

ASSIGNMENT FILE – 2

SECTION 1: THEORY QUESTIONS

Q1. Explain different types of Software Architecture patterns.

Answer: Software architecture patterns provide reusable solutions to common problems in software system design. One widely used pattern is the layered architecture, where the system is divided into presentation, business, and data layers. This separation improves maintainability and scalability.

Another important pattern is the client-server architecture, where clients request services and servers respond. This model is commonly used in web applications and distributed systems. Microservices architecture is a modern approach where applications are broken into small, independent services.

Event-driven architecture is based on producing and consuming events. It is useful in real-time systems and asynchronous processing. Service-oriented architecture focuses on reusable services that communicate over networks.

Each architecture pattern has advantages and trade-offs. Choosing the correct architecture improves performance, security, and system longevity.

Q2. Explain the concept of normalization in DBMS.

Answer: Normalization is the process of organizing data in a database to reduce redundancy and improve data integrity. It involves dividing large tables into smaller ones and defining relationships between them. The goal is to eliminate anomalies during insert, update, and delete operations.

The first normal form ensures atomic values. Second normal form removes partial dependency.

Third normal form eliminates transitive dependency. Higher normal forms further optimize data storage.

Normalization improves consistency and reduces duplication. However, excessive normalization can affect performance. Therefore, a balance between normalization and denormalization is often required in real-world applications.

Q3. Describe process scheduling in operating systems.

Answer: Process scheduling is the method by which an operating system decides which process should execute at a given time. The scheduler allocates CPU time among processes to ensure fairness and efficiency.

Scheduling algorithms include First Come First Serve, Shortest Job First, Priority Scheduling, and Round Robin. Each algorithm has strengths and weaknesses depending on workload.

Effective scheduling improves system responsiveness and resource utilization. Modern operating systems use hybrid scheduling approaches to balance performance.

Q4. Explain TCP/IP model layers.

Answer: The TCP/IP model is a conceptual framework used for network communication. It consists of four layers: Application, Transport, Internet, and Network Access.

The application layer supports user-level protocols like HTTP and FTP. The transport layer ensures reliable data delivery using TCP or UDP. The internet layer handles routing using IP. The network access layer manages hardware transmission.

TCP/IP is the foundation of internet communication and enables interoperability between systems.

Q5. Explain virtualization in cloud computing.

Answer: Virtualization allows multiple virtual machines to run on a single physical server. It maximizes hardware utilization and reduces operational costs.

Hypervisors manage virtual machines and allocate resources dynamically. Virtualization supports scalability, fault isolation, and disaster recovery.

It is a core technology behind cloud computing platforms and enables efficient resource management.

SECTION 2: CODING QUESTIONS

Q6. Implement a queue using Python lists.

```
Answer: class Queue: def __init__(self): self.items = []
def enqueue(self, item): self.items.append(item)
def dequeue(self): if self.items: return self.items.pop(0) return None
def is_empty(self): return len(self.items) == 0
def size(self): return len(self.items)
q = Queue() q.enqueue(1) q.enqueue(2) print(q.dequeue()) print(q.size())
```

Q7. Write a Python program for user registration validation.

Answer: def register_user(username, password): if len(password) < 6: return False if not username: return False return True

```
print(register_user("student", "pass123"))
```

Q8. Write a program to count word frequency.

Answer: def word_count(text): words = text.split() freq = {} for w in words: freq[w] = freq.get(w, 0) + 1 return freq

```
print(word_count("this is a test this is"))
```

Q9. Implement linear search in Python.

Answer: def linear_search(arr, target): for i in range(len(arr)): if arr[i] == target: return i return -1

```
print(linear_search([1,2,3,4], 3))
```

Q10. Write a Python program to check palindrome.

Answer: def is_palindrome(s): return s == s[::-1]

```
print(is_palindrome("madam"))
```

SECTION 3: PROGRAMMING QUESTIONS

Q11. Explain Python lists.

Answer: Lists are ordered, mutable collections used to store multiple items in Python.

Q12. Explain tuples in Python.

Answer: Tuples are ordered but immutable collections, useful for fixed data.

Q13. Explain dictionaries.

Answer: Dictionaries store data in key-value pairs for fast lookup.

Q14. Explain conditional statements.

Answer: Conditional statements execute code based on conditions using if, elif, and else.

Q15. Explain exception handling.

Answer: Exception handling manages runtime errors using try, except, and finally blocks.