service.

module-03

① Define deadlock. What are necessary conditions for deadlock to occur?

Ans:- A deadlock is a situation in a multi-tasking systems where two or more processes are unable to proceed because each is waiting for the other to release a resource.

In a deadlock, processes never finish executing, and system resources are tied up, preventing other jobs from starting.

Necessary Conditions for Deadlock:-

* Mutual Exclusion:-

⇒ At least one resource must be held in a non-shareable mode. If another process requests this resource, it must wait until the resource is released.

Ex:- A printer can only be used by one process at a time.

2. Hold and wait.

- A process holding at least one resource is waiting to acquire additional resources held by other processes.

Ex: Process A holds resource 1 and wait for resource 2, while process B holds resource 2 and wait for resource 1.

3. No preemption: or No preemtive

- A resource cannot be forcibly removed from a process holding it. The process must release the resource voluntarily.

Ex: If a process A is printing, the printer cannot be taken away until the print job finishes.

4. Circular wait:

- A circular chain of processes exists, where each process is waiting for a resource held by the next process in the chain.

Ex: Process A waits for a resource held by Process B, which waits for a resource held by process C and so on, until the chain loops back to Process A.

(2) What is critical section? what are the requirements for the solution to critical section problem? Explain Peterson's solution.

⇒ A critical section is a part of a program where shared resources are accessed. To prevent inconsistencies or unexpected behavior, only one process should execute in the critical section at any given time.

Requirements for a Solution to the critical Section

1. Mutual Exclusion:-
   - Only one process is allowed to Execute in the critical section at any given time.

2. Progress:-
   - If no process is in the critical section and some processes wish to Enter it, the selection of the process to Enter cannot be postponed indefinitely.

3. Bounded waiting:-
   - There should be a limit on the number of times other processes can Enter the critical section before a waiting process gets its turn.

Peterson's Solution is a software-based solution that satisfies all three requirements and works for two processes. It uses two variable.

1. flag[]: Indicates wheather a process wants to Enter the critical section.
2. - flag[i] = true means Process i want to Enter.

2. turn: Determines whose turn it is to Enter the critical section.

```
do {
    flag[i] = TRUE;
    turn = j
    while (flag[j] && turn == j);
        critical section
    flag[i] = FALSE;
        remainder section
} while (TRUE);
```

---

③. What is a semaphore?

→ A semaphore S is an integer variable that, apart from initialization, is accessed only through two standard atomic operations: wait and signal.

These operations were originally termed P and V.

The classical definition of wait in pseudo code is

```
wait(S) {
    while (S<=0)
        S--;
    }
```

The classical definition of signal in pseudo code is

```
Signal(S) {
    S++;
    }.
```

④ Explain Reader's- writer's problem using Semaphores.

Ans:- Readers who only read the data can work simultaneously without any problem because they don't change the data.

writers who modify the data need exclusive access, so only one writer can work at a time.

Readers and writers cannot access the resources simultaneously because that could lead to inconsistencies.

Solution with Semaphores:-

→ mutex:- Controls access to a variable that counts the active reads.

→ writelock:- Ensure only one writer or multiple readers can access the resource at any time.

→ readcount:- A counter to track the number of active readers.

---

⑤ what is resource-allocation graph? Consider an example to Explain how it is very useful in describing a deadly embrace.

→ A Resource allocation graph is a directed graph used to represent the allocation of resources to processes in a system.

It helps in visualizing and analysing the possibility of a deadlock

Components of RAG:-

1. Processes (P):- represented as Circles
2. Resources (R):- Represented as Squares.

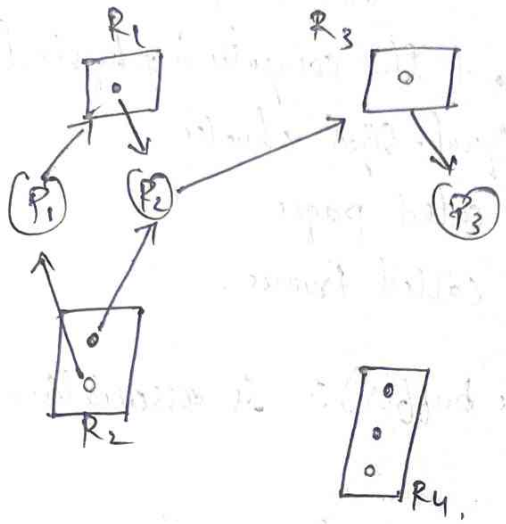
the sets P, R and E

→ $P = \{P_1, P_2, P_3\}$

$R = \{R_1, R_2, R_3, R_4\}$

$E = \{P_1 \rightarrow R_1, P_2 \rightarrow R_3, R_1 \rightarrow P_2, R_2 \rightarrow P_2, R_2 \rightarrow P_1, R_3 \rightarrow P_3\}$

Diagram



→ Graph contains no cycles, then no process in the system is deadlocked

→ If each resource type has exactly one instance, then a cycle implies that a deadlock has occurred.

module 4

① What is paging? Explain with neat diagram paging hardware with TLB.

Ans:- Paging is a method used by operating Systems to manage computer memory more effectively.

The program's memory and the computer's physical memory are divided into Equal-Sized chunks.

→ logical memory :- called pages

→ physical memory :- called Frames.

TLB (Translation look-aside buffer) :- is associative, high-speed memory.

- Each Entry in the TLB consist of two parts : a key and a value.
- TLB is used with page tables. And it contains only a few of the page-table Entries.
- When a logical address is generated by the CPU, its page number is presented to the TLB.
- If the page number is found, its frame number is immediately available and is used to access memory.
- If the page number is not in the TLB, a memory reference to the page table must be made.

When the frame number is obtained, we can use it to access memory.

(9) Explain fragmentation in detail?

→ Fragmentation in computer system refers to the inefficient use of memory or storage space caused by how memory is allocated and deallocated.

Types of fragmentation:-

1. Internal fragmentation:
2. External fragmentation:

→ Internal fragmentation:-

* Occurs when a memory block allocated to a process is larger than the process requires.
* The leftover memory within the allocated block remain unused, wasting space.

2- External Fragmentation

→ Happens when free memory is available but is not contigous

→ Both the first-fit and best-fit strategies for memory allocation suffer from External Fragmentation

---

③ Describe the steps in handling the page fault

→ we check an internal table for this process to determine whether the reference was a valid or an invalid memory-acc

→ If the reference was invalid, we terminate the process

→ we find the free frame

→ we schedule a disk operation to read the desired page into the newly allocated frame.

→ when the disk read is complete, we modify the internal table kept with the process and the page table to indicate that page is now in memory.

→ we restart the instruction that was interrupted by the trap.

④ What is Segmentation? Explain the basic method of Segmentation with an Ex?

→ Segmentation is a memory-management scheme that supports the user view of memory

- A logical address space is a collection of segments. Each segment has a name and a length.

Methods of Segmentation:-

1. Fixed-size Segmentation:- memory is divided into fixed-size segments, just like paging. Each segment has the same size.

   Ex:- program might have fixed-size segments of code, data, stack.

2. Variable size Segment:-

- Segments are divided based on the actual needs of the program.

  Ex:- Code Segment : 5000 bytes
  Data Segment :- 2000 bytes
  Stack Segment :- 1000 bytes.

3. Logical Segmentation:- this method divides into logical parts based on its structure. this part might include code, data, stack and heap segments.

4. Segmentation with Paging:-

Segmentation is combined with paging to reduce fragmentation.
The idea is to first divide the memory into segments,
then divide each segments into fixed size pages;

- It helps to make memory allocation more efficient.

---

module - 5

(1) What is access Matrix? Explain Access Matrix method of system protection with domain as objects and its implementation?

Ans:- An Access matrix is a way to manage and control who can access what in a computer system.

- It like a table where:
  Rows of the access matrix represent domains
  Colums represent objects.
- There are four domains and four objects - three files (f1, f2, f3) and one laser printer.

D1:- can read File F1 and F3
D2:- can write to File F2.
D3:- can Execute file F1 and write to file F3
D4:- Has both read and write permissions for file F1 and F3.

* Implementation of Access Matrix.

- Global Table
- Access lists for objects
- Capability lists for Domains
- A lock-key mechanism.

Diagram:

domain \ object	F_1	F_2	F_3	Printer
P_1	read		read	
P_2				print
P_3		read	Execute	
P_4	read write		read write	

(2) Define - file. List and Explain the different file attributes and operations?

Ans:- A file is a collection of data or information that is stored on a storage device, such as a hard drive, SSD, or cloud storage.

File contains various types

- Text file
- Binary files
- Document files
- Media files

File attributes:-

- Name:- human-readable name assigned to the file.
- Type:- Indicate the kind of file, such as text, image or Executable. Ex:- .txt, .jpg, .exe.
- location:- Specifies where the file is stored on the disk.
- Size:- the amount of space the file occupies on the storage device.
- Protection:- Defines who can access the file and what actions they can perform.
- Time stamps:- Creation Time, Last Access Time, Last modified Time.

File operation:-

Create:- A new file is created in storage.

Open:- A file is opened for reading, writing or both.

Read:- Data is retrieved from a file and transferred into memory.
or file system provides access to the content of a file

Write:- Here, existing content in a file can be updated or new content can be added.

Delete:- A file is permanently removed from storage, freeing up space.

Rename:- Changes the name of a file while keeping its same contents and location

(3) Explain the different allocation methods?

Ans: Linked Allocation:

- Solves the problems of contiguous allocation
- Each file is a linked list of disk blocks & blocks may be scattered any where on the disk.
- the directory contains a pointer to the first and last block blocks of the file.
- Creating a new file requires only creation of a new Entry in the directory
- writing a file causes the free-space management System to find a free block.

Advantages:

- No External fragmentation
- The size of a file need not be declared when the file is created.

Contiguous Allocation:

- It requires that Each file occupy a set of contiguous blocks on the disk
- Contiguous allocation of a file is defined by the disk address and length of the first block.
- Accessing a file that has been allocated contiguously is Easy.
- Contiguous allocation has one problem finding space for a new file.