customers.

26) What is the difference between logical address space and physical address space?

Logical address space specifies the address that is generated by CPU. On the other hand physical address space specifies the address that is seen by the memory unit.

27) What is fragmentation?

Fragmentation is a phenomenon of memory wastage. It reduces the capacity and performance because space is used inefficiently.

28) How many types of fragmentation occur in Operating System?

There are two types of fragmentation:

- o **Internal fragmentation:** It is occurred when we deal with the systems that have fixed size allocation units.
- o **External fragmentation:** It is occurred when we deal with systems that have variable-size allocation units.

29) What is spooling?

Spooling is a process in which data is temporarily gathered to be used and executed by a device, program or the system. It is associated with printing. When different applications send output to the printer at the same time, spooling keeps these all jobs into a disk file and queues them accordingly to the printer.

30) What is the difference between internal commands and external commands?

Internal commands are the built-in part of the operating system while external commands are the separate file programs that are stored in a separate folder or directory.