1a) What is kernel of an operating system? Explain the state transition diagram of process 3.5.

Kernel of OS :- The kernel is a computer program at the core of a computer's operating system and generally has complete control over everything in the system. It is the portion of the operating system code that is always resident in memory and facilitates interactions between hardware & ~~cof~~ software components. It is an integral part of an operating system.

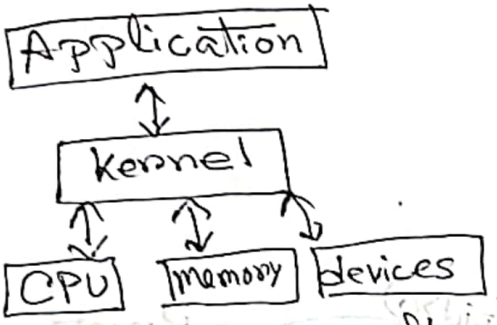


Figure: location of kernel

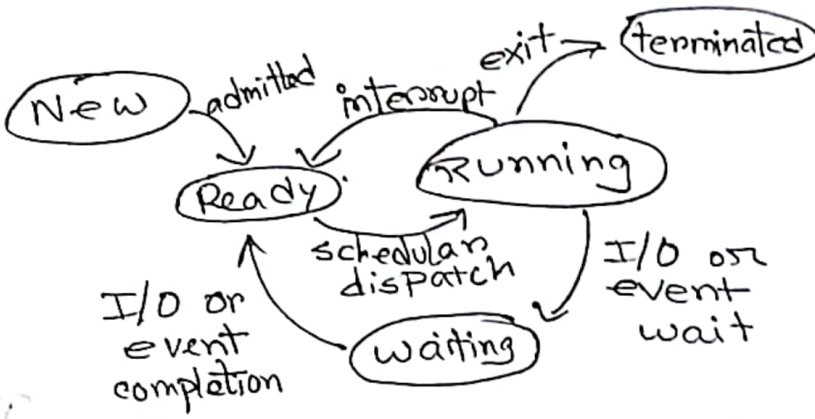


Diagram of Process State.

As a process executes, it changes state.

New : The process is ~~beig~~ being created.

Running: Instructions are ~~beig~~ being executed.

Waiting:- The process ~~was~~ waiting for some event to occur.

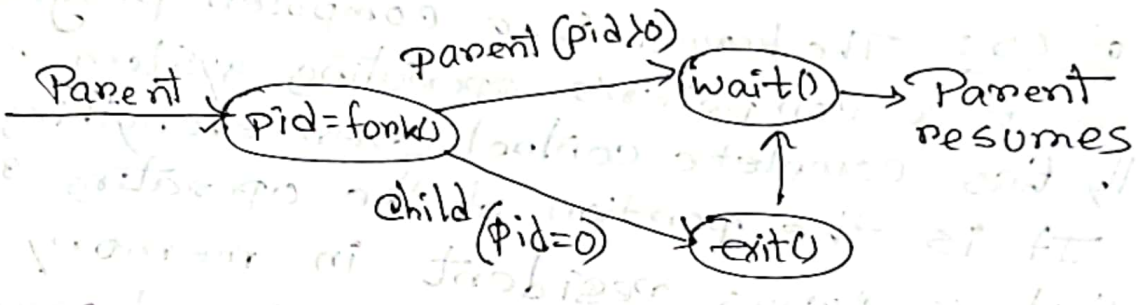
Ready :- The process is waiting to be assigned to a processor.

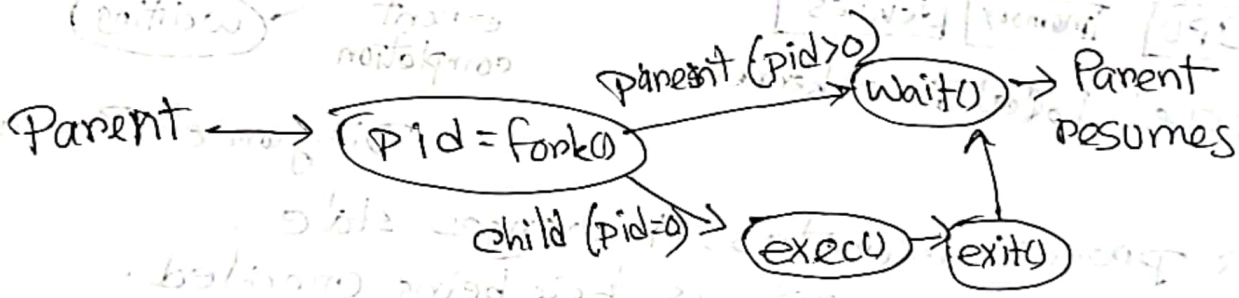
Terminated:- The process has finished execution

1 (a) In the UNIX environment what happens in the following three cases as shown in fig 1.

5.25

Soln:



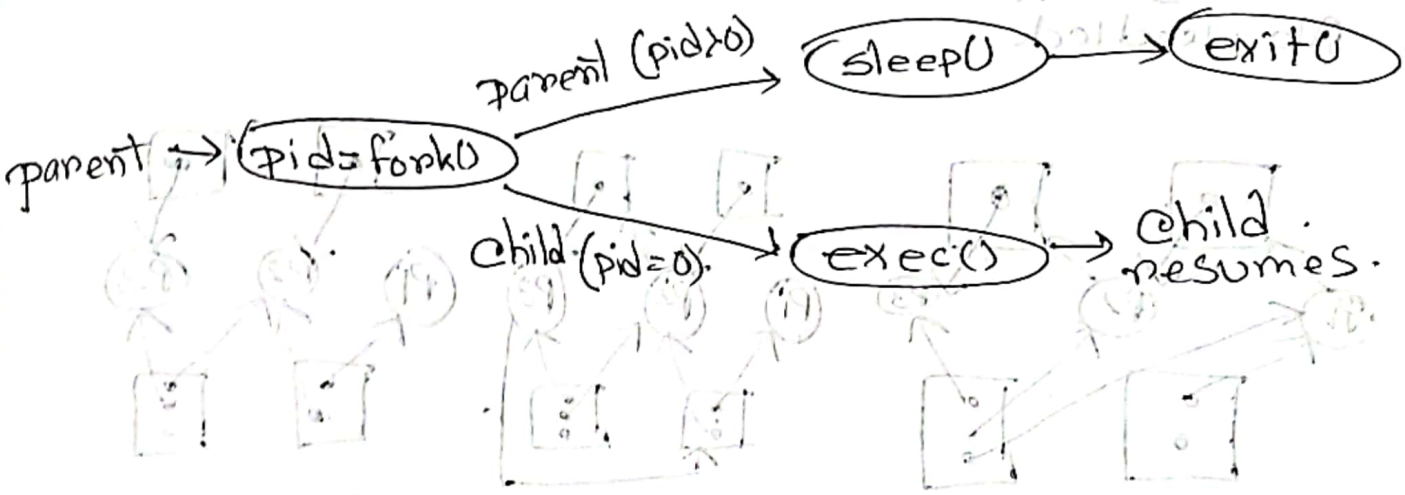


MAHI STORE
01915-085088

CamScanner

SAT SUN MON TUE WED THU FRI

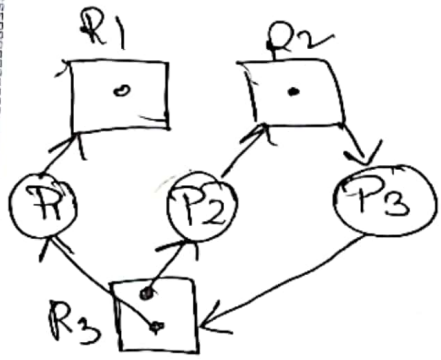
Date :.....



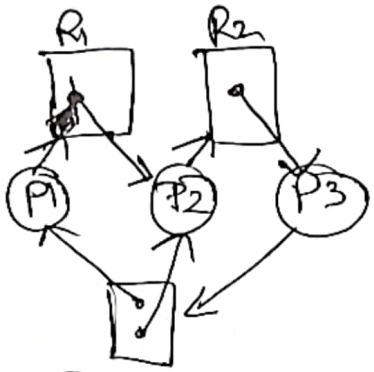
SAT SUN MON TUE WED THU

Date :

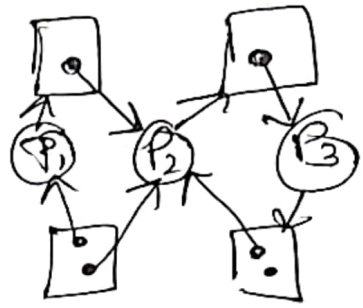
2(a) Check deadlock or not.



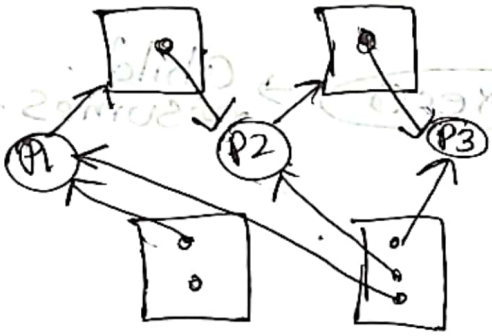
$P_1 \rightarrow P_3 \rightarrow P_2$
No deadlock



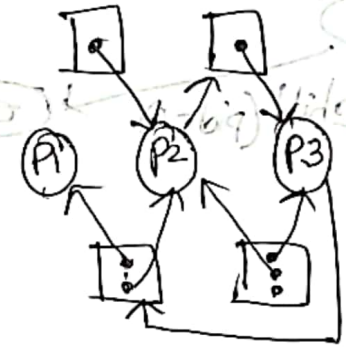
Deadlock



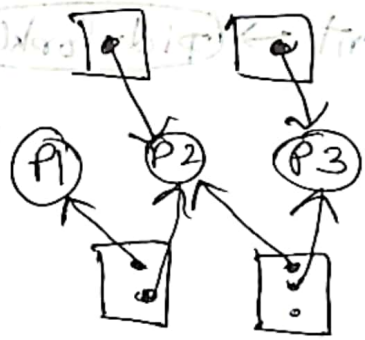
$P_3 \rightarrow P_2 \rightarrow P_1$



$P_3 \rightarrow P_2 \rightarrow P_1$



$P_1 \rightarrow P_3 \rightarrow P_2$



$P_1 \rightarrow P_2 \rightarrow P_3$

(b) As process executes, it changes state. How many processes can be in the New, Running, Waiting, Ready and terminated state in a system having six cores per socket and two thread per core? why??

MAHI STORE
01915-085088

CamScanner

SAT SUN MON TUE WED THU FRI

Date :

SAT SUN MON TUE WED THU FRI

Date :

2) When processor is switched from one process to another process in a system with a single process: (4)

i) Why does an operating system need to access process control block?

ii) Mark the place in figure when P_0, P_1 and OS are in the idle state

iii) Will the process be even in the idle state? why?

iv) Is it true that the idle time of the user process will increase if no of user process increases? why?

i) Ans: Process Control block contains all the necessary informations of a process.

Process state: Running, ready etc

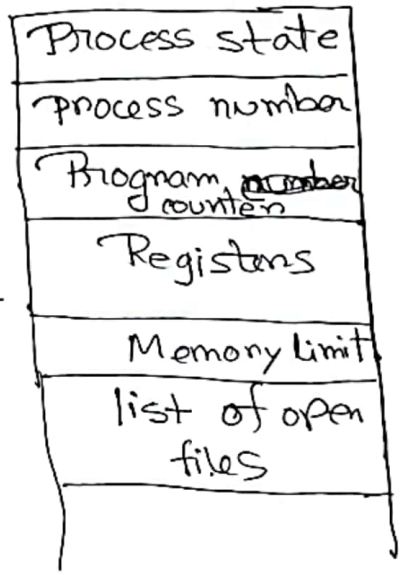
Process counter: location of instruction for next execute

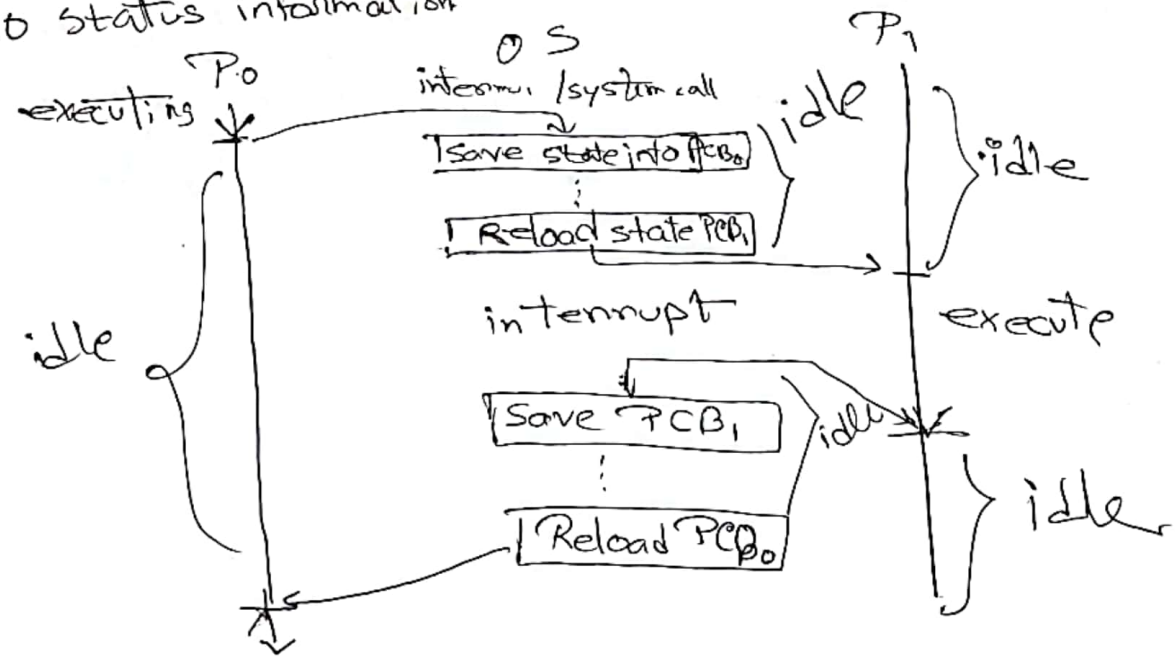
CPU register - contents of all process-centric registers.

CPU scheduling info

Memory info.

I/O status information





SAT SUN MON TUE WED THU FRI

Date

iii) Yes, P_0 will remain idle while P_1 executes and vice-versa

iv) Yes, it is generally true that idle time of a user process may increase if the number of users processes increases, depending on specific system & workload. ~~Bec~~ Because OS must prioritize ~~system and workload~~ resources such as CPU time & memory among all running processes.

With the increase of no. of user processes running on system must schedule & switch between them more frequently. This can cause some processes to experience idle time while they wait for OS to allocate the necessary resources.

And if there are too many processes competing for resources, the OS may need to use techniques like time-sharing and/or preemption to allocate resources fairly among all processes. This can result in some process experiencing longer idle time.

2a) How can a multi-threaded process be ~~defi~~ benefited by a a multi-core system?

Ans:- The processes can be run in parallel by the Operating system

MAHI STORE 01975-065088

CamScanner

52

SAT SUN MON TUE WED THU FRI

Date :

3. (b) Can a multi-threaded process with a global variable be in trouble? If yes, how can we solve it?

Ans: Yes because multiple threads can access and modify the same global variable concurrently. This can lead to race condition and other synchronization issues.

To solve this problem, –

1. Use thread-safe synchronization mechanism. Allow only one thread to access the global variable at a time by using synchronization mechanism like locks, semaphores. This mechanism can help to prevent multiple threads from concurrently modifying the same variable and causing synchronization issues.

2. Use thread local storage: Here each thread has its own copy of variable that are independent of other threads. This can help to prevent race condition and synchronization issues.

MAHI STORE
01915-085088

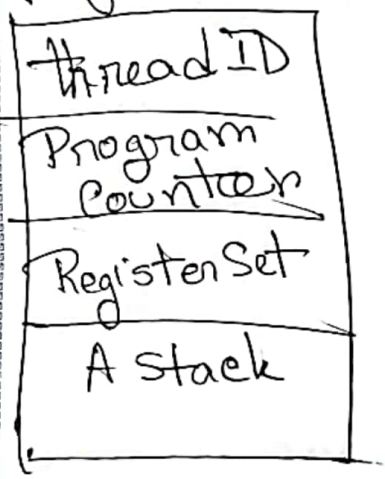
CamScanner

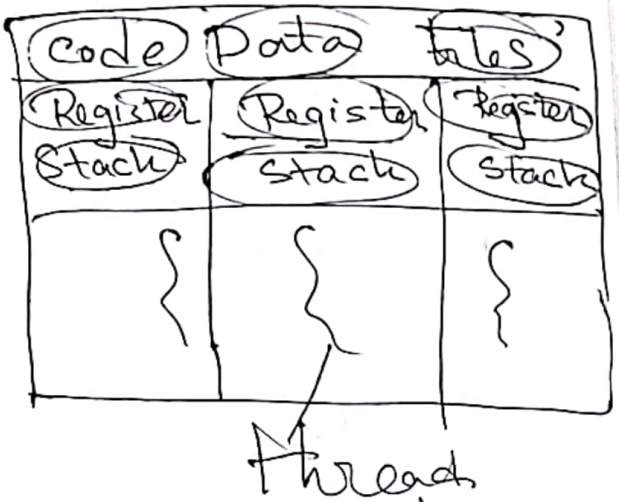
SAT SUN MON TUE WED THU FRI

Date :

Q What is thread? What you know about fork() & exec system call in multithreaded program?

Thread is a basic ~~sum~~ unit of execution within a process. A process can have one or more threads, and each thread has it's own stack and program counter, but shares memory and other resources with other threads in the same process. Threads are used to achieve concurrency in programs, allowing multiple tasks to be performed at a time.





Fork();- The system call that is used to create a separate, duplicate process of the calling process which is called parent process. The new created process only has a different id. The duplicate process is called child process.

exec();- When an exec() system call is invocked, the program specified in the parameter to exec() will replace the entire process — including all threads

MAHI STORE 01915-085088

CamScanner

SAT SUN MON TUE WED THU FRI

Date :

Replace a process with another process as we are not creating any new but replacing existing on. So, the process id remains same.

2020 3 (a)

MAHI STORE
01915-085088

CamScanner

SAT SUN MON TUE WED THU FRI

Date :

4 (a) Which kind of CPU scheduling algorithm would you like to be implemented in your operating system? why?

2.75

Ans. The choise of cpu scheduling algorithm depends on the requirement of the operating system and the workloads it's expected to handle.

If ~~fairness~~ fairness is a priority, Round Robin may be good choice, if low latency is important, Priority Scheduling may be more appropriate. It is also possible to combine multiple algorithm to achieve the desired trade-offs.

So, for implementing in OS i think i would prefer Priority Scheduling. As then I will get the task which is ~~crucial~~ crucial to complete done first.

There are many other scheduling algorithms like - SJF, LRTF.

(b) How can an operating system avoid 'starvation' of a user process by implementing fair process scheduling policies.

Starvation occurs when a process is never allocated CPU time despite being eligible for execution. this happen when OS always prioritizes certain process over others or if a process is continually preempted by other higher-priority processes.

The OS must ensure that lower-priority processes also gets CPU time and not entirely starved of CPU time

MAHI STORE 01915-085088

CamScanner

SAT SUN MON TUE WED THU FRI

Date :

4 (a) How can two process communicate with each other if there is an parent-child relation betn 2 process
• no parent-child relationship exist betn 2 process.

Ans:- Interprocess communication mechanism allow process to ~~continue~~ communicate with each other. If there is a parent-child relationship between processes, communication can be established using mechanism such as pipes, sockets, shared memory, and signals.

If there is no relationship between processes communication ca be established using mechanism such as message queue and semaphores. The specific IPC mechanism used depends on the type of communication required and the relationship betn the process.

Pipes-Communication between two related process. This is a half-duplex method, so the first process communicates with the second process. However to achieve full duplex, another pipe is needed.

MAHI STORE
01915-035088

CamScanner

SAT SUN MON TUE WED THU FRI

Date :

FIFO:- Communication between unrelated processes, FIFO is a full duplex, meaning the first process can communicate with the second and vice versa.

Message Queues: Communication betn two or more processes with full duplex capacity. The processes ~~will~~ will communicate with ~~the second process and vice versa at the same time~~ each other by posting a message and retrieving it out of the queue. Once retrieved, the message is no longer available in the queue.

Shared Memory: Communication btn 2 or more process ~~with full duplex capacity. The process will communicat with each other by posting a message and~~ is achieved through a shared piece of memory among all processes. The shared memory needs to be protected from each other by synchronizing access to all the process.

Semaphores – Means for synchronizing access to multiple processes. When one process wants to access the memory (for reading / writing) it needs to be locked (or protected) and released when the access is removed. This needs to be repeated by all the processes to secure data.

MAHI STORE
01915-085088

CamScanner

,8

SAT SUN MON TUE WED THU FRI

Date :

Signals :- Methods to communicate betn multiple processes by way of signaling. The source process will send a signal which is recognized by number and the destination will handle it.

1 (d) Briefly discuss message passing techniques
3 of inter process communication.

Ans: (Ans) Message passing :-

The cooperating processes can communicate with the help of a message passing facility. There are different ways communication

- Betn same threads of a process.
- Betn processes of same node or computer
- Btn two processes on different nodes or computer

Example : Chat system over computer.

A message passing system has atleast two primitive operations :-

1. Send (message)
2. Receive (message).

The message could be of a fixed or variable size. If system uses fixed size messages, the

MAHI STORE
01915-085088

CamScanner

system level implementation is easy. If chosen variable size messages, the system level implementation is difficult, but the programming level task is easy.

If P and Q process wants to communicate, then a communication link must exist betn them. There are several ways to implement physical link (shared memory, hardware, or network). But logical implementation of link provides different ways to communicate using message passing:-

1. Direct and Indirect Communication
2. Synchronous or asynchronous communication
3. Automatic or explicit buffering.

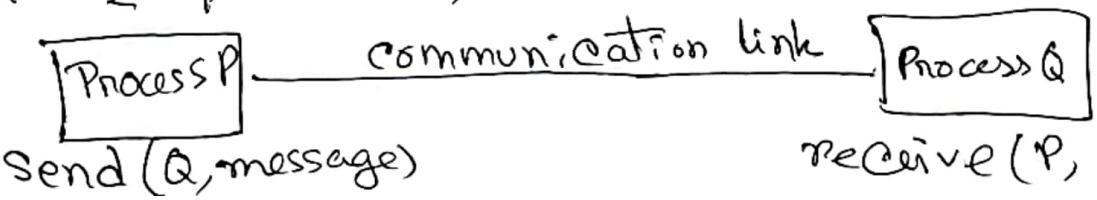
1. Direct Communication: Must explicitly name the sender or receiver in the communication.

The send() and receive() are defined as:

send (P, message) - send a message to process P.

receive (Q, message) - receive a message from process Q

- The link is established automatically. Processes need each other's identity to send messages.
- A link is associated with exactly two processes
- Betn 2 processes, there is only one link



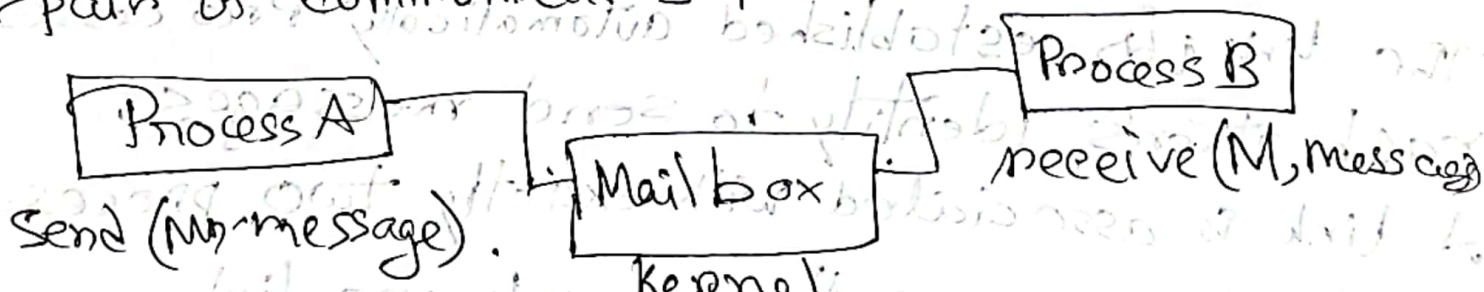
Indirect communication:- Message are send and received from mailbox or ports. The processes can replace messages into a mailbox or remove messages from them. The mailbox has a unique identification. Two processes can communicate if they have a shared mailbox.

Send (A, message) – send a message to mailbox A

receive (A, message) – receive a message from mailbox A

The communication scheme has the following properties –

1. A link may be associate with more than two processes.
2. A link is established between a pair of processes only if both have some shared mailbox.
3. There may be different links, with each link corresponding to one mailbox, between pair of communicating process.



MAHI STORE
01915-085088

CamScanner

Synchronous & Asynchronous Communication:- Using send() & receive(). Message passing may be blocking or non-blocking aka sync. & asyn.

- Blocking Send - Sending is blocked, until a message is received by receiving process or mailbox.

- Non-blocking send - sending process sends the message and resumes operation.

- Blocking receive - Receiver blocks until a message is available

- Non-blocking receive - the receiver retrieves either a valid message or a null

Automatic and Explicit Buffering

the messages exchanged between communicating processes resides is a temporary queue. The queue can be implemented in 3 way:

1. Zero capacity:- With zero capacity the link cannot have messages waiting. Blocking send is the correct option. It is no buffering

2. Bounded capacity:- the queue is of finite length. If queue not full, sender can continue to send messages. If full, block send.

3. Unbounded capacity - Infinite queue length, any number of messages wait and sender never blocks.

62

Section B

5 (a) What are the pros and cons of the following two ways of allocating disk space for a file named 'Cat.jpg'?

An: First figure shows Indexed block allocation & 2nd is linked allocation

~~Linked~~ Indexed allocation –

Pros: (1) There is no external fragmentation

(2) There is no size-declaration problems of contiguous allocation.

(3) There allows direct access ie. there is a list of all of the file location at once.

~~(4) There is no file limit.~~

~~(5) Dynamic file size - Allocate Allows for dynamic file size which means that files can grow or shrink as needed.~~ (4) Faster Access: Allows faster access to files since it uses an index to locate the blocks on disk.

Cons: ~~(1) Fragmentation:~~

(5) Reduce the number of blocks that are scattered across the disk.

(6) Improved data reliability: Indexed allocation is more reliable since it uses

MAHI STORE 01915-085088

CamScanner

a centralized index to track the location of the files, reducing the number making it easier to recover in case of disk failure.

Cons of Index Allocation:-

① When the index block is filled with index number of the file then an overflow block is taken to its index is added into the primary index block

② Index allocation does suffer from wastes space

③ In case in which we have a file of only one or two blocks. With linked allocation, we lose the space of only one pointer per block. With indexed allocation, we lose an entire block must be allocated even if only one or two pointers will be non-null.

Linked Allocation: each file is a linked list of blocks that can be scattered.

Pros- ① No file size limit ② No external fragmentation

③ Easier to find space for new file allocation.

④ Efficient use of space: Linked allocation can effectively utilize the available space in the disk

⑤ Easy to implement and maintain since only information required is the pointer to the next block

MAHI STORE
01915-085088

CamScanner

(5) Dynamic file size: Allows the file to grow or shrink as needed.

Cons:- (1) Pointer takes up space but by clustering internal fragmentation occurs.

(2) No random access possible. Linked allocation can be slower since each block needs to be traversed to find the desired block

(3) Vulnerable to data loss: If a pointer is lost or corrupted, the data in the blocks become inaccessible

(b) How large the index block should be?

5.

Ans) Each file in indexed alocation has it's own block, which is an array of disk block addresses. The ith entry in the index block points to the ith block of the file. The directory contains the address of the index block.

Generally index block sizes between 512 bytes and 4KB are commonly used in modern file system.

MAHI STORE 01915-085058

CamScanner

However the optimal index block size for a specific file depends on a variety of factors, including the expected file size, the performance characteristics of the underlying storage h/w,

5(c) First-fit, best-fit & worst-fit strategies are used to select a free hole from set of available hole. Write pros & cons.

2

	Pro	con
(7 hole) First-fit:	Fast Allocation	Poor space utilization & fragmentation (both internal & external)
(Best fit: smallest hole difference)	Improved Space utilization	Increased allocation time (delay) and potential fragmentation. (Slower)
Worst fit:	Fast Allocation, Reduced Fragmentation	Poor space utilization

MAHI STORE
01915-085088

CamScanner

6.(a) Define virtual memory. What are the advantages of it?

3 Virtual Memory is a technique that allows the execution of processes that are not completely in memory.

One major advantage of this scheme is that programs can be larger than physical memory

or Virtual memory abstracts main memory into an extremely large, uniform array of storage, separating logical memory as viewed by the user from physical memory. This technique frees programmers from the concerns of memory storage limitation.

It works by temporarily transferring inactive parts of a program or data from the computer's RAM to the hard-disk. This frees up space in RAM for currently active programs, allowing the computer to run more programs simultaneously that it could otherwise

MAHI STORE
01915-085088

CamScanner

1. Virtual memory helps to gain speed when only a particular segment of the program is required for the execution of the program.

2. Helpful in implementing multiprogramming environment.

3. It allows to run more applications at once.

4. It helps to fit many large programs into smaller programs.

(1) Increased usable Memory: V.M. allows a Computer to use more memory than it physically has available, making it possible to run more programs or larger programs.

(2) Efficient use of memory: Virtual memory allows the computer to allocate memory to program on demand, only allocating the amount of memory that is actually needed at a given time this leads to efficient use of memory resource and reduces memory wasteage.

(4) Better multitasking: With virtual memory, a computer can run multiple programs simultaneously even if they require more memory than the computer has physical available. This improves computer's multitasking ability and makes it produ

MAHI STORE
01915-085088

CamScanner

4. Improved performance: Virtual memory can improve overall system performance by reducing the need to access the hard disk for memory, as the computer can use the RAM more efficiently. This results in faster processing times and less waiting for program to load.

5. Memory protection: Virtual memory allows for memory protection, where each program is allocated its own virtual address space that is isolated from other programs. This prevents one program from accessing or modifying the memory space of another program, increasing system stability and security.

MAHI STORE
01915-085088

CamScanner

6 (b) String: 1, 2, 3, 4, 2, 1, 5, 6.

LRU

	1	2	3	4	2	1	5	6
	1	1	1	4	4	4	5	5
1		2	2	2	$\boxed{2}$	2	2	6
2			3	3	3	1	1	1
3					*			

hit = 1. P.F = 8 − 1 = 7

FIFO

	1	2	3	4	2	1	5	6
1	1	1	1	4	4	4	4	6
2		2	2	2	2	1	1	1
3			3	3	3	3	5	5
					*			

hit = 1 P.F = 8 − 1 = 7

Optimal Replacement

	1	2	3	4	2	1	5	6
1	1	1	1	1	1	1	5	6
2		2	2	2	2	2	2	2
3			3	4	4	4	4	4
					*	*		

hit = 2. P.F = 8 − 2 = 6

6 (c) Say that four program named PA, PB, PC, PD are written so that when they will execute -

- Process A will creat 3 named pipes (P1, P2, P3)
- Pro B will write to Pipe 1
- Pro C " " " Pipe 2 at first then read from Pipe 2
- Process D will write to P3 at first then read from P2

MAHI STORE 01915-085088

CamScanner

What will happen if you run processes by order - A, B, C, D

6(d) What is the difficulty of implementing optimal page replacement algorithm?

Ans: Optimal page Replacement algorithm is not practical to implement in general purpose OS because it is impossible to reliably compute how long it will be before a page is going to be used. The algorithm requires knowledge of future memory reference pattern which is not possible in most cases. This is a computationally expensive task, as it requires processing a large amount of data and making predictions based on the data.

7(a) Explain graphically what happens when we modify and delete one file which is softly linked to another file.

25.

Ans: When we create a soft link between two files in a Unix-based file system, we create a pointer from the link name to the original file. This means that any changes made to the original file can reflect in the soft link, and vice versa.

When we modify or delete the original file, the behaviour of the soft link depends on the specific operation performed.

MAHI STORE
01915-085088

CamScanner

$1KB = 2^{10}$ byte
$1MB = 2^{20}$ byte
$1GB = 2^{30}$
$1TB = 2^{40}$
$2^{50} = 1$ Peta

SAT SUN MON TUE WED THU FRI
Date :

7. (b) What will be the size of a page table in a:-

- 32-bit OS having 4kB page
- 32-bit OS " 8 KB Page
- 64-bit OS " 4KB page

3

Solⁿ

- 32-bit OS, 4KB page

Virtual Address space size $= 2^{32}$ bytes $= 4GB$

Page Size $= 4kB = 2^2 \times 2^{10} = 2^{12}$ bytes

no. of pages $= \dfrac{\text{Total posible logical Address}}{\text{Page size}} = \dfrac{2^{32}}{2^{12}} = 2^{20}$

$\therefore$ Page table size $= 2^{20} \times 4 = 2^{22} = 4MB$

MAHI STORE
01915-085088

CamScanner

32 bit OS having 8KB page

$\therefore$ V.A space size $= 2^{32}$

$\therefore$ Page size $= 8KB = 2^{10} \times 2^{3} = 2^{13}$

$\therefore$ No. of page $= \frac{2^{32}}{2^{13}} = 2^{19}$

$\therefore$ Page table size $= 2^{19} \times 8 = 2^{22} = 4MB$

64 bit OS having 4KB

V.A space size $= 2^{64}$

Page size $= 2^{2} \times 2^{10} = 2^{12}$

No. of page $= \frac{2^{64}}{2^{12}} = 2^{52}$

$\therefore$ Page table size $= 2^{52} \times 4 = 2^{52} \times 2^{2}$

$= 16PB$

7 (e) Does a two-level page table occupy less space than a single level page table?

A multilevel table for logical address translation without virtual memory has the benefit of being able to have a dynamic page table size (even if it is not paged out). Paging is based only on the prospect of a benefit.

→ A multi level (hierarchical) page table ~~has the~~ takes up significantly less space if there are huge area of unoccupied memory. A single-level page table requires a large contiguous block of memory to store all of the page table entries, which can be prohibitively large for systems with a large virtual address. 2-level P.T uses hierarchical structure to store the page table entries. This allows the page table entries to be distributed across multiple smaller blocks of memory, which can be more efficiently managed by OS.

MAHI STORE 01915-085088

CamScanner

$0.6 - x \times 0.6$

SAT SUN MON TUE WED THU FRI

Date :

(d) Say three documents are opened for editing in an office software. How many processes are running for the three documents? Will they be benefitted by paging techniques?

5

Ans: The number of processes running for the three documents depends on the specific office software being used. Typically each document is opened in a separate process, but some software may use a single processes for all documents.

Whether paging technique will benefit the three documents depends on the amount of memory available on the system and the size of the documents. Paging techniques can be benefecial if memory required to hold all three documents simultaneously exceeds the physical memory available on the system. In this case, the operating

MAHI STORE
01915-085088

CamScanner

system can use paging technique to transfer some of the documents' data to the hard drive temporarily, freeing up physical memory for other processes. But if the total memory requirement for the three documents is less than the available physical memory, paging technique will not provide any significant benefit and may even slow down the system due to the overhead involved in managing the paging process.

8(a)

	Virtual Memory	Physical memory
1.	Technique that allows a computer to use more memory than it physically has available, by temporarily transferring data from RAM to a hard drive	The actual physical memory chips installed in a computer system
2.	Increases the amount of memory available to a computer system	Provides a working space for programs and the operating system.
3	Limited by the size of hard drive	Limited by the number of physical memory chips install in the system

	Preemptive	Non-Preemptive
1.	The OS can interrupt a running process to start another process with a higher priority	OS doesn't interrupt a running process until it completes its task or voluntarily releases control
2	Ensures the high priority processes are executed quickly	Reduces overhead of sched & Increase efficiency of sys
3	Can lead to race condition and require more complex synchronization mechanism	Can result in longer respon time for high-priority proces

MAHI STORE 01915-085088

CamScanner

	Swap-in	Swap-out
1	The process of transferring a process or data from disk to memory	The process of transfering process or data from memory to disk
2	Allows a process to be loaded into memory when it is needed	Frees up memory from other processes when a process is not currently being used.
3	Occures when a process is scheduled to run but it is not currently in memory	Occurs when the operating system needs to free up memory for other processes

	Process ID	Thread ID
1	A unique identifier assigned to a process by OS	A unique identifier assigned to a thread within a process by the operating system
2	Enables OS to manage and control execution of processes	Enables OS to manage & control the execution of threads within a process
3	Unique for each process running in system	Unique for each thread within a process

8 (b) Book 403 Assume that a process is trying to access a page that is not brought into memory. Write down the steps written inside the circle of fig 8 for handling the page fault.

MAHI STORE 01915-085088

CamScanner

2019

1-ⓐ = 2020 – 1ⓐ

1.ⓑ What are the main functions of an operating system? How does it govern the entire system? (3)

Ans:- An operating system is a piece of software that manages, files, manages memory, ~~ma~~ processes, handles input & output, and controls peripheral devices like disk drives and printers.

The main functions –

1. Resource management:- The operating system manages system resources such as CPU, memory, storage, I/P O/P devices, to ensure they are utilized effectively and efficiently.

2. Process management:- The OS manages the creation, execution and termination of processes or programs.

3. Memory management:- The OS manages the allocation & and deallocation of memory to processes, and manag virtual ~~to~~ memory to ensure that each process has enough memory to execute.

4. File system management:- The operating system manages the creation, deletion and access to files and directories.

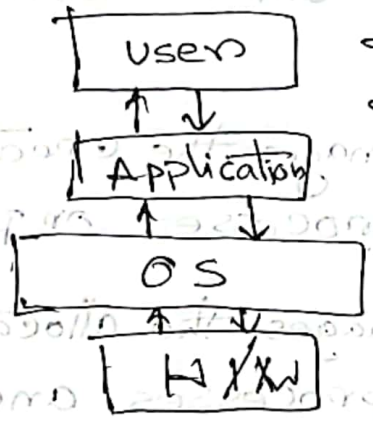
5. Security management:- The OS provides security measures such as user authentication and authorizati to protect system and it's resources.

MAHI STORE
01915-085088

CamScanner

6. Network Management: Manages network connection and communication among different computers and devices.

The OS governs the entire system by providing a layer of abstraction between h/w and application that run on it. OS provides a standard interface for application to access system resources, such as memory and i/p o/p devices, without needing to know the details of how hardware works.



O/S manages execution of multiple processes and ensures that each process gets the resources it needs to run. It manages the allocation of memory and provides virtual memory to allow processes to access more memory than is physically available.

1(c) Describe action taken by kernel to context-switch betn processes:-

175

Ans: Context-switching is the process of saving the state of a currently running process and restoring the state of another process so that it can run. The act.

MAHI STORE
01915-085088

CamScanner

Actions :

1. In response to a clock interrupt, the OS saves the PC and user stack pointer of the currently executing process, and transfers control to the Kernel clock interrupt handler.

2. The clock interrupt handler saves the rest of the registers, as well as other machine state, such as the state of floating-point registers, in the process PCB.

3. The OS invokes the scheduler to determine the next process to execute.

4. The OS then retrives the state of the next process from its PCB, and restores the register. This restore operation takes the processor back to the state in which this process was previously interrupted, executing in user C code with user mode privileges.

2ⓐ - 2020 - 8ⓐ

2ⓑ Illustrate criteria used for comparing CPU scheduling algorithm

1. CPU Utilization :- CPU utilization should be maximized. This can be achieved by keeping CPU busy as much as possible

2. Throughout :- The number of processes that complete their execution per unit time is maximized.

3. Turnaround time : The time taken from the submission of a process to its completion should be minimized

4. Waiting time:- The time a process spends waiting in the ready queue should be minimized

5. Response time:- The time taken for the system to respond to a request ~~for~~ from a process should be minimized.

6. Fairness:- The scheduling algorithm should allowcate CPU time fairly among all the processes

7. Priority:- The scheduling algorithm should allow for processes with higher priorities to be executed before processes with lower priority.

8. Preemption:- The ability to interrupt a running process and allocate the CPU to another process should be considered

By comparing CPU scheduling.

2(c)

Process	BT
P_1	10
P_2	29
P_3	3
P_4	7
P_5	12

quantum = 10 ms

SJF

P_3	P_4	P_1	P_5	P_2

0 3 10 20 32 61

CT	TAT	WT
20	20	10
61	61	32
3	3	0
10	10	3
32	32	10

$ATAT = \frac{20+61+3+10+32}{5} = 25.2$

$AWT = \frac{10+32+3+10}{5} = 11$

FCFS

P_1	P_3	P_2	P_3	P_4	P_5

0 10 20 39 42 49 61

Pi	AT	BT	CT	TAT	WT
P_1	0	10	10	10	0
P_2	0	29	39	39	10
P_3	0	3	42	42	39
P_4	0	7	49	49	42
P_5	0	12	61	61	49

TAT = CT − AT

WT = TAT − BT

$A.WT = \frac{10+39+42+49}{5} = 28$

$ATAT = \frac{10+39+42+49+6}{5} = 40.2$

RR

~~P_1~~	~~P_2~~	~~P_3~~	~~P_4~~	~~P_5~~	~~P_2~~	P_5

P_1	P_2	P_3	P_4	P_5	P_2	P_5

0 10 20 23 30 40 49 51

CT	TAT	WT
10	10	0
49	49	10
23	23	20
30	30	23
51	51	39

ATAT = 32.6

AWT = 18.4

MAHI STORE 01915-085088

CamScanner

3(a) What is the difference between asymmetric and symmetric multiprocessing technique?

Multiprocessing:- The use of two or more central processing units within a single computer system. Work should be able to be distributed evenly among the system's available CPUs in a multiprocessing system.

Asymmetric		Symmetric
Processes are not equally treated	1	All processes are treated equaly.
Tasks of OS are done by master processor	2	Tasks of OS are done by individual processor
No communication betn Processes as they are controlled by master process	3	All processors communicat with another processor by a shared memory
Processor Scheduling approach used is master slave	4	Processor is taken from the ready queue
Assymetric multiprocessing system are Cheaper	5.	Are Costlier
Easier to design	6	Complex to design
Exhibit different architecture	7.	Architecture of each Processor is same

MAHI STORE 01915-085088

CamScanner

3(b) Explain different migration technique for load balancing betn different processors in multiprocessing environment?

25

MAHI STORE

CamScanner

86

SAT SUN MON TUE WED THU FRI

Date

3 (c) What do you mean by "Race condition"? Explain how it occurs in case of co-operating multiprocess environment.

2.25

MANI STORE
01915-085088

CamScanner

SAT SUN MON TUE WED THU FRI

Date

3(a) What do you mean by 'busy waiting'? Propose some alternative approach in place of busy waiting to synchronize cooperating processes
2

88

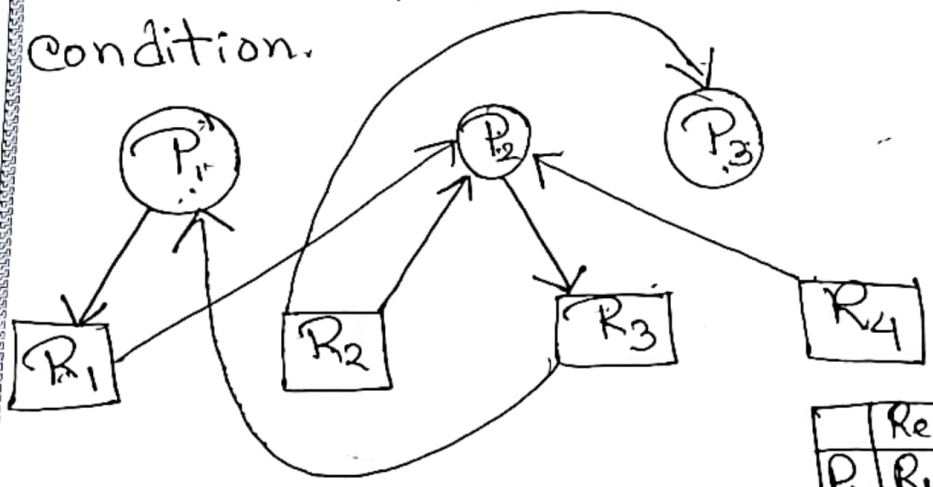
Teresa Jency Bala

4 (a) Define deadlock. Consider three processes $\{P_1, P_2, P_3\}$ four resource types $\{R_1, R_2, R_3, R_4\}$ and six edges $E = \{P_1 \to R_1, P_2 \to R_3, R_1 \to P_2, R_4 \to P_2, R_3 \to P_1, R_2 \to P_3\}$ Draw the resource allocation graph and explain whether the system is in deadlock or not.

Deadlock:- A situation in which more than one process is is blocked because it is holding a resource and also requires some resource that is acquired by some other processes, The four necessary condition for a deadlock situation to occur are mutual exclusion, hold & wait, no preemption, and circular wait. We can prevent deadlock by preventing any one of these condition.

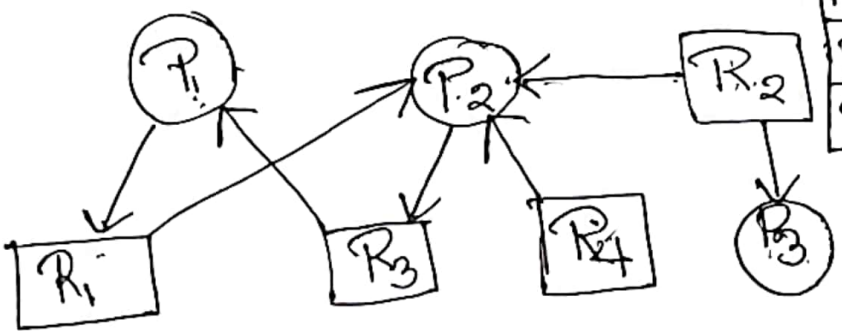
$3x = 0$

$3x = 4 \times 0.3$



P_3 will execute And ~~recive~~ release resource R_2. But

	Require	Assign
P_1	R_1	R_3
P_2	R_3	R_4, R_2
P_3		R_2



P_3,

But

MAHI STORE
01915-085038

CamScanner

Yes, deadlock could exist, The 4 necessary conditions all are true.

- Mutual exclusion: True because Resource cannot be shared.
- Hold & Wait: It i true because P_2 holds resource R_4 but also requests (waits for) resource R_2
- No preemption: Is stated as we are not not breaking down resources.
- Circular Wait: Exists: P_2 wants R_3 but R_3 is held by P_1, P_1 wants R_1 which is held by P_2.

(b)

Process	Allocation A B C D	MAX A B C D	Need A B C D	Available A B C D
P_0	0 0 1 2	0 0 1 2	0 0 0 0	1 5 2 0
P_1	1 0 0 0	1 7 5 0	0 7 5 0	2 8 11 10
P_2	1 3 5 4	2 3 5 6	1 0 0 2	1 5 3 2
P_3	0 6 3 2	0 6 5 2	0 0 2 0	2 8 8 6
P_4	0 0 1 4	0 6 5 6	0 6 4 2	2 8 10 6

MAX - Allocation

1520
0012
153
135
288
002
2810
001
281

$P_0 \rightarrow P_2 \rightarrow P_3 \rightarrow P_4 \rightarrow P_1$

MAHI STORE 01915-085088

CamScanner

SAT SUN MON TUE WED THU FRI

Date :

4. (c) What are advantages of aborting one process at a time untill the deadlock cycle is eliminated to recover a system from deadlock situation?

Process termination:- to eliminate the deadlock we can simply kill one or more processes. For this, we use two method:

(a) Abort all the Deadlocked Processes:-

Abort all the deadlocked Processes: Aborting all the processes will certainly break the deadlock, but with a great expense. The deadlock processes may have computed for a long time and the result of those partial computations must be discarded and there is a probability to recalculate them later.

(b) Abort one process at a time until deadlock is eliminated:- Abort one deadlocked process at a time, untill deadlock cycle is eliminated from the system. Due to this method, there may be considerable overhead, because after aborting each process, we have to run deadlock detection algorithm to check whether any process are still deadlocked.

MAHI STORE
01915-085088

CamScanner

2. Resource Preemption:- To eliminate deadlocks using resource preemption, we preempt some resources from processes and give those resources to other processes. This method will rise 3 issues -

(a) Selecting a victim:- We must determine which resources and which processes are to be preempted and also the order to minimize the cost.

(b) Rollback:- We must determine what should be done with the process from which resources are preempted. One simple idea is total rollback. That means abort the process and restart it.

(c) Starvation: In a system, it may happen that same process is always picked as a victim. As a result, that process will never complete its designated task. This situation is called Starvation. and must be avoided. One solution is that a process must be picked as a victim only a finite number of times.

MAHI STORE
01915-085088

CamScanner

5(a) What do u mean by external & internal fragmentation? 2

Ans:- <u>Internal fragmentation:</u> When a process is allocated more memory than required few space is left in the block which causes Internal Fragmentation. It occurs when memory is divided into fixed sized partitions. It can be cured by allocating memory dynamically or having partitions of different sizes

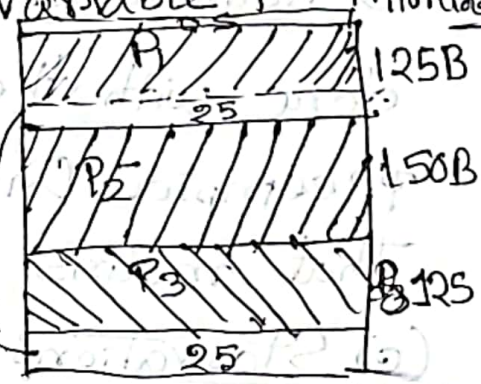
E.g ⇔ Variable partitioning

$P_1 = 100B$

$P_2 = 150B$

$P_3 = 100B$

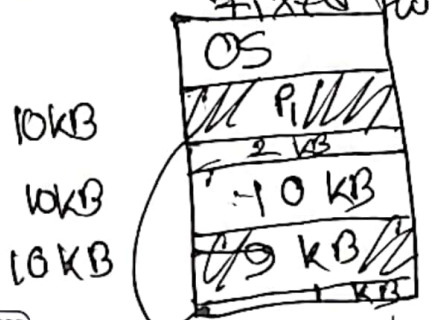
$P_4 = 50B$



<u>External fragmentation:-</u>

After execution of processes when they are swapped out of memory and smaller processes replace them many small non-contiguous blocks of empty spaces are formed which can serve a new request if put together but as they are non-contiguous they cannot serve a new request causing external fragmentation. It occurs when memory is divided into variable sized partition based on the size of process. It can be cured by compaction, paging & segmentation

Fixed partition.



$P_1 = 8KB$

$P_2 = 10KB$

$P_3 = 9KB$

→ internal fragmentation

MAHI STORE
01915-085088

CamScanner

2019

5 (b) When do we need hierarchical paging? Explain the address translation for two level-32-bit paging architecture?

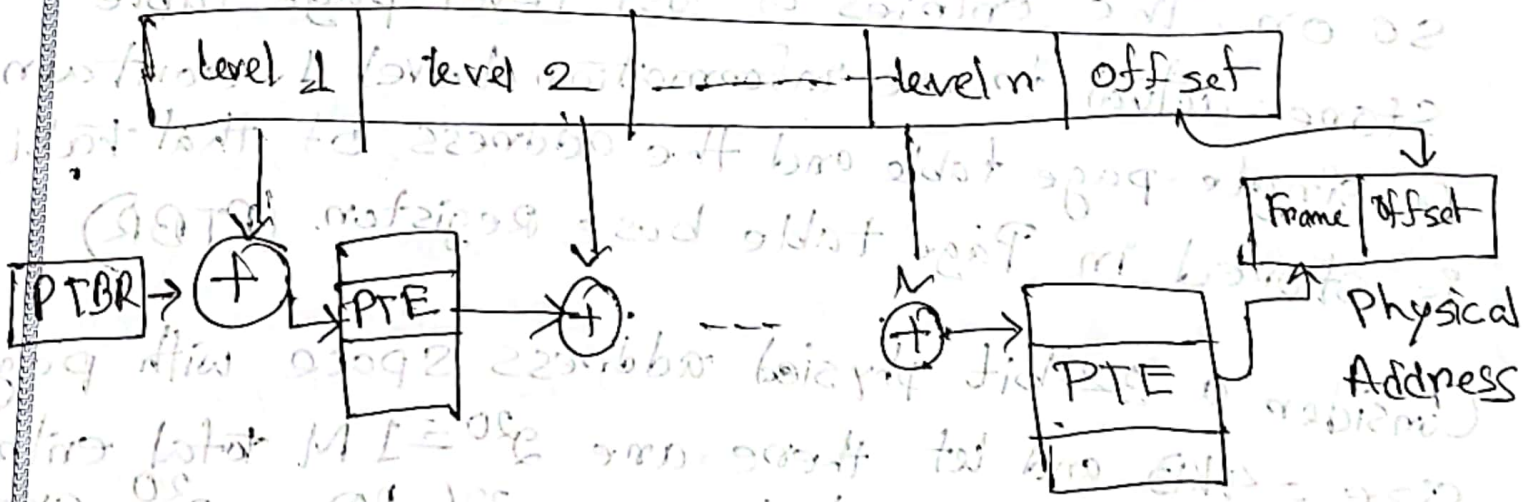
3.5. Ans: ~~Here~~ Multilevel Paging ~~scheme that aka hierarchical paging scheme~~ ~~consis~~ that consists of two or more levels of pagetables in a hierarchical manner. It is also known as hierarchical paging. The entries of the ~~last~~ level ~~page ta~~ 1 page table are pointers to a level 2 page table and entries of the level 2 page tables are pointers to a level 3 page table and so on. The entries of last level page table store actual frame information. Level 1 contains a single-page table and the address of that table is stored in Page table base Register (PTBR)

Consider a 32-bit physical address space with page size = 4KB and let there are $2^{20} = 1M$ total entries in the page table, Page table size = $2^{32}/2^{12} = 2^{20}$ and adding some protection bits & dirty bit in the page table entry. Now Page table size = $2^{20} \times 24 \approx 3MB$ which should be the physical memory and since each process has it's own page table there are mu memory wastage only for storing pagetables.

MAHI STORE
01915-085088

CamScanner

~~For~~ For solving this issue of large memory requirement of page table is to use multilevel paging. Only the outer most page table will reside in the ~~memor~~ main memory and other pages table will be brought to main memory as per the requirement because at a particular time we do not need complete page table. also we can save much memory space because outermost page table can fit in exactly one frame

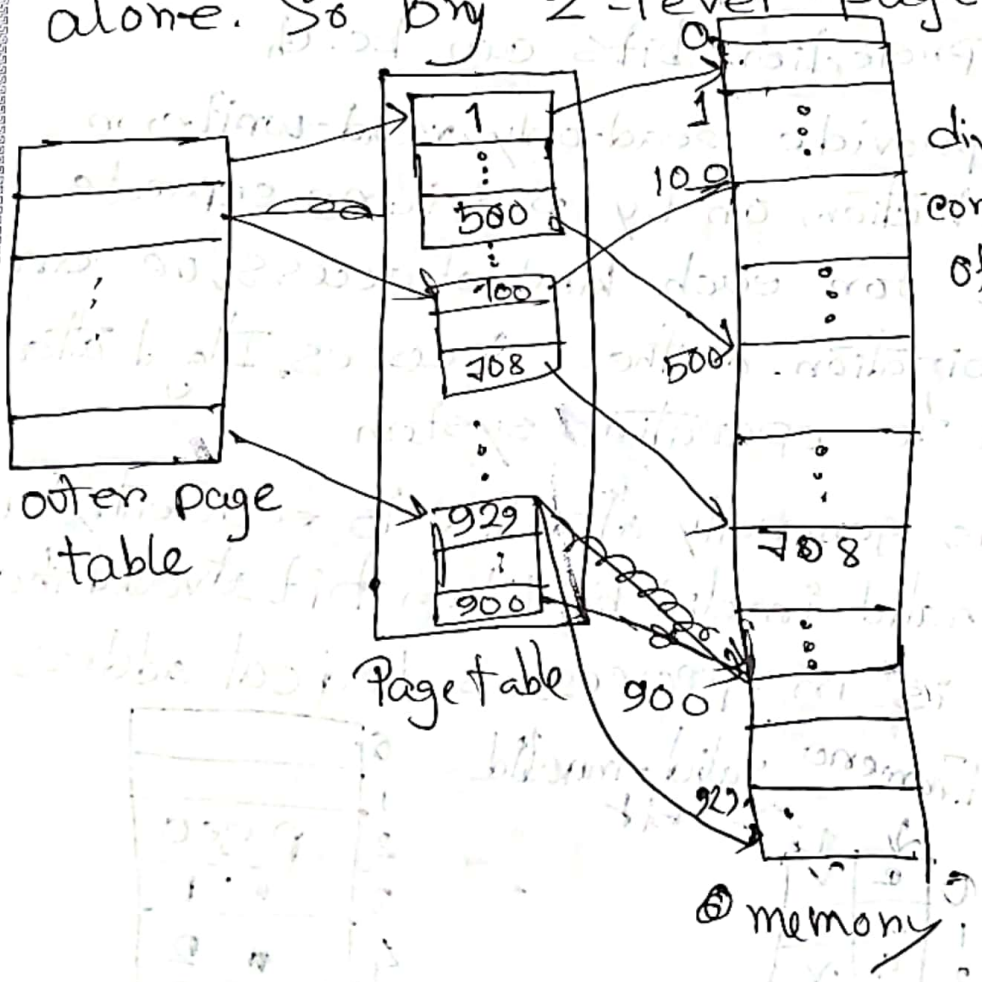


SAT SUN MON TUE WED THU FRI

Date :

2-level 32 bit paging architecture address translation:-

~~In 2~~ 32-bit architetune with 2^{12} = 4kbyte pages. Suppose page size in system is 4KB (2^{12}) then a page table may consist of up to 1 million entries ($2^{32}/2^{12}$). Assume that each entry consists of 4 bytes, each process may need up to 4MB of physical address space for the page table alone. So by 2-level pageing algorithm,



A logical address is divided into a page no. consisting of 20 bit & offset consisting 12 bit:

Page number		page offset
P_1	P_2	d
10	10	12

AHI STORE
915-085088

CamScanner

9C
Pg book - 375

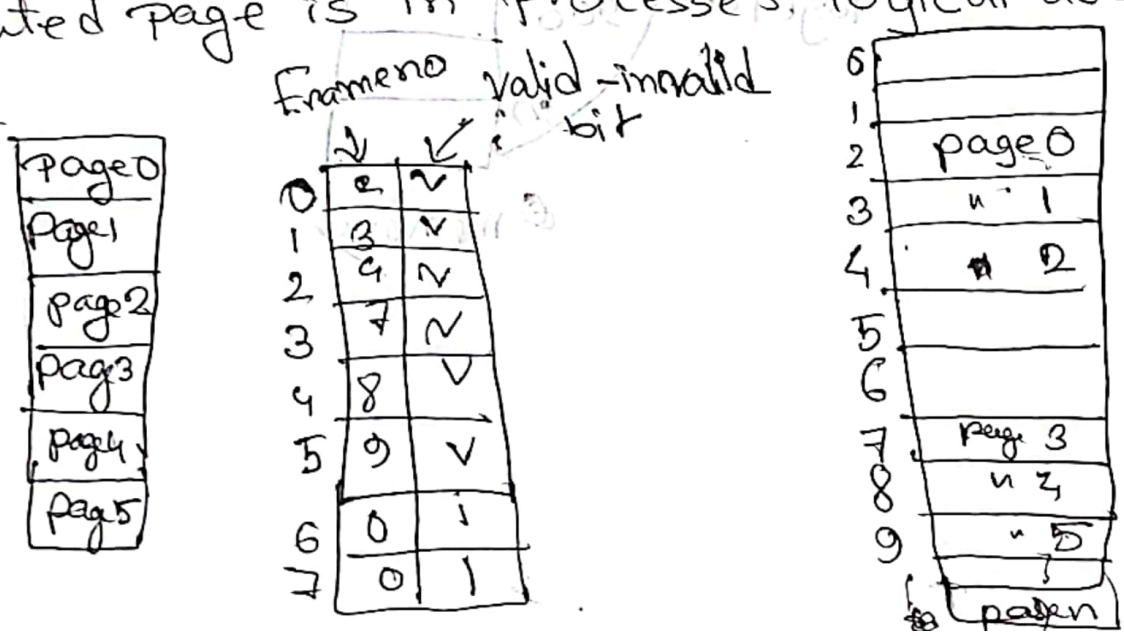
5(c) Explain how paging environment can be used for memory protection.
3.5

Ans:- Memory protection in a paged environment is accomplished by protection bits associated with each frame. These bits are kept in page table.

One bit can define a page to be read-write or read-only. Every reference to memory goes through the page table to find the correct frame number. At the same time that the physical address is being computed, the protection bits can be ch

Create h/w to provide read-only, read-write, or execute-only protection, or by providing separate protection bits for each kind of access, we can allow any combination of these accesses. Illegal attempts will be trapped to operating system

One additional bit is generally attached to each entry in the page table. a valid-invalid bit. When bit is valid the associated page is in processes logical address space



MAHI STORE 01915-085088

$\theta = \theta_1 - g(\theta_1 - \theta_2)$

$\theta_1 \quad \theta \quad \theta_2$

Any attempt to generate page 6 or 7 will find that the valid-invalid bit is set to invalid, & computer will trap to the OS (invalid page reference).

6(a) Define Virtual memory. What are advantages of it?
↳ 2020 – 6(a).

6(b) Calculate page fault for given memory reference string: 7, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2, 1, 2, 0, 1, 7, 0, 1

FIFO Consider frame = 3

7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1
7	7	7	2	2	2	2	4	4	4	0	0	0	0	0	0	0	7	7	7
	0	0	0	0	3	3	3	2	2	2	2	2	1	1	1	1	1	0	0
		1	1	1	1	0	0	0	3	3	3	3	3	2	2	2	2	2	1

* * * * *

hit = 5 P.F = 20 − 5 = 15

LRU

7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1
7	7	7	2	2	2	2	4	4	4	0	0	0	1	1	1	1	7	1	1
	0	0	0	0	0	0	0	0	3	3	3	3	3	3	0	0	0	0	0
		1	1	1	3	3	3	2	2	2	2	2	2	2	2	2	7	7	7

* * * * * * * *

hit = 7 Page fault = 20 − 7 = 13

Optimal

7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1
7	7	7	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	7	7
	0	0	0	0	0	0	4	4	4	0	0	0	0	0	0	0	0	0	0
		1	1	1	3	3	3	3	3	3	3	3	1	1	1	1	1	1	1

* * * * * * * * * * * *

hit = 11
f = 9

MAHI STORE 01915-085518

CamScanner

6 (e) What's Belady's Anomaly?

For some page-replacement algorithms the page-fault rate may increase as the number of allocated frames increases.

also known as FIFO anomaly. Page fault are instances where a required page isn't present in physical memory and needs to be fatched from disk

7 (a) What is a file? What are common attributes maintained for a file? (3)

Ans:- A file is a named collection of related information that is stored on a computer's storage device, such as hard disk, flash drive or network serve. Files are used to store data documents and other types of digital contents

Files can be 2 types

- Data → binary, numeric, character
- Program

Contents in file are defined by the file creation: text file, source file, executable file

Some file attributes are -

1. Name

MAHI STORE
01915-085088

CamScanner

pg- 504

A file is a named collection of related info. that are recorded in the secondary storage. Data cannot be written to secondary storage unless they are written within a file.

File is a sequence of bits, bytes, lines or records the meaning of which is defined by the files creator and user.

Attributes -

1. Name - Only information kept in human-readable form
2. Identifier - Unique tag (number) identifies file within file system
3. Type - needed for systems that support different types.
4. Location - Pointer to file location on device.
5. Size - Current file size
6. Protection - Controls who can do reading, writing, executing
7. Time, date and user identification - Data for protection, security, and usage monitoring.

Information about files are kept in the directory structure which is maintained on the disk. Information are kept in the directory structure

MAHI STORE
01915-085088

CamScanner

100

$x^3 = 1$

$x^3 - 1 = 0$

$(x-1)(ax^2+bx+c)$

$x = \frac{-1 \pm \sqrt{1-4}}{2}$

$= \frac{-1 \pm \sqrt{3}i}{2}$

$x = \sqrt{1}$

$x^2 = i$

$x^2 - i =$

lenes ax^2+bx+c

tresa: en cy Jecy

SAT SUN MON TUE WED THU FRI

Date: Jein.

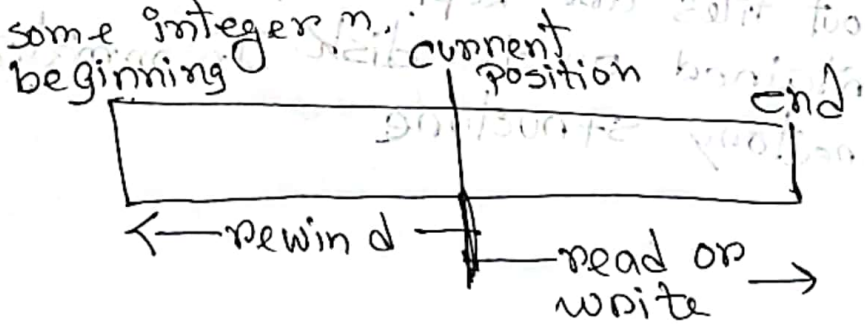
7 (b) Describe the sequential access and direct access method for file.

(4)

Ans:- Sequential access method processed in order while Direct access (or relative access) allows

Sequential Access:- Information in file is processed in order, one record after the other. Editors and compilers usually access files in this way.

pg-513 Reads and writes make up the bulk of the operations on a file. A read operation ~~read~~ read_next() - reads the next portion of the file and automatically advances a file pointer which tracks the I/O location. The write operation - write_next() - appends to the end of file and advances to the end of newly written material (the new end of file). Such a file can be reset to the beginning, and some systems, a program may be able to skip forward or backward n recor for some integer n.



MAHI STORE
01915-085088

CamScanner

SAT SUN MON TUE WED THU FRI
Date :

Direct Access:- A file is made up of fixed-length logical records that allows programs to read and write records rapidly in no particular order. It is based on a disk model of a file, since disks allow random access to any file block.

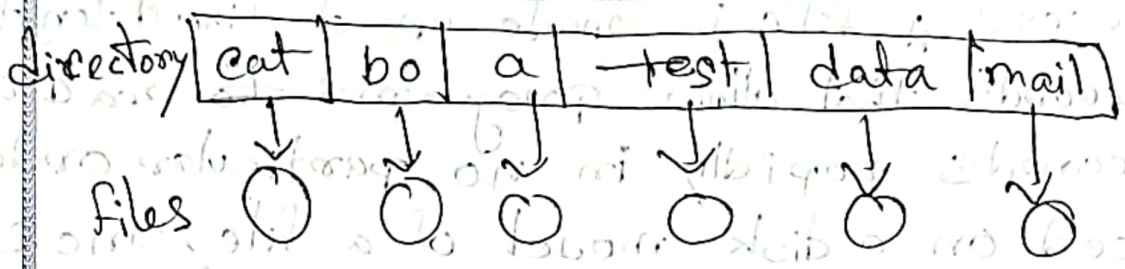
* The file is viewed as a numbered sequence of blocks or records. We have read(n), write(n), position to n, read_next(n), write_next(n) etc where n is the relative block number. The relative block number is an index relative to the beginning of the file.
Relative block numbers allows OS to decide where file should be placed.

7(c) What is the limitation of single level directory structure. explain.

Ans. 1) With the number of files increase there is need of more unique names to identify the files.
2) Where there is more than one user and the want same name for individuals files then the unique-name role is violated.
3) Even a single user on a single-level directory may find it difficult to remember names of all the files as number of files inceases.
4) Searching will become time taking if the directory is large

MAHI STORE
01915-085088

CamScanner



Single-level directory.

8(a) Define 'boot control block' & 'Volum control block'.

Pg-546

UFS-Unix file system

FCB-file Control block

Boot control block (Per volume) can contain information needed by the system to boot an OS from. If the disk does not contain any OS, this block can be empty. It is typically the first block of a volume. In UFS, it is called the boot block. In NTFS, it is the partition boot sector.

Volume Control block (per volum) contains volume (or partition) details, such as the number of blocks in the partition, the size of the blocks, a free-block count and free block pointers, and a free-FCB count and FCB pointers. In UFS, this is called a super block. In NTFS, it is stored in the master file table.

MAHI STORE
01915-085088

CamScanner

New technology file system

A Directory Structure (per file System) is used to organize the files. In UFS this includes file names and associated inode numbers. In NTFS, it is stored in the master file table.

Pre-file FCB contain many details about the file. It has a unique identifier number to allow association with a directory entry. In NTFS this information is actually stored within the master file table, which uses a relation database structure, with a row per file.

8 (b) What is FCB? Explain its importance and uses for implementing a file system.

2.25

FCB: Stands for File Control Block. It is a data structure used by file system to store information about a particular file, such as it's name, location, size, permission, ~~and~~ size, dates.

File permission
File dates (create, access, write)
File owner, group, ACL
File Size
file data blocks or pointers to file data blocks

MAHI STORE 01915-085088

CamScanner

SAT SUN MON TUE WED THU FRI

Date :

FCB plays important role in implementing a file system, as it provides a way for the OS to keep track of all the files in the system and to manage them efficiently. Some key uses of FCB include:-

1. File creation & deletion:- When a new file is created the application program calls the logical file system. The logical file system knows the format of the directory structure. To create a new file, it allocates a new FCB. And after a file gets deleted, OS removes corresponding FCB from system.

2. File Openining & Closing:- When a file is opened, the OS retrieves its FCB from File system & loads it into memory. When file is closed, the FCB is updated with any changes that were made to the file, and then written back to disk.

3. File Reading & Writing:- read() & write() operation the FCB determine the location of file on disk and to manage any buffer on cache operation that may be necessary.

4. File locking & unlocking:- The FCB can be used to implement file locking mechanism which prevents multiple processes from accessing same file at same time. This helps to prevent

MAHI STORE 01915-085088

CamScanner

data corruption & ensure data consistency.

Q5 (c) Explain different approaches from implementing indexed allocation technique for storage allocation.

Ans: Indexed file allocation is a file allocating system where the pointers to all elements of file are put together into one location which is called index block. That means, we will get all the locations of blocks in one index file. ~~The blocks and pointers were spread over the memory in the linked allocation method, where retrival was accomplished by~~ The data retrival is accomplised by visiting index block to retrieve.

Advantages

1. Reduces possibilities of external fragmentation
2. Rather than accessing sequentially it has direct access to block.

Disadvantages:-

1. Have more pointer overhead is there
2. If we lose the index block we cannot access the complete file.
3. It becomes heavy for small files
4. It is possible that a single index block.

6

SAT SUN MON TUE WED THU FRI

Date :

answer

cannot keep all pointers for some large files.

To resolve this issue —

1. Linked Scheme : If the file is big then more blocks are required so one index block is Insufficient to store all the pointers, therefore to store the pointers two or more index blocks are used where these index boxes are connected using linked file allocation that is each index block stores the pointer to the next index block.

2. Multilevel Index :- The multiple indexes blocks along with the levels of these blocks. Here, the level 1 block is used for pointing to the level 2 block which points to the blocks occupied by the file. These index blocks can be extended to three or more levels according to the size of the file.

3. Combined Scheme :- A special block is used to store all the informations relation to the file like name, authority, size etc. The special block is called

inode (information-node). Some space of this special block is used to store the information related to the feild as mentioned above and the remaining space is used to store the addresses of blocks that contain the actual file.

pg-561

3.(d) Explain how a bit vector can be used to manage free space in storage.

A bit vector, also known as a ~~bitmap~~ bitmap, is a data structure that is commonly used in Computer systems to manage free space in storage. Bit vector is to use a binary digit, or bit, to represent each unit of space in the storage system. (eg. each block or sector on a disk).

The main advantage of this approach is its relative simplicity and its efficiency in finding the first free block or n consecutive free blocks on the disk.

One technique for finding the first free block on a system that uses a bit-vector to allocate disk space is to sequentially check eac

word in the bit map to see whether that value is not 0, since a 0 value word contains only 0 bits and represents a set of allocated blocks. The first non-0 word is scanned for the first 1 bit, which is the location of the first free block.

MAHI STORE
01915-085088

CamScanner