

Assignment \Rightarrow II

Name :- Subhadeep Bhadra

Roll :- 33200118010

Sem :- 5th Sem

Sub :- Operating System (PCC-CS-502)

17) What do you mean by deadlock? Write down all necessary condition of Deadlock

→ In multiprogramming environment, several processes may compete for a finite number of resources. A resource is requested by a process, but if the resource is not available at that time, the process enters the waiting state. Sometimes, a waiting process is never able to change state, because the resource it has requested are held by other waiting processes. This situation is called a deadlock.

Necessary Conditions

Deadlock situation can arise if the following 4 conditions held simultaneously in a system:

- Mutual exclusion: At least one resource must be held in a non-sharable mode; that is, only one process at a time can use the resource. If another process requests that resource, the requesting process must be delayed until the resource has been released.
- Hold and Wait: A process must be holding at least one resource and waiting to acquire additional resources that are currently being held by other processes.
- No preemption: Resources cannot be preempted, a resource can be released only voluntarily by the process holding it after the process completed its task.
- Circular Wait: A set $\{P_0, P_1, \dots, P_n\}$ of waiting processes must exist such that P_0 is waiting for a resource held by P_1 , P_1 is waiting for a resource held by P_2 , ..., P_{n-1} is waiting for a resource held by P_n , and P_n is waiting for the resource held by P_0 .

18) Consider the following snapshot of a system where P_0, P_1, P_2, P_3, P_4 are the processes and A, B, C are the resource type (A(10), B(5), C(7)) and snapshot at time T_0 .

Processes	Allocation			Max			Available		
	A	B	C	A	B	C	A	B	C
P_0	0	1	0	7	5	3	3	3	2
P_1	2	0	0	3	2	2			
P_2	3	0	2	9	0	2			
P_3	2	1	1	2	2	2			
P_4	0	0	2	4	3	3			

i) What is the content of need matrix?

Process	A	B	C
P0	7	4	3
P1	1	2	2
P2	6	0	0
P3	0	1	1
P4	4	3	1

ii) Is the system in safe state?

A	B	C	
3	3	2	
5	3	2	(P1)
7	4	3	(P3)
7	4	5	(P4)
7	5	5	(P0)
10	5	7	(P2)

∴ The system is in safe state

iii) If a request from process P₁ arrives for (1, 0, 2). Can this request be granted immediately?

→ P₁ requests (1, 0, 2)

Allocation P₁ = 3 0 2

Need P₁ = 0 2 0

Available = 2 3 0

A	B	C	
2	3	0	
5	3	2	(P1)
7	4	3	(P3)
7	4	5	(P4)
7	5	5	(P0)
10	5	7	(P2)

Request of P₁ can be granted immediately.

19) Write the difference between logical and physical address space

→ Logical Address

i) The address generated by CPU is a logical address

ii) The set of all logical addresses generated by a program is referred to as a logical address space

Physical Address

i) Address actually available on memory unit is a physical address

ii) The set of all physical addresses ~~generated~~ corresponding to logical address is called physical address space

iii) The user program deals with virtual (logical) address

iii) The user program never sees the real physical address.

10) What do you mean by dynamic loading?

→ In dynamic loading, a routine of a program is not loaded until it is called by the program. All routines are kept on disk in a relocatable load format. The main program is loaded into memory and is executed. Other routines or modules are loaded on request. It makes better memory space utilization and unused routines are never loaded.

21) What do you mean by dynamic linking?

→ Linking is the process of collecting and combining various ~~module~~ modules of code and data into an executable file that can be loaded into memory and executed. OS can link system level libraries to a program. When it combines the library at load time, the linking is called static linking but when this linking is done during the execution time, then it is called dynamic linking. In static linking libraries are linked at compile time so program code size becomes bigger whereas in dynamic linking libraries are linked at execution time so program code size remains smaller.

22) First fit: Allocate the biggest hole that is big enough. Searching can start either at the beginning of the set of holes or at the location where the previous first fit search ended. We can stop searching as soon as we find a free hole that is large enough.

Best fit: Allocate the smallest hole that is big enough. We must search the entire list. Unless the list is ordered by size. This strategy produces the smallest leftover hole.

Worst fit: Allocate the largest hole. Again we must search the entire list. Unless it is sorted by size. This strategy produces the largest leftover hole, which may be more useful than the smaller leftover hole from the best-fit approach.

23) As processes are loaded and removed from memory, the free memory space is broken into little pieces. It happens after sometimes that processes cannot be allocated as memory blocks remain unused. This is called fragmentation.

Types of fragmentation:

i) External fragmentation

ii) ~~Internal~~ Internal fragmentation

→ External fragmentation: Total memory space is enough to satisfy a request or to reside a process in it, but it is not contiguous. So it cannot be used.

ii) Internal fragmentation: Memory block assigned to a process is bigger some portion of memory is left unused as it cannot be used by another process.

24) Paging is a memory-management scheme that permits the physical address space of a program to be non contiguous. It avoids ~~external~~ external fragmentation and the need for compaction. It also solves the considerable problem of fitting memory chunks of varying sizes onto the backing stores; most memory management used before the introduction of paging suffered from this problem. The problem arises because when some fragments or residing in main memory need to be swapped out. Space must be found on the backing store. The backing store has the same fragmentation problems discussed in connection with main memory but the access is much slower. So compaction is impossible because of its advantages over earlier methods. Paging in its various forms is used in most operating system.

25) Segmentation is a technique to break memory into ~~logical~~ logical pieces, where each piece represents a group of related information. For exam, data segments for operating system and so on. Segmentation can be implemented with or without using paging.

Online paging, segment is having varying sizes and there eliminated internal fragmentation. External fragmentation still exists but to lesser extent.

26) Virtual memory is a technique that allows the execution of processes which are not completely available in memory. Here the program can be longer than physical memory. Virtual memory is a separation of user logical memory from physical memory. This separation allows an extremely large virtual memory to be provided for programmes when only a smaller physical memory is available.

27) Considering an executable program to be loaded from disk into memory. One option is to load the entire program in physical memory at program execution time. But the problem with this approach is that we may not initially need the entire program in memory. Suppose a

program start with a list of available options from which the user is to select. Loading the entire program into memory results in loading the executable code for all options regardless of whether an option is ultimately selected by the user or not. An alternate strategy is to load pages only as they are needed. This technique is known as demand-paging.

A demand-paging system is similar to a paging system with swapping where processes reside in secondary memory. When we want to execute a process we swap it into memory. Rather than swapping the entire process into memory. However, we use ~~lazy~~ lazy swapping. A lazy swapper never swaps a page into memory unless the page will be needed.

28) 7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1
 newest → 7 0 1 2 3 0 4 2 3 0 3 1 2 2 7 0 1
 7 0 1 3 3 0 4 2 3 3 1 2 2 7 0
 7 0 0 3 0 4 2 2 3 0 0 2 2 7
 h h h h

FIFO

Page hit = 5

Page fault = 15

$$\text{Page fault rate} = \frac{15}{20} = \frac{3}{4}$$

7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1
 newest → 7 0 1 1 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
 7 0 0 0 0 0 4 4 4 0 0 0 0 0 0 0 0 0 0 0
 7 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 7 7 7
 h h h h h h h h

optimal

Page hit = 11

$$\text{page fault rate} = \frac{9}{20}$$

Page fault = 9

7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1
 newest → 7 0 1 1 1 3 3 3 2 2 2 2 2 2 2 2 2 7 7 7
 7 0 0 0 0 0 0 0 3 3 3 3 3 3 0 0 0 0 0
 7 2 2 2 2 4 4 4 0 0 0 1 1 1 1 1 1 1 1
 h h h h h h h h

Page hit = 8

$$\text{page fault rate} = \frac{12}{20} = \frac{3}{5}$$

Page fault = 12

29> In computer storage, Belady's anomaly is the phenomenon in which increasing the number of page frames results in an increase in the number of page faults for certain memory access patterns. This phenomenon is commonly experienced when using the first-in-first-out page replacement algorithm.

Example:-

page req:- 3 2 1 0 3 2 4 3 2 1 0 4

newest page:- 3 2 1 0 3 2 4 4 4 1 0 0

3 2 1 0 3 2 2 2 4 1 1

3 2 1 0 3 3 3 2 4 4

F F F F F F H H F F H

page ~~fault~~ fault = 9

page req:- 3 2 1 0 3 2 4 3 2 1 0 4

newest page:- 3 2 1 0 0 0 4 3 2 1 0 4

3 2 1 1 1 0 4 3 2 1 0

3 2 2 2 1 0 4 3 2 1

3 3 3 2 1 0 4 3 2

F F F F H H F F F F F F

Page fault = 10

30> If the number of frames allocated to a low priority process falls below the minimum number required by the computer architecture, we must suspend that process' execution. We should then page out its remaining pages freeing all its allocating frames. This provision introduces a swap-in, swap-out level of intermediate CPU scheduling.

In fact, looking at any process that does not have enough frames. If the process does not have the number of frames it needs to support pages in active use, it must replace a page that will be needed again right away. Consequently, it quickly faults again and again replacing pages that it must bring back immediately. This high paging activity is called thrashing. It is spending more time in paging than executing.

31> The main disadvantage of linked list allocation is that the Random access to a particular block is not provided. In order to access a block, we need to access all its previous blocks.

File allocation table overcomes this drawback of linked

list allocation. In this scheme, a file allocation is maintained, which gathers all the disk block links. This table has one entry for each disk block and is indexed by block number.

File allocation table needs to be searched in order to reduce the number of head seeks. Now the head doesn't need to traverse all the disk blocks in order to access one successive block.

32) In Unix based operating system each file is indexed by an inode. Inodes are special disk blocks they are created when the file system is created. The number of Inode limits the total number of files/direction that can be stored in the file system.

33) Direct Memory Access (DMA):- Many computers avoid burdening the main CPU with programmed I/O by offloading some of this work to a ~~data~~ special purpose processor. This type of processor is called a DMA controller. A special control unit is used to transfer block of data directly between an external device and main memory without intervention by the processor. This approach is called direct memory access.

34) Kernel I/O subsystem is responsible to provide many services related to I/O. Following are some of the services provided.

- Scheduling:- Kernel schedules a set of I/O requests to determine a good order in which to execute them. When an application issues a blocking of I/O system call, the request is placed on the queue for that device. The kernel I/O scheduler rearranges the order of the queue to improve the overall system efficiency and the average response time experienced by the applications.
- Buffering:- Kernel I/O subsystem maintains a memory area known as buffer that stores data while they are transferred between two devices between a device and with an application operation.
- Caching:- Kernel maintains cache memory which is region of fast memory that holds copies of data. Access to the cached copy is more efficient than access to the original.

- Spooling and device reservation :- A spool is a buffer that holds output for the device. Such as a printer, that cannot accept interleaved data streams. The spooling system copies the queued spool files to the printer at a time.
- Error handling :- An operating system that uses protected memory can guard against many kinds of hardware and application errors.

35) Authentication :- Authentication refers to identifying each user of the system and associating the executing program with those user. It is the responsibility of operating system to create a protection system that ensures that a user who is running a particular program is authentic.

Program thread :- Operating system's processes and kernels do the designated task as instructed. If a user program made these process do malicious tasks then it is known as program thread.

System threat :- They refer to misuse of system services and network connections to put user in trouble. System threats can be used to launch program attack. System threats create such an environment that operating system resources/files are misused.