I will first generate the answers to all the questions and then create a PDF. Here are the answers:

Unit 3: Memory Management

1. Explain various memory management strategies with an example.

Memory management strategies determine how processes are allocated and managed in the system's memory. The common strategies are:

1. **Single Contiguous Allocation**: Entire memory is allocated to a single process. Example: MS-DOS.
2. **Partitioned Memory**: Divides memory into fixed or variable-sized partitions.
   - *Fixed Partitioning*: Memory is divided into equal-sized blocks, leading to fragmentation.
   - *Variable Partitioning*: Memory is allocated dynamically, reducing fragmentation.
3. **Paging**: Divides memory into fixed-sized pages and processes into page frames, reducing fragmentation.
4. **Segmentation**: Divides memory into variable-sized segments based on logical divisions of a program.
5. **Virtual Memory**: Uses disk space as an extension of RAM, allowing execution of large processes.

Example: In paging, if a process requires 10 KB of memory and the page size is 2 KB, the process will be divided into 5 pages stored in available frames in RAM.

---

2. Page Fault Calculation using LRU

Given page reference string: **7,6,4,1,7,6,2,1,7,2**
Frames available: **3**
Least Recently Used (LRU) Page Replacement

Step	Page	Frame 1	Frame 2	Frame 3	Page Fault
1	7	7	-	-	Yes
2	6	7	6	-	Yes
3	4	7	6	4	Yes
4	1	6	4	1	Yes
5	7	4	1	7	Yes
6	6	1	7	6	Yes
7	2	7	6	2	Yes

Step	Page	Frame 1	Frame 2	Frame 3	Page Fault
8	1	6	2	1	Yes
9	7	2	1	7	Yes
10	2	1	7	2	No

Total Page Faults: 9
Hit Ratio = (Total References - Page Faults) / Total References = (10 - 9) / 10 = **0.1 (10%)**
Miss Ratio = 1 - Hit Ratio = **90%**

---

3. Discuss about thrashing in detail.

Thrashing occurs when a system spends more time swapping pages in and out of memory rather than executing processes. This happens due to high page fault rates when too many processes compete for limited memory.

Causes of Thrashing:

- High multiprogramming level
- Insufficient memory allocation
- Frequent page replacements

Solutions to Thrashing:

- Using the **Working Set Model**
- Implementing **Page Fault Frequency (PFF) Algorithm**
- Increasing physical memory

---

4. Distinguish between paging and segmentation.

Feature	Paging	Segmentation
Division	Fixed-size pages	Variable-sized segments
Basis	Physical memory	Logical program structure
Fragmentation	Internal fragmentation	External fragmentation
Address Translation	Page number + offset	Segment number + offset
Usage	Used in OS memory management	Used in programming

---

5. What is a page fault? Explain its handling with a diagram.

A **page fault** occurs when a process requests a page that is not in memory.

Steps in Handling Page Fault:

1. The CPU detects a missing page.
2. The OS identifies the page location on disk.
3. The required page is read into memory.
4. Page table is updated.
5. The process resumes execution.

Unit 4: Deadlocks and Storage Management

1. Define deadlock and explain necessary conditions for its occurrence.

A **deadlock** is a state where multiple processes are waiting indefinitely for resources held by each other.

Necessary Conditions (Coffman's Conditions):

1. **Mutual Exclusion** – Resources are non-shareable.
2. **Hold and Wait** – A process holds resources while waiting for others.
3. **No Preemption** – Resources cannot be forcibly taken.
4. **Circular Wait** – A cycle exists where each process waits for another.

---

2. Explain Banker's Algorithm.

Banker's Algorithm is used to avoid deadlocks by ensuring safe resource allocation.

Steps:

1. Check if request ≤ available resources.
2. Assume the request is granted and check if the system remains in a safe state.
3. If the state is safe, allocate the resources.

---

3. Disk Structure with Diagram

A **disk structure** consists of platters, tracks, sectors, and read/write heads.

---

4. RAID and its Classifications

RAID (Redundant Array of Independent Disks) improves performance and fault tolerance.

RAID Levels:

- **RAID 0 –** Striping (no redundancy, high speed)
- **RAID 1 –** Mirroring (data redundancy)
- **RAID 5 –** Striping with parity (fault tolerance)
- **RAID 6 –** Double parity (better fault tolerance)

5. Single-Level vs Two-Level Directory Structure

Feature	Single-Level Directory	Two-Level Directory
Hierarchy	Flat structure	User-specific directories
File Naming	Unique across system	Unique per user
Access Control	Less secure	More secure

Unit 5: System Protection and Security

1. System Protection: Goals and Principles

Goals:

- Prevent unauthorized access
- Ensure process isolation
- Control resource access

Principles:

- Least privilege
- Separation of duty

2. Access Matrix for System Protection

An **Access Matrix** defines permissions for users and resources.

Resource/User	User A	User B
File X	Read	Write
File Y	Write	Read

3. Computer Security Classifications

- **A (Verified Protection)**: High security
- **B (Mandatory Security)**: Access control policies
- **C (Discretionary Security)**: Basic protection
- **D (Minimal Security)**: No security enforcement

4. Firewall and its Protection

A **firewall** filters network traffic based on rules to protect against unauthorized access.

5. Program and Network Threats

Program Threats:

- Virus
- Worms
- Trojan Horses

Network Threats:

- Denial of Service (DoS)
- Man-in-the-middle attacks