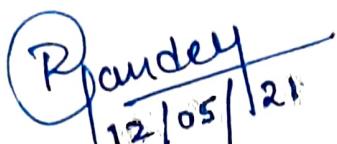


## **Declaration and statement of authorship**

I, bearing Registration Number 106119100, agree and acknowledge that:

1. The assessment was answered by me as per the instructions applicable to each assessment, and that I have not resorted to any unfair means to deliberately improve my performance.
2. I have neither impersonated anyone, nor have I been impersonated by any person for the purpose of assessments.

Signature of the Student :



Rajneesh  
12/05/21

Full Name : RAJNEESH PANDEY

Roll No. : 106119100

Sub Code : CSPC43

Mobile No. : 8290968008

### **Instructions:**

**Do not include this in the declaration**

1. Either print the declaration or Write in hand on a separate sheet of paper with
  - ✓ Write your Registration Number (Roll No.)
  - ✓ Sign against Signature of the Student with date
  - ✓ Full Name (in Capital Letters)
  - ✓ Roll Number
  - ✓ Sub Code
  - ✓ Mobile Number
2. Scan the document and save it in PDF format
3. Upload along with the Answer Sheet as first page.
4. Without this declaration, the answer sheet will not be evaluated.

12/05/2021

Rajneesh Pandey

END - SemesterQuestion 1

1) A) Multithreading allows ~~multiple~~ to execution of multiple programs at the same time.

parts → thread and are lightweight process available within the process.

Hence multithreading leads to maximum utilization of resources.

\* Resource sharing:

All the threads of the process share its resources such as memory, data files etc. So, single app. have diff. threads.

\* Responsiveness

Program responsiveness allows to run a program to run even if part of it is blocked using multithreading

\* Utilization of Multiprocessor Architectures

Each thread can run on a different processor in parallel using multithreading. This increases the concurrency of the system

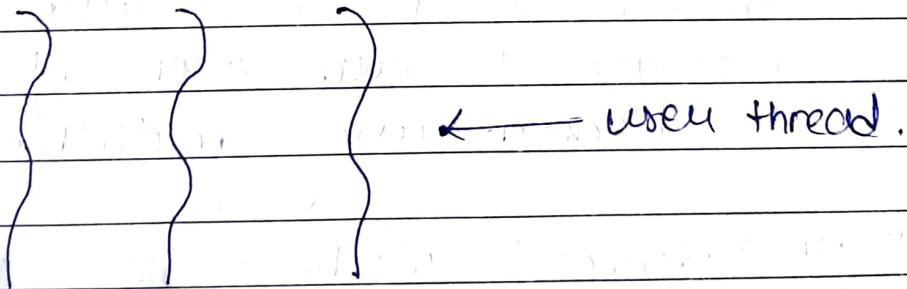
\* Economy

They are more economical to use threads as they share the process resources.

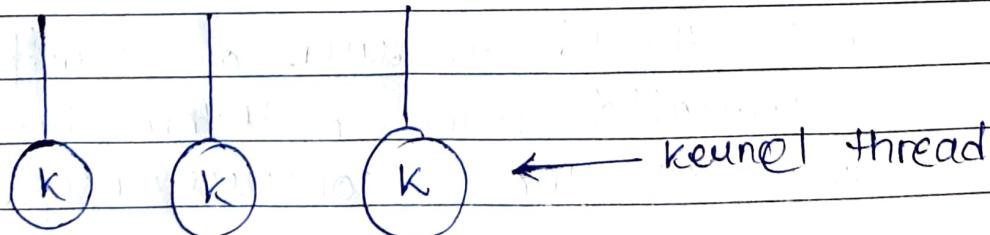
## # Ways to establish relationship between User and kernel threads:

### 1) One-to-One :

- The one-to-one model creates a separate kernel thread to handle each and every user thread.
- Most implementations of this model place a limit on how many threads can be created.
- This model provides more concurrency than that of many-to-one Model.
- Disadv. of this is that creation of a user thread requires a corresponding kernel thread.

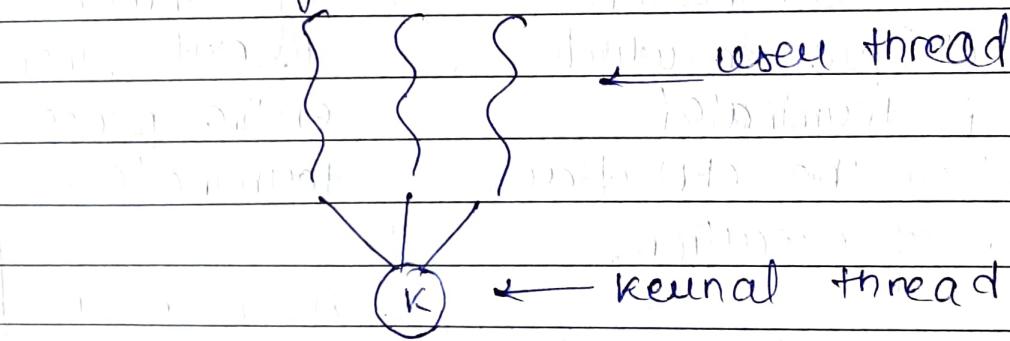


One-to-one model.



## 2) Many-to-One:

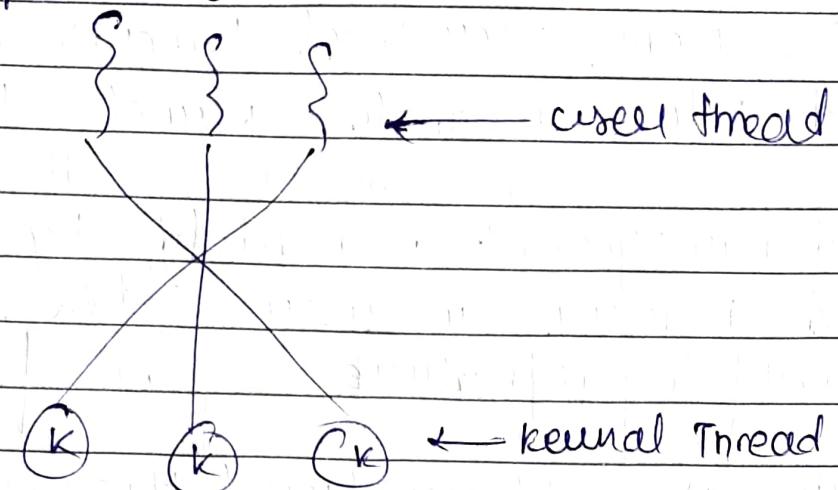
- In this, many user-level threads are all mapped onto a single kernel thread.
- Thread management is handled by the thread library in user space, which is efficient in nature.
- In this case, if user-level thread libraries are implemented in the operating sys in some way that the system does not support them, then user kernel threads use this many-to-one model of relationship.



## 3) Many-to-many:

- This model multiplexes any number of user threads onto an equal or smaller number of kernel threads, combining the best features of the one-to-one and many-to-one models.
- User can create any number of threads.
- Blocking the kernel system will not block the entire process.

→ Process can be split across multiple processes.



### Question

1) B)

### Process Termination v/s Thread termination

Process termination is the technique in which process is terminated and releases the CPU after completion of execution.

ways for process termination

1) By protection error

2) By parent request

3) By parent Termination

4) By Time over Run.

5) By Arithmetic error

If thread call exit, then entire process terminates.

1) A single thread can exit in three ways

1) simply return from the start routine.

return value is thread exit code

2) A thread can be cancelled by another thread in the process

- due to  
 6) memory requirement      8) A thread can exit  
                                   as pthread\_exit

## Question 2

(i) Gantt chart: SJF arrival time = 0.

P <sub>2</sub>	P <sub>4</sub>	P <sub>1</sub>	P <sub>3</sub>	
0	4	9	19	26

Turnaround time

for the process

$$P_2 = 4 - 0 = 4, \quad P_4 = 9 - 0 = 9$$

$$P_1 = 17 - 0 = 17, \quad P_3 = 26 - 0 = 26$$

$$\text{Avg. Turnaround time} = \frac{9+4+17+26}{4} = 14.$$

(ii) SJF: job arrival time is different

Gantt chart

P <sub>1</sub>	P <sub>2</sub>	P <sub>4</sub>	P <sub>1</sub>	P <sub>3</sub>	
0	1	5	10	17	26

Turnaround time

for the process

$$P_4 = 10 - 3 = 7$$

$$P_1 = 17 - 0 = 17, \quad P_2 = 5 - 1 = 4.$$

$$P_3 = 26 - 2 = 24$$

$$\text{avg. turnaround time} = \frac{17+4+7+24}{4} = \frac{52}{4} = 13$$

(iii) SJF : when one completes diff arrival time

Gantt chart

	P <sub>1</sub>	P <sub>2</sub>	P <sub>4</sub>	P <sub>3</sub>	
	0	8	12	17	26

Turnaround time

of process.

$$P_1 = 8 - 0 = 8 \quad P_2 = 12 - 1 = 11$$

$$P_3 = 26 - 2 = 24, \quad P_4 = 17 - 3 = 14$$

Avg turnaround time

$$\text{time} = \frac{8+11+24+14}{4} = \frac{57}{4} = 14.25$$

Question 3

3) A) Techniques used for structuring the Page tables:

1) Hierarchical Paging

Also known as multilevel Paging



- There can be chance where page table is too big to fit in a contiguous space, so we may have hierarchy with several levels.
- Logical address space is broke up into multiple pages table.
- This is the simplest technique for this purpose, Two-level page & three level page is used

### Two level page Table

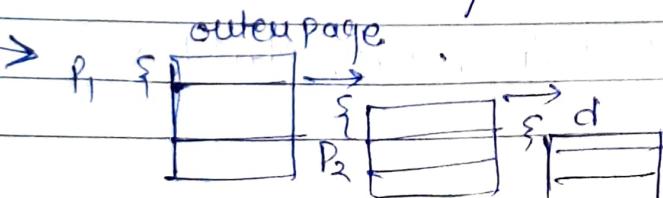
32-bit logical address space  
± KB page size

Page Num. → 22 bits → 12 bits PgNum.  
Page offset → 10 bits → 10 bits Page offset

#### logical address

Page number	Page offset
P <sub>1</sub>   P <sub>2</sub>	d

Index of outer page  
displacement within the page innerpage



### Three level page table

64-bit logical address

4KB page size.

To avoid large table, divide outer page table, and it will result in

Three page table level.

and outer page	outer	inner	offset
P <sub>1</sub>   P <sub>2</sub>	P <sub>1</sub>   P <sub>2</sub>	P <sub>2</sub>	d

32      10      10      12

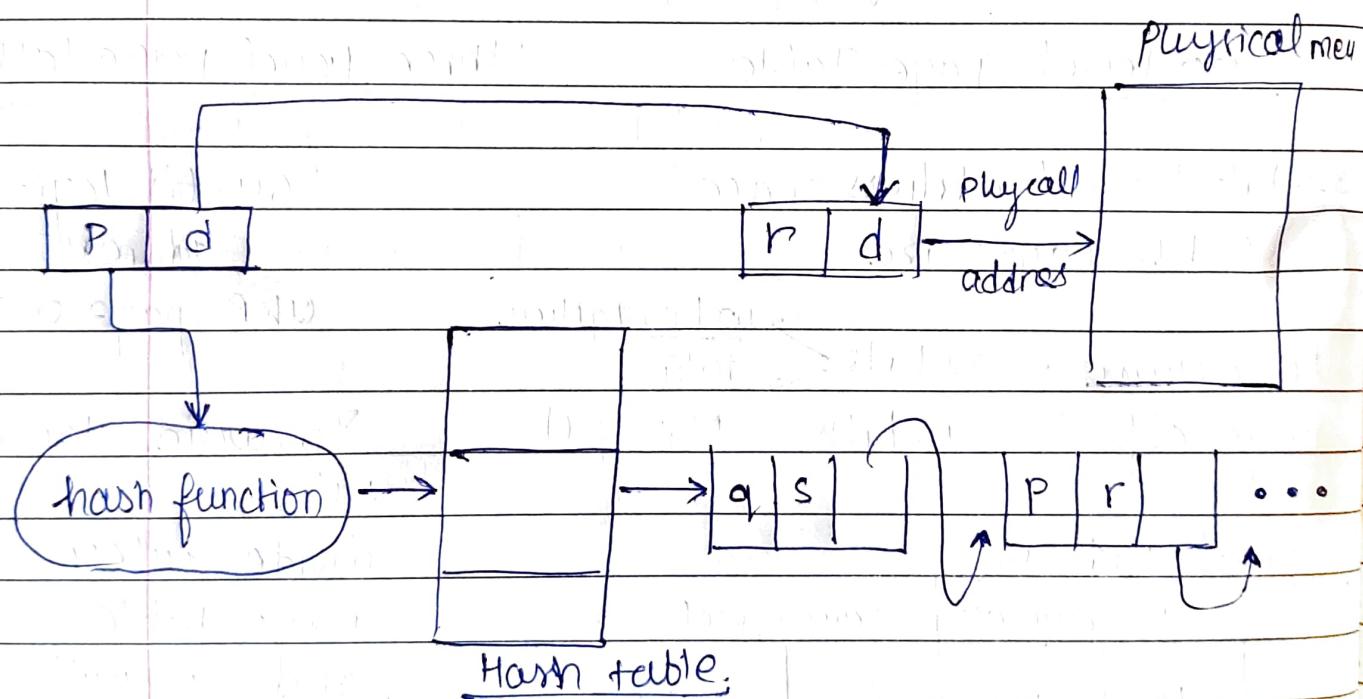


## Hashed Page Table

This technique used to handle address spaces larger than 32 bits.

→ In this virtual page number is hashed into page table.

→ This page table mainly contains a chain of elements hashing to the same elements.



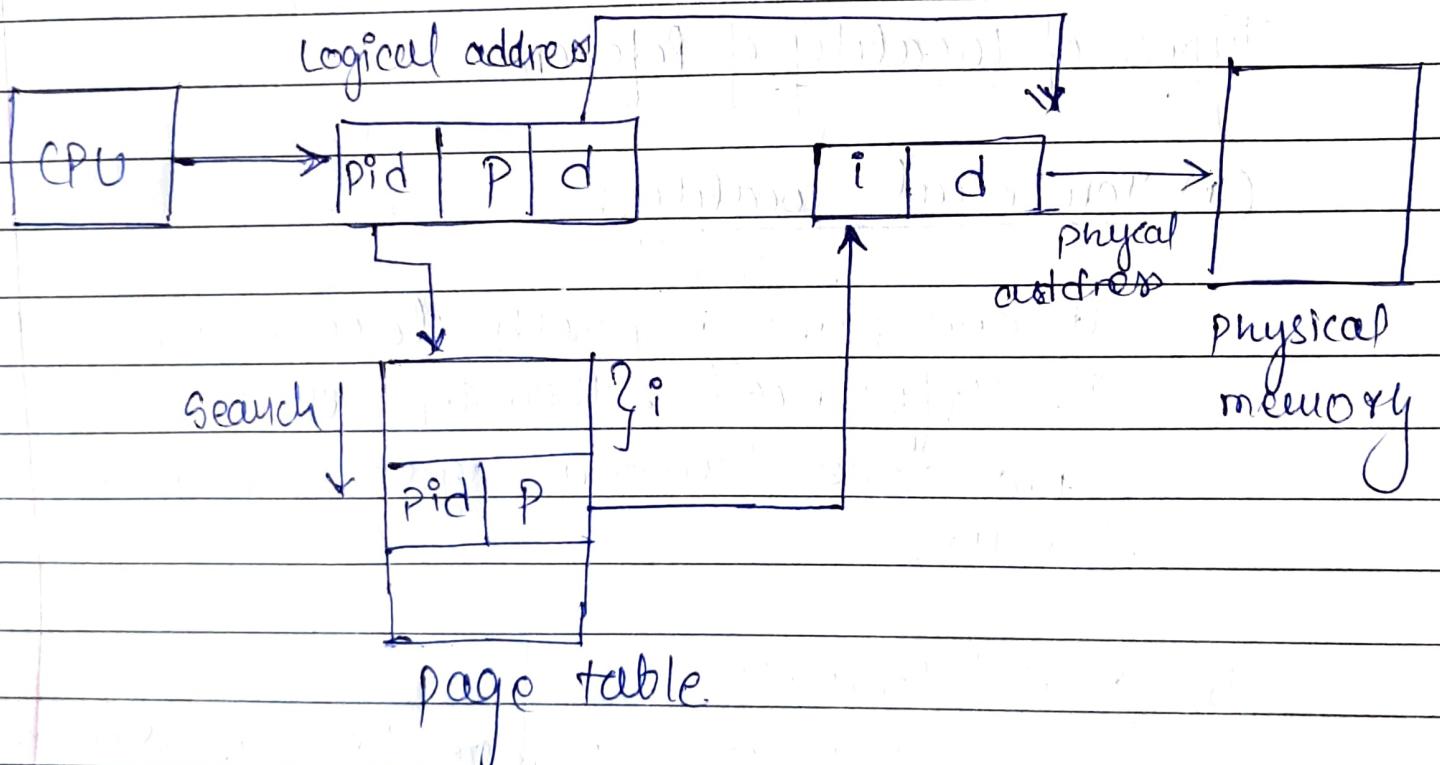
The virtual page numbers are compared in this chain. Scrolling for a match, if the match is found then corresponding physical frame is extracted.

Variant of 64-bit address space commonly uses clustered page tables.

### 3) Inverted Page Table

combines a Page table and A frame table into single data structures

- There is one entry for each virtual page number of each page of memory
- Decreases memory used, but increase the time that needed to search.
- Entry stored virtual address of page



keep track of process id of each entry.

Question 3

3) B)

Locality of Reference

when a computer program tends to access same set of memory location for a particular time period.

OR

its the tendency of the computer program to access instruction whose address are near one another

Types of Locality of Reference(a) Temporal Locality:

at one point a particular memory located is referenced, then its likely to reference the same location again in near future.

(b) Spatial Locality:

if a particular storage location referenced the nearby memory location will be referenced.

(c) Brand Locality

few possible alternative for path in spatial, temporal, coordinate space.

(d) equidistant Locality

Halfway between spatial & branch locality



In walking set algorithm,

walking set window tell how many page reference are there in a locality.

If all page reference are there in memory then there is no page fault else page fault occurred.

Ex.

Page reference = 3, 2, 3, 4, 6, 2, 4, 1, 6, 7, 4, 3, 3, 5, 4, 1

lets set window = 4

So,

4 locality of page reference tables are there. when process execute. it change one to another locality & no frames needed.

In first no. of frame needed = 3

In 2nd = 4

and in last 4, frames needed

Question 4

4) A) Role of Access Matrix is that it is the security model for protection state of computer system.  
It is represented as matrix.

It defines the right of each process executing in the domains with respect to the each object.

Method to Implement Access Matrix:

a) Capabilities:

Row wise decomposition of the access matrix. Each subject is assigned with a list of tuples ( $O, M[s, o]$ ) for all objects "o" that allows to access. These tuples are called Capabilities.

A subject is allowed to access any object for which it holds capabilities.

Capability Format

Object Descriptor	Right of the sub read, write, execute
-------------------	--

### b) Access control list

- column wise decomposition of access control.

object contains tuples like  $(s, M[\alpha])$   
for all subject access Object.

→ system searches the access control list of object to find out if an entry  $(s, \alpha)$  exists for subject.

→ If required access is permitted, the request is executed else an appropriate exception is raised

Subject	Access rights
ravi	Read, write
rana	Read
rayneesh	execute

### c) Lock & key method:

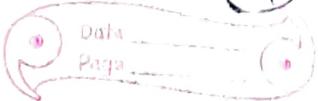
It's an hybrid of the access control list and capabilities method.

When subject make the request to access object  $o$  in mode  $\alpha$ , the system executes in following manner

→ System locate tuple  $(o, \text{key})$  to the subject  
If id not matched, access denied.

→ otherwise, the access granted.

→ Then control list of object  $o$  is lock for  $\alpha \in \alpha$ .



## Question (4)

4) B)

(i) → RAID is the technique used to improve reliability and performance in mass storage system.

RAID system consist of two or more drives working in parallel.

(ii) → Also, we can create a cooling system that prevent from the failure of the drives.

(iii) → Designing to prevent vibration,

A data storage system needs an anti-vibration design for optimal performance. When hard drive are placed nearby, that cause disrupt.

(iv) → Enabling Active Updates also helps for the Reliability for the mass storage.

(v) → We also can improve Reliability via Redundancy:

The more disk a system has, the greater the likelihood that one

of them will go bad at any given time.

Hence increasing disk on system actually decreases the Mean Time to failures.

(ii) Also, we can improve via Parallelism:

There is also a performance benefit to mirroring, particularly with respect to reads, since every block of data is duplicated on multiple disk, read operation can be satisfied from any available copy and multiple disk can reading different data block simultaneously in parallel.

Question 5

5) A)

Distributed system :

It's a system with multiple components located on different machine that communicate and coordinate actions in order to appear as a single coherent system to the end-user.

Difference b/w Distributed system & NetworksDistributed system

→ The existence of multiple autonomous computer in a computer network is transparent to the user.

→ The operating system automatically allocates jobs to processor, moves files among various computer without explicit user intervention.

Networks

→ A computer network is an interconnected collection of autonomous computer

able to exchange information.

→ A computer network, correctly requires user to explicitly

login onto one machine, explicitly submit job remote & explicitly move file/data around the network.

## features of Distributed Operating System.

### (a) Transparency

Hide the fact that process and resources are physically distributed across multiple computers.

Concept of transparency can be applied to many aspects of distributed OS.

### (b) Connecting User and Resources

Easy for user to access remote resources and to share them with other users in a controlled manner.

### (c) Openness

Offer service in standards that describes syntax and semantics of those service instances.

### (d) Scalability :

trend in-distributed system is towards larger system. This observation has implication for distributed file system design.

### (e) Reliability :

Reliability of the overall system can be boolean or component reliability.

### (f) Performance

The performance is better as it's flexible, reliable.

Question 5 :

5) B)

Not possible to implement centralized control in distributed system due to problem imposed by architecture itself of the system.

so,

decentralized control is used that is all control functions are distributed among various sites of the system.

→ But in distributed system due to the absence of global shared memory it's not possible to have upto date of process and resources.

→ Also, Due to the absence of global physical clock, scheduling is also difficult to implement. scheduling in the distributed systems