Course Title: Theory of Computation **Full Marks: 60+20+20**
Course No: CSC 257 **Pass Marks: 24+8+8**
Nature of the Course: Theory + Lab **Credit Hours: 3**
Year: Second, Semester: Fourth

Course Description: This course presents a study of Finite State Machines and their languages. It covers the details of finite state automata, regular expressions, context free grammars. More, the course includes design of the Push-down automata and Turing Machines. The course also includes basics of undecidabilty and intractability.

Course Objectives: The main objective of the course is to introduce concepts of the models of computation and formal language approach to computation. The general objectives are to, introduce concepts in automata theory and theory of computation, design different finite state machines and grammars and recognizers for different formal languages, identify different formal language classes and their relationships, and determine the decidability and intractability of computational problems.

Detail Syllabus

Chapters / Units	Teaching Methodology	Teaching Hours
Unit I: Basic Foundations 1.1. Review of Set Theory, Logic, Functions, Proofs 1.2. Automata, Computability and Complexity: Complexity Theory, Computability Theory, Automata Theory 1.3. Basic concepts of Automata Theory: Alphabets, Power of Alphabet, Kleen Closure Alphabet, Positive Closure of Alphabet, Strings, Empty String, Suffix, Prefix and Substring of a string, Concatenation of strings, Languages, Empty Language, Membership in Language	**Class Lecture**	**3 Hours**
Unit II: Introduction to Finite Automata 2.1. Introduction to Finite Automata, Introduction of Finite State Machine 2.2. Deterministic Finite Automata (DFA), Notations for DFA, Language of DFA, Extended Transition Function of DFA Non-Deterministic Finite Automaton (NFA), Notations for NFA, Language of NFA, Extended Transition 2.3. Equivalence of DFA and NFA, Subset-Construction	**Class Lecture** **+** **Lab Session**	**8 Hours**

2.4. Method for reduction of NFA to DFA, Theorems for equivalence of Language accepted by DFA and NFA: For any NFA, N = (Q_N, $\sum$, δ_N, q_0, F_N) accepting language L $\subseteq \sum^*$ there is a DFA D = (Q_D, $\sum$, δ_D, q_0',F_D) that also accepts L i.e. L (N) = L (D), A language L is accepted by some NFA if L is accepted by some DFA. 2.5. Finite Automaton with Epsilon Transition (ε - NFA), Notations for ε - NFA, Epsilon Closure of a State, Extended Transition Function of ε – NFA, Removing Epsilon Transition using the concept of Epsilon Closure, Equivalence of NFA and ε –NFA, Equivalence of DFA and ε – NFA 2.6. Finite State Machines with output: Moore Machine and Mealy Machines, Illustration of the Moore and Mealy Machines		
Unit III: Regular Expressions 3.1. Regular Expressions, Operators of Regular Expressions (Union, Concatenation, Kleen), Regular Languages and their applications, Algebraic Rules for Regular Expressions 3.2. Equivalence of Regular Expression and Finite Automata, Reduction of Regular Expression to ε–NFA, Conversion of DFA to Regular Expression, Arden's Theorem 3.3. Properties of Regular Languages, Pumping Lemma for regular expression, Application of Pumping Lemma, Closure Properties of Regular Languages over (Union, Intersection , Complement), Minimization of Finite State Machines: Table Filling Algorithm	**Class Lecture** **+** **Lab Session**	**6 Hours**
Unit IV: Context Free Grammar 4.1. Introduction to Context Free Grammar (CFG), Components of CFG, Use of CFG, Context Free Language (CFL) 4.2. Types of derivations: Bottomup and Topdown approach, Leftmost and Rightmost, Sentential Form (Left, Right), Language of a grammar 4.3. Parse tree and its construction, Ambiguous	**Class Lecture** **+** **Lab Session**	**9 hours**

grammar, Use of parse tree to show ambiguity in grammar, Inherent Ambiguity 4.4. Regular Grammars: Right Linear and Left Linear, Equivalence of regular grammar and finite automata 4.5. Simplification of CFG: Removal of Useless symbols, Nullable Symbols, and Unit Productions, Chomsky Normal Form (CNF), Greibach Normal Form (GNF), Backus-Naur Form (BNF) 4.6. Context Sensitive Grammar, Chomsky Hierarchy(Type 0, 1, 2, 3) , Pumping Lemma for CFL, Application of Pumping Lemma, Closure Properties of CFL		
Unit V: Push Down Automata 5.1. Introduction to Push Down Automata (PDA), Representation of PDA, Operations of PDA, Move of a PDA, Instantaneous Description for PDA 5.2. Deterministic PDA, Non Deterministic PDA, Acceptance of strings by PDA, Language of PDA 5.3. Construction of PDA by Final State , Construction of PDA by Empty Stack, Conversion of PDA by Final State to PDA accepting by Empty Stack and vice-versa, Conversion of CFG to PDA, Conversion of PDA to CFG	**Class Lecture** + **Lab Session**	**7 Hours**
Unit VI: Turing Machines 6.1. Introduction to Turing Machines (TM), Notations of Turing Machine, Language of a Turing Machine, Instantaneous Description for Turing Machine, Acceptance of a string by a Turing Machines 6.2. Turing Machine as a Language Recognizer, Turing Machine as a Computing Function, Turing Machine with Storage in its State, Turing Machine as a enumerator of stings of a language, Turing Machine as Subroutine	**Class Lecture** + **Lab Session**	**10 Hours**

6.3. Turing Machine with Multiple Tracks, Turing Machine with Multiple Tapes, Equivalence of Multitape-TM and Multitrack-TM, Non-Deterministic Turing Machines, Restricted Turing Machines: With Semi-infinite Tape, Multistack Machines, Counter Machines 6.4. Curch Turing Thesis, Universal Turing Machine, Turing Machine and Computers, Encoding of Turing Machine, Enumerating Binary Strings, Codes of Turing Machine, Universal Turing Machine for encoding of Turing Machine		
Unit VII: Undecidability and Intractability 7.1. Computational Complexity, Time and Space complexity of a Turing Machine, Intractability 7.2. Complexity Classes, Problem and its types: Absract, Decision, Optimization 7.3. Reducibility, Turing Reducible, Circuit Satisfiability, Cooks Theorem 7.4. Undecidability, Undecidable Problems: Post's Correspondence Problem, Halting Problem and its proof, Undecidable Problem about Turing Machines	**Class Lecture** **+** **Lab Session**	**5 Hours**

Text Books

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, **Introduction to Automata Theory, Languages, and Computation,** 3rd Edition, Pearson - Addison-Wesley.

Reference Books

1. Harry R. Lewis and Christos H. Papadimitriou, ***Elements of the Theory of Computation***, 2nd Edition, Prentice Hall.

2. Michael Sipser, **Introduction to the Theory of Computation,** 3rd Edition, Thomson Course Technology

3. Efim Kinber, Carl Smith, **Theory of Computing: A Gentle introduction,** Prentice- Hall.

4. John Martin, **Introduction to Languages and the Theory of Computation,** 3rd Edition, Tata McGraw Hill.

Laboratory Work Manual

Student should write programs and prepare lab sheets for most of the units in the syllabus. Majorly, students should practice design and implementation of Finite State Machines viz. DFA, NFA, PDA, and Turing Machine. Students are highly recommended to construct Tokenizers/ Lexical analyzer over/for some language. The nature of programming can be decided by the instructor and students as per their comfort. The instructors have to prepare lab sheets for individual unit covering the concept of all units as per the requirement. The sample lab sessions can be as following descriptions;

Unit I: Basic Foundations (5 Hrs)

- Write programs for illustrating the concepts of Strings, Prefix, Suffix and Substring of a String.

Unit II & III: Introduction to Finite Automata and Regular Expressions (14 Hrs)

- Write programs for illustrating the concepts of
    - Determinstic Finite Automata
    - Non-Deterministic Finite Automata
- Write programs for implementing Tokenizers like for valid C-identifiers, Keywords, e-mail validators, phone number etc.
- Write programs that implement NFA for text search.
- Write programs for implementing regular expressions.

Unit IV & V: Context Free Grammar and Push Down Automata (14 Hrs)

- Write Program for simulation of Leftmost/Rightmost Derivations.
- Write Program for Parse Tree Contruction.
- Write programs for illustrating the concepts of context free grammar and its accptance using the concepts of Push Down Automata
    - Acceptance by Final State
    - Acceptance by Empty Stack

Unit VI: Turing Machines (12 Hrs)

- Write programs for illustrating the concepts of Turing Machine as a Language Recognizer.

Model Question
Tribhuvan University
Institute of Science and Technology

Course Title: Theory of Computation
Course No: CSC257
Level: B. Sc CSIT Second Year/ Fourth Semester

Full Marks: 60
Pass Marks: 24
Time: 3 Hrs

Section A
Long Answer Questions

Attempt any Two questions. *[2*10=20]*

1. Define the extended transition function of DFA. Draw a DFA accepting language L= $\{1^n \mid n=2,3,4\ldots\ldots\}$. Show acceptance of strings 1110011 and 1110 using extended transition function. [2+4+4]
2. What is deterministic pushdown automaton? Configure a pushdown automaton accepting the language, L= $\{wCw^R \mid w \in (a,b)^*\}$. Show instantaneous description of strings abbCbba and baCba. [2+4+4]
3. How a Turing Machine works? Construct a Turing Machine accepting the language, L= $\{ (^n)^n \}$. Also show the transition diagram of the machine. Illustrate whether a string (( )) is accepted by the Turing Machine or not. [2+6+2]

Section B
Short Answer Questions

Attempt any Eight questions. *[8*5=40]*

4. When a grammar is said to be in CNF? Convert following grammar to CNF; [1+4]
   $S \rightarrow 1A \mid 0B \mid \epsilon$
   $A \rightarrow 1AA \mid 0S \mid 0$
   $B \rightarrow 0BB \mid 1 \mid A$
   $C \rightarrow CA \mid CS$
5. Define epsilon NFA. Configure equivalent epsilon NFA for the regular expression (ab U a)*. [1+4]
6. Differentiate Kleen Closure from Positive Closure. For $\sum = \{0,1\}$, compute $\sum^*$ and $\sum^2$. [3+2]
7. Write the regular expression over {0, 1} for strings [2.5+2.5]
   a. not ending with 0.
   b. of length at least 3 that ends with 00.
8. What is undecidable problem? Define Post's Correspondence Problem with an example. [1+4]
9. How pumping lemma can be used to prove that any language is not a regular language? Show that language, $L=\{0^r 1^r \mid n \geq 0\}$ is not a regular language. [4+1]
10. Discuss how Turing Machine with multiple tracks differs from a Turing Machine with multiple tapes. [5]
11. How context free grammars are defined? Write a context free grammar over {0,1}, where the strings start and end with the same symbol. [2+3]
12. What is halting problem? How can you argue that halting problem is undecidable? [1+4]

Course Title: Computer Networks **Full Marks: 60+20+20**
Course No: CSC258 **Pass Marks: 24+8+8**
Nature of the Course: Theory + Lab **Credit Hours: 3**
Year: Second, Semester: Fourth

Course Description: This course introduces concept of computer networking and discuss the different layers of networking model.

Course Objective: The main objective of this course is to introduce the understanding of the concept of computer networking with its layers, topologies, protocols & standards, IPv4/IPv6 addressing, Routing and Latest Networking Standards.

Unit	Contents	Hour
1. Introduction to Computer Network [6 Hour]	1.1. Definitions, Uses, Benefits 1.2. Overview of Network Topologies *Mesh, Star, Tree, Bus* 1.3. Overview of Network Types *LAN, PAN, CAN, MAN, WAN*	1
	1.4. Networking Types P2P, Multipoint, Client/Server 1.5. Overview of Protocols and Standards *Protocols: Syntax, semantics, timing; Standards: De facto, De jure; Standards Organizations*	1.5
	1.6. OSI Reference Model 1.7. TCP/IP Model and its comparison with OSI	2.5
	1.8. Connectionless and Connection-Oriented Network Services *Basic working Mechanism* 1.9. Internet, ISPs, Backbone Network Overview Basic concept of Internet and ISPs, Bus backbone, Star backbone, connecting remote LANs	1
2. Physical Layer and Network Media [4 Hour]	2.1. Network Devices *Repeater, Hub, Switch, Bridge, Router* 2.2. Different types of transmission medias *Wired: twisted pair, coaxial, fiber optic, Wireless: Radio waves, micro waves, infrared* 2.3. Ethernet Cable Standards *UTP, Fiber cable standards*	1.5
	2.4. Circuit, Message & Packet Switching	2
	2.5. ISDN *Interface and Standards*	0.5
3. Data Link Layer [8 Hour]	3.1. Function of Data Link Layer (DLL) 3.2. Overview of Logical Link Control (LLC) and Media Access Control (MAC) 3.3. Framing and Flow Control Mechanisms *Stop-and-wait ARQ, Piggybacking, Go-Back-N ARQ, Selective Repeat ARQ*	3

	3.4. Error Detection and Correction techniques *Parity checks, Cheksumming Methods, CRC, Hamming code* 3.5. Channel Allocation Techniques *ALOHA, Slotted ALOHA, CSMA, CSMACD,CSMA/CA* 3.6. Ethernet Standards *802.3 CSMA/CD, 802.4 Token Bus, 802.5 Token Ring*	3
	3.7. Wireless LAN *Spread Spectrum, Bluetooth, Wi-Fi* 3.8. Overview Virtual Circuit Switching, Frame Relay & ATM 3.9. DLL Protocol *HDLC, PPP*	2
4. Network Layer [10 Hour]	4.1. Introduction and Functions 4.2. IPv4 Addressing 4.3. Class-full and Classless Addressing 4.4. IPv4 Sub-netting/ Super-netting 4.5. IPv6 Addressing and its Features 4.6. IPv4 and IPv6 Datagram Formats 4.7. Comparison of IPv4 and IPv6 Addressing 4.8. NATing 4.9. Example Addresses *Unicast, Multicast and Broadcast*	4
	4.10. Routing 4.10.1. Introduction and Definition 4.10.2. Types of Routing *Static vs Dynamic, Unicast vs Multicast, Link State vs Distance Vector, Interior vs Exterior* 4.10.3. Path Computation Algorithms *Bellman Ford, Dijkstra's* 4.10.4. Routing Protocols *RIP, OSPF & BGP*	4
	4.11. Overview of IPv4 to IPv6 Transition Mechanisms 4.12. Overview of ICMP/ICMPv6 4.13. Overview of Network Traffic Analysis 4.14. Security Concepts *Firewall & Router Access Control*	2
5. Transport Layer [6 Hour]	5.1. Introduction, Functions and Services 5.2. Transport Protocols *TCP, UDP and Their Comparisons* 5.3. Connection Oriented and Connectionless Services	1
	5.4. Congestion Control *Open Loop & Closed Loop, TCP Congestion Control* 5.5. Traffic Shaping Algorithms 5.6. *Techniques to improve QOS* *Scheduling, traffic shaping, resource reservation, admission control*	2.5

	5.7. Queuing Techniques for Scheduling 5.8. Introduction to Ports and Sockets, Socket Programming *Socket programming with UDP and TCP (e.g. client Server Application)*	2.5
6. Application Layer [7 Hour]	6.1. Introduction and Functions 6.2. Web & HTTP *Overview of HTTP, Non-Persistent and Persistent Connections, HTTP Message Format*	2
	6.3. DNS and the Query Types *Services provided by DNS, Overview of how DNS works, DNS records and messages* 6.4. File Transfer and Email Protocols *FTP, SFTP, SMTP, IMAP, POP3*	3
	6.5. Overview of Application Server Concepts *Proxy, Web, Mail* 6.6. Network Management *SNMP and Transport mapping*	2
7. Multimedia & Future Networking [4 Hour]	7.1. Overview Multimedia Streaming Protocols *SCTP*	1
	7.2. Overview of SDN and its Features, Data and Control Plane	1
	7.3. Overview of NFV	1
	7.4. Overview of NGN	1

Text Books:

1. Data Communications and Networking, 4^{th} Edition, Behrouz A. Forouzan. McGraw-Hill
2. Computer Networking; A Top Down Approach Featuring The Internet, 2nd Edition, Kurose James F., Ross W. Keith PEARSON EDUCATION ASIA

Laboratory works:

The lab activities under this subject should accommodate at least the following

S.N.	Contents
1.	Understanding of Network equipment, wiring in details
2.	Practice on basic Networking commands (ifconfig/ipconfig, tcpdump, netstat, dnsip, hostname, route)
3.	Overview of IP Addressing and sub-netting, static ip setting on Linux/windows machine, testing
4.	Introduction to Packet Tracer, creating of a LAN and connectivity test in the LAN, creation of VLAN and VLAN trunking.
5.	Basic Router Configuration, Static Routing Implementation
6.	Implementation of Dynamic/interior/exterior routing (RIP, OSPF, BGP)
7.	Firewall Implementation, Router Access Control List (ACL)
8.	Packet capture and header analysis by wire-shark (TCP,UDP,IP)
9.	Basic concept of DNS, Web, FTP (shall use packet tracer, GNS3)

Model Question

Bachelor Level/ Second Year/ Fourth Semester/ Science Full Marks: 60
Computer Networks (CSC 258) Pass Marks: 24
Time: 3 hours.

Candidates are required to give their answers in their own words as for as practicable.
The figures in the margin indicate full marks.

Group A (Long Answer Question Section)
Attempt any TWO questions. **(2x10=20)**

1. Suppose you are assigned to design a LAN for an office having 3 departments. Each department will have 50 computers locating in 10 rooms each equipped with 5 computers. Make your own justification while selecting connecting devices and accessories.
2. Highlight on the importance of routing algorithm. Explain Distance Vector Routing algorithm and compare it with link state routing.
3. Explain various congestion control approaches.

Group B (Short Answer Question Section)

Attempt any EIGHT questions. **(8x5=40)**

4. Is 192.16.144.64/27 a host, network or broadcast address? In which layer of OSI model do HUB, Switch and Router operate on.9+99999999999999
5. Describe the working procedure of Token bus and Token ring.
6. Why do you think network traffic analysis is carried out? How does IPv6 overcome the disadvantages of IPv4?
7. Find Hamming Code for data 01100111.
8. Differentiate between frame relay and ATM.
9. What is the function of proxy server? Explain about electronic mail.
10. Demonstrate the use of socket programming for creating network application using UDP and TCP with necessary diagrams.
11. Explain DNS with reference to its hierarchy and records.
12. Write Short Notes (Any Two):
    a) Firewall
    b) Packet Switching
    c) NGN

Operating Systems

Course Title: Operating Systems **Full Marks:**60+ 20+20

Course No: CSC259 **Pass Marks:** 24+8+8

Nature of the Course: Theory + Lab **Credit Hrs:** 3

Course Description: This course includes the basic concepts of operating system components. It consists of process management, deadlocks and process synchronization, memory management techniques, File system implementation, and I/O device management principles. It also includes case study on Linux operating system.

Course Objectives

- Describe need and role of operating system.
- Understand OS components such a scheduler, memory manager, file system handlers and I/O device managers.
- Analyze and criticize techniques used in OS components
- Demonstrate and simulate algorithms used in OS components
- Identify algorithms and techniques used in different components of Linux

Course Contents:

Unit	Teaching Hour	References
Unit 1: Operating System Overview (4)		
1.1 Introduction: Definition, Two views of operating system, Evolution/*History* of operating system, Types of OS (*Mainframe, Server, Multiprocessor, PC, Real-Time, Embedded, Smart Card Operating Systems*), Operating System Structures	**2 Hour**	
1.2 System Calls: Definition, Handling System Calls, *System calls for Process, File, and Directory Management*, System Programs, The Shell, Open Source Operating Systems	**2 Hour**	
Unit 2: Process Management (10)		

2.1 Introduction: Process vs Program, Multiprogramming, Process Model, Process States, Process Control Block/*Process Table*. **2.2 Threads:** Definition, Thread vs Process, *Thread Usage*, User and Kernel Space Threads. **2.3 Inter Process Communication:** Definition Race Condition, Critical Section **2.4 Implementing Mutual Exclusion**: Mutual Exclusion with Busy Waiting (Disabling Interrupts, Lock Variables, Strict Alteration, Peterson's Solution, Test and Set Lock), Sleep and Wakeup, Semaphore, Monitors, Message Passing **2.5 Classical IPC problems**: Producer Consumer, Sleeping Barber, and Dining Philosopher Problem **2.6 Process Scheduling**: Goals, Batch System Scheduling (First-Come First-Served, Shortest Job First, Shortest Remaining Time Next), Interactive System Scheduling (Round-Robin Scheduling, Priority Scheduling, Multiple Queues), Overview of Real Time System Scheduling (*No need to discuss any real time system scheduling algorithm*)	**1 Hour** **1 Hour** **1 Hour** **3 Hour** **1 Hour** **3 Hour**	
Unit 3: Process Deadlocks (6)		
3.1 Introduction: Definition, Deadlock Characterization, Preemptable and Non-Preemptable Resources, Resource–	**1.5 Hour**	

Allocation Graph, *Necessary* Conditions for Deadlock **3.2 Handling Deadlocks**: Ostrich Algorithm, Deadlock prevention, *Safe and Unsafe States*, Deadlock Avoidance (*Bankers algorithm for Single and Multiple Resource Instances*), , Deadlock Detection (*For Single and Multiple Resource Instances*), Recovery From Deadlock (Through Preemption and Rollback)	**4.5 Hour**	
Unit 4: Memory Management (8)		
4.1 Introduction: Monoprogramming vs Multi-programming, Modelling Multi-programming, Multiprogramming with fixed and variable partitions, Relocation and Protection. **4.2 Space Management:** *Fragmentation and Compaction*, Memory management (Bitmaps & Linked-list), Memory Allocation Strategies **4.3 Virtual Memory:** Paging, Page Table, Page Table Structure, *Pages and Frames*, Handling Page Faults, TLB's **4.4 Page Replacement Algorithms**: *Hit Rate and Miss Rate*, Concept of Locality of Reference, FIFO, Belady's Anomaly, Second Chance, LRU, Optimal, LFU, Clock, WS-Clock. **4.5 Segmentation**: Why Segmentation, *Drawbacks of Segmentation*, Segmentation	**1 Hour** **1 Hour** **2 Hour** **3 Hour** **1 Hour**	

with Paging(MULTICS)		
Unit 5: File Management (6)		
5.1 File Overview: File Naming, File Structure, File Types, File Access, File Attributes, File Operations, Single Level, Two Level and Hierarchical Directory Systems, File System Layout. **5.2 Implementing Files:** Contiguous allocation, Linked List Allocation, *Linked List Allocation using Table in Memory/ File Allocation Table*, Inodes. **5.3 Directory:** Directory Operations, Path Names, Directory Implementation, Shared Files **5.4 Free Space Management:** Bitmaps, Linked List	**1 Hour** **3 Hour** **1 Hour** **1 hour**	
Unit 6: Device Management (6)		
6.1 Introduction: Classification of IO devices, Controllers, Memory Mapped IO, DMA Operation, Interrupts **6.2 IO Handling**: Goals of IO Software, Handling IO(*Programmed IO, Interrupt Driven IO, IO using DMA*), IO Software Layers (Interrupt Handlers, Device Drivers) **6.3 Disk Management:** Disk Structure, Disk Scheduling (*FCFS, SSTF, SCAN, CSCAN, LOOK, CLOOK*), Disk Formatting (*Cylinder Skew, Interleaving, Error handling*), RAID	**1 Hour** **2 Hour** **3 Hour**	
Unit 7: Linux Case Study (5)		
7.1 History, Kernel Modules, Process	**5 Hour**	

Management, Scheduling, Inter-process Communication, Memory Management, File System Management Approaches, Device Management Approaches.		

Text Book

- Modern Operating Systems: Andrew S. Tanenbaum, PH1 Publication, Third edition, 2008

Reference

- Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, "Operating System Concepts", John Wiley & Sons (ASIA) Pvt. Ltd, Seventh edition, 2005.
- Harvey M. Deitel, Paul J. Deitel, and David R. Choffnes, "Operating Systems, Prentice Hall, Third edition, 2003.

Laboratory Work

The laboratory work includes solving problems in operating system. The lab work should include;

1 Demonstration of basic Linux Commands
2 Process creation and termination, thread creation and termination
3 Simulation of IPC techniques
4 Simulation process Scheduling algorithms
5 Simulation of deadlock avoidance and deadlock detection algorithms
6 Simulation of page replacement algorithms
7 Simulation of File allocation techniques
8 Simulate free space management techniques
9 Simulation of disk scheduling algorithms

Model Question

Long Questions

Attempt any two questions. (2 × 10 = 20)

1 What is sleep and wakeup? Demonstrate problem with suitable code snippet and illustration.

2 When page fault occurs and how it is handled? Demonstrate Second Chance, and LRU page replacement algorithm for memory with three frames and following reference string: *1,3,7,4,5,2,3,6,4,5,7,8, 5,1,4*

3 What is Inode? Why it is superior to other file allocation approaches? Consider 20-GB disk with 8-KB block size. How much memory space will be occupied if contiguous, and File allocation table is used for file allocation. Assume that each FAT entry takes 4 byte.

Short Questions

Attempt any eight questions. (8 × 5 = 40)

4 Define the terms shell and system call? How it is handled? Illustrate with suitable example.

5 What are main goals of interactive system scheduling? Discuss priority scheduling along with its pros and cons.

6 How starvation differs from deadlock? Consider the following situation of processes and resources:

Process	**Has**	**Max**
P1	2	6
P2	1	5
P3	2	5
P4	2	6

Free=3

- What will happen if process P3 requests 1 resource?
- What will happen if process P4 requests 1 resource?

7 Consider a virtual memory and physical memory of size 128-MB and 32-MB respectively. Assume that page size is 4-KB. What will be the number of bits required for page number, frame number, and offset? Find physical address for the virtual address 20500. (Assume that value at index 5 of page table is 2)

8 Define the term race condition? Justify that race condition leads data loss or incorrect data.

9 Explain directory implementation techniques employed in operating systems briefly.

10 What is the main purpose of disk scheduling algorithms? Which disk scheduling technique is best but impractical? Explain the algorithm with example.

11 How threads differ from processes? Explain thread usages.

12 Write short notes on:

a) Linux Scheduling

b) Fragmentation

Database Management System

Course Title: Database Management System
Course No: CSC260
Nature of the Course: Theory + Lab
Semester: IV

Full Marks: 60 + 20 + 20
Pass Marks: 24 + 8 + 8
Credit Hrs: 3

Course Description: The course covers the basic concepts of databases, database system concepts and architecture, data modeling using ER diagram, relational model, SQL, relational algebra and calculus, normalization, transaction processing, concurrency control, and database recovery.

Course Objective: The main objective of this course is to introduce the basic concepts of database, data modeling techniques using entity relationship diagram, relational algebra and calculus, basic and advanced features SQL, normalization, transaction processing, concurrency control, and recovery techniques.

Detail Syllabus:

Unit 1	**Database and Database Users**	**Teaching Hours (2)**
Introduction	Traditional file processing system; Definition of database and database management system with example	1 hr
Characteristics of the Database Approach	Self-describing nature of a database system; Insulation between programs and data, and data abstraction; Support of multiple views of the data; Sharing of data and multiuser transaction processing	
Actors on the Scene	Database administrators; Database designers; End users; System Analysts and Application Programmers	
Workers behind the Scene	DBMS system designers and implementers; Tool developers; Operators and maintenance personnel	1 hr
Advantages of Using the DBMS Approach	Controlling redundancy; Restricting unauthorized access; Providing persistent storage; Providing storage structures and search techniques for efficient query processing; Providing backup and recovery; providing multiple user interfaces; Enforcing integrity constraints; Reduced application development time; Flexibility; Availability of up-to-date information; Economies of scale	
Unit 2	**Database System – Concepts and Architecture**	**Teaching Hours (3)**
Data Models, Schemas, and Instances	Definition of data abstraction and data model; Categories of data models (high level, low level, and representational data models) – Introduction to entity-relationship model, relational data model, network data model, hierarchical model, network model, object data model, and self-describing data models; Concept of schema and instance	1 hr

Three-Schema Architecture and Data Independence	Concept of three-schema architecture; Logical and physical data independence	1 hr
Database Languages and Interfaces	Concept of DDL, SDL, VDL, DML, procedural and non-procedural languages; Concept of interfaces	
The Database System Environment	Concept of database system environment	
Centralized and Client/Server Architectures for DBMSs	Basics of centralized and client/server architectures	1 hr
Classification of Database Management Systems	Classification based on data models, number of users, number of sites, cost and type of access path	
Unit 3	**Data Modelling Using the Entity-Relational Model**	**Teaching Hours (6)**
Using High-Level Conceptual Data Models for Database Design	Concept of conceptual design	2 hrs
Entity Types, Entity Sets, Attributes, and Keys; Relationship Types, Relationship Sets, Roles, and Structural Constraints	Concept of entity types, entity sets, attributes, and keys; Concept of relationship types and relationship sets, roles and constraints	
Weak Entity Types	Concept of weak entity types and partial keys	
ER Diagrams, Naming Conventions, and Design Issues	Drawing ER diagrams using ER notations, naming conventions and design issues	2 hrs
Relationship Types of Degree Higher Than Two	Concept of higher degree relationships	
Subclasses, Superclasses, and Inheritance	Concept of enhanced ER (EER) model, superclasses, subclasses and subclasses	2 hrs
Specialization and Generalization	Concept of specialization and generalization	
Constraints and Characteristics of Specialization and Generalization	Different constraints and characteristics of specialization and generalization	
Unit 4	**The Relational Data Model and Relational Database Constraints**	**Teaching Hours (3)**
Relational Model Concepts	Concept of domain, attributes, tuples, and relations; Characteristics of relations; Relational model notation	2 hrs
Relational Model Constraints and Relational Database	Different categories of constraints; Domain constraints; Key and NULL values constraints;	

Schemas	Relational databases and relational database schemas; Entity integrity, referential integrity, and foreign key	
Update Operations, Transactions, and Dealing with Constraint Violations	Concept of insert, delete, and update operations; Concept of transactions	1 hr
Unit 5	**The Relational Algebra and Relational Calculus**	**Teaching Hours (5)**
Unary Relational Operations: SELECT and PROJECT	Concept and example of SELECT and PROJECT operations; Sequences of operations; RENAME operation	3 hrs
Relational Algebra Operations from Set Theory	Concept and example of UNION, INTERSECTION, MINUS, and CARTESIAN PRODUCT operations	
Binary Relational Operations: JOIN and DIVISION	Concept and example of JOIN operation and its variations; Concept and example of DIVISION operation	
Additional Relational Operations	Concept of generalized projection, aggregate functions, OUTER JOIN operations, and OUTER UNION operation	2 hrs
the Tuple Relational Calculus	Introduction to tuple relational calculus with examples	
the Domain Relational Calculus	Introduction to domain relational calculus with examples	
Unit 6	**SQL**	**Teaching Hours (8)**
Data Definition and Data Types	CREATE TABLE command; Attribute data types and domains; ALTER and DROP commands	1 hr
Specifying Constraints	Attribute constraints and attribute defaults; Key and referential integrity constraints	1 hr
Basic Retrieval Queries	SELECT-FROM-WHERE structure; Ambiguous attribute names, aliasing, renaming, and tuple variables; Unspecified WHERE clause and use of asterisk (*); Pattern matching and arithmetic operators	5 hrs
Complex Retrieval Queries	Comparisons involving NULL; Nested queries	
INSERT, DELETE, and UPDATE Statements	Concept and example of INSERT, DELETE, and UPDATE commands	1 hr
Views	Concept of views; CREATE VIEW command	
Unit 7	**Relational Database Design**	**Teaching Hours (7)**
Relational Database Design Using ER-to-Relational Mapping	Converting ER / EER models to relations with examples	1 hr
Informal Design Guidelines for Relational Schemas	Imparting clear semantics to attributes in relations; Redundant information in tuples and update anomalies; NULL values in tuples; Generation of	2 hrs

	spurious tuples	
Functional Dependencies	Definition and concept of functional dependencies with example	2 hrs
Normal Forms Based on Primary Keys	Concept of normalization; Practical use of normal forms; Keys and attributes participating in keys; Concept of first, second, and third forms with example	
General Definitions of Second and Third Normal Forms	General definitions of second and third normal forms	1 hr
Boyce-Codd Normal Form	Concept and example of boyce-codd normal form	
Multivalued Dependency and Fourth Normal Form	Definition and concept of multivalued dependencies with example; Concept of fourth normal form	1 hr
Properties of Relational Decomposition	Dependency preservation property and nonadditive (lossless) join property	
Unit 8	**Introduction to Transaction Processing Concepts and Theory**	**Teaching Hours (4)**
Introduction to Transaction Processing	Single-user versus multiuser system; Transactions, Database items, Read and write operations, and DBMS buffers; Why do we need concurrency control? Why do we need recovery?	1 hr
Transaction and System Concepts	Transaction states and operations; The system log; Commit point; Buffer replacement policies	1 hr
Desirable Properties of Transactions	Desirable properties of transactions	
Characterizing Schedules Based on Recoverability	Concept of schedule; Characterizing schedules based on recoverability	2 hrs
Characterizing Schedules Based on Serializability	Conflict serializability; Testing for conflict serializability; View equivalent and view seializability; How serializability is used for concurrency control	
Unit 9	**Concurrency Control Techniques**	**Teaching Hours (4)**
Two-Phase Locking Technique	Concept of two-phase locking; Types of locks and system lock tables; Lock conversion; Guaranteeing serializability by two-phase locking; Basic, conservative, strict, and rigorous two-phase locking; Dealing with deadlock and starvation	2 hrs
Timestamp Ordering	Timestamp ordering concurrency control concept; Basic and strict timestamp ordering; Thomas's Write rule	2 hrs
Multiversion Concurrency Control	Concept of multiversion concurrency control technique; Multiversion technique based on timestamp ordering; Multiversion locking using certify locks	
Validation (Optimistic) Techniques and Snapshot Isolation Concurrency Control	Concept of validation (optimistic) techniques and snapshot isolation concurrency control	

Unit 10	**Database Recovery Techniques**	**Teaching Hours (3)**
Recovery Concepts	Recovery outline and categorization of recovery algorithms; Caching (Buffering) of disk blocks; Write-ahead logging, steal/no-steal, and force/no-force; Checkpoints and fuzzy checkpointing; Transaction rollback and cascading rollback	2 hrs
NO-UNDO/REDO Recovery Based on Deferred Update	Concept of no-undo/redo recovery based on deferred update	
Recovery Technique Based on Immediate Update	Concept of recovery technique based on immediate update	
Shadow Paging	Concept of Shadow Paging	1 hr
Database Backup and Recovery from Catastrophic Failures	Concept of database backup and recovery from catastrophic failures	

Laboratory Works:

The laboratory work includes writing database programs to create and query databases using basic and advanced features of structured query language (SQL) like

- Data definition and data Types
- Specifying constraints (primary key, foreign key, referential integrity etc.)
- Basic and complex retrieval queries
- Aggregate functions
- INSERT, DELETE, and UPDATE Statements
- Using join and views

Text Books:

1. Fundamentals of Database Systems; Seventh Edition; Ramez Elmasri, Shamkant B. Navathe; Pearson Education
2. Database System Concepts; Sixth Edition; Avi Silberschatz, Henry F Korth, S Sudarshan; McGraw-Hill

Reference Books:

1. Database Management Systems; Third Edition; Raghu Ramakrishnan, Johannes Gehrke; McGraw-Hill
2. A First Course in Database Systems; Jaffrey D. Ullman, Jennifer Widom; Third Edition; Pearson Education Limited

Model Question

Course Title: Database Management System
Course No: CSC260
Semester: IV

Full Marks: 60
Pass Marks: 24
Time: 3 Hrs

Section A (Long questions)
Attempt any two questions. (2 × 10 = 20)

1. Consider the following database and write SQL as given:
   Customer (Cno, Cname, Caddress, Ccontact)
   Purchase (Cno,Pid)
   Product (Pid, Pname, price, quantity) (5 × 2 = 10)
   a. Find the names of all products having price 1000.
   b. Find the name of those customers who purchased Dell Laptop.
   c. Find the total number of products purchased by customer 'Ram'
   d. Increase price of all products by 5 %
   e. Find total price of Apple Mobiles
2. What are the benefits of using normalization? Discuss 1NF, 2NF, and 3NF with suitable example. (2.5 + 7.5 = 10)
3. Define Relational Algebra (RA) and explain its six fundamental operations with suitable example. (2 + 8 = 10)

Section B (Short questions)
Attempt any eight questions. (8 × 5 = 40)

4. What database schema? What are functions of database administrator? (2 +3 = 5)
5. Construct an E-R diagram for online course registration where students register courses online.(5)
6. Discuss referential integrity with example. (5)
7. What is functional dependency? Why do we need inference rules? (2 + 3 = 5)
8. Why do we need concurrency control? Discuss two phase locking protocol. (2 + 3 = 5)
9. Why do we need database recovery? Discuss shadow paging technique for database recovery. (2 + 3 = 5)
10. Differentiate concept of Centralized and Client/Server Architectures for DBMSs with suitable example. (5)
11. Define Transaction and explain its desirable properties. (5)
12. Explain constraints and characteristics of Specialization and Generalization of data model. (5)

Course Title: Artificial Intelligence **Full Marks: 60+20+20**
Course No: CSC261 **Pass Marks: 24+8+8**
Nature of the Course: Theory + Lab **Credit Hours: 3**
Year: Second, Semester: Fourth

Course Description: The course introduces the ideas and techniques underlying the principles and design of artificial intelligent systems. The course covers the basics and applications of AI including: design of intelligent agents, problem solving, searching, knowledge representation systems, probabilistic reasoning, neural networks, machine learning and natural language processing.

Course Objectives: The main objective of the course is to introduce concepts of Artificial Intelligence. The general objectives are to learn about computer systems that exhibit intelligent behavior, design intelligent agents, identify AI problems and solve the problems, design knowledge representation and expert systems, design neural networks for solving problems, identify different machine learning paradigms and identify their practical applications.

Detail Syllabus

Chapters / Units	**Teaching Methodology**	**Teaching Hours**
Unit I: Introduction 1.1. Intelligence, Artificial Intelligence (AI), AI Perspectives: acting and thinking humanly, acting and thinking rationally 1.2. History of AI 1.3. Foundations of AI: Philosophy, Economics, Psycology, Sociology, Linguistics, Neuroscience, Mathmatics, Computer Science, Control Theory 1.4. Applications of AI	**Class Lecture**	**3 Hours**
Unit II: Intelligent Agents 2.1. Introduction of agents, Structure of Intelligent agent, Properties of Intelligent Agents 2.2. Configuration of Agents, PEAS description of Agents, PAGE 2.3. Types of Agents: Simple Reflexive, Model Based, Goal Based, Utility Based, Learning Agent 2.4. Environment Types: Deterministic, Stochastic, Static, Dynamic, Observable, Semi-observable, Single Agent, Multi Agent	**Class Lecture** **+** **Lab Session**	**4 Hours**

Unit III: Problem Solving by Searching 3.1. Definition, State space representaion, Problem as a state space search, Problem formulation, Well-defined problems 3.2. Solving Problems by Searching, Search Strategies: Informed, Uninformed, Performance evaluation of search strategies: Time Complexity, Space Complexity, Completeness, Optimality 3.3. Uninformed Search: Depth First Search, Breadth First Search, Depth Limited Search, Iterative Deepening Search, Uniform Cost Search, Bidirectional Search 3.4. Informed Search, Heuristic Function, Admissible Heuristic, Informed Search Techniques: Greedy Best First Search, A* Search, Optimality and Admissibility in A*, Hill Climbing Search, Simulated Annealing Search 3.5. Game Playing, Adversarial Search Techniques: Mini-max Search, Alpha-Beta Pruning 3.6. Constraint Satisfaction Problems, Examples of Constraint Satisfaction Problems	**Class Lecture** **+** **Lab Session**	**9 Hours**
Unit IV: Knowledge Representation 4.1. Definition and importance of Knowledge, Issues in Knowledge Representation, Knowledge Representation Systems, Properties of Knowledge Representation Systems 4.2. Types of Knowledge Representation Systems: Semantic Nets, Frames, Conceptual Dependencies, Scripts, Rule Based Systems (Production System), Propositional Logic, Predicate Logic 4.3. Propositional Logic(PL): Syntax, Semantics, Formal logic-connectives, truth tables, tautology, validity, well-formed-formula, Inference using Resolution, Backward Chaining and Forward Chaining 4.4. Predicate Logic: FOPL, Syntax, Semantics, Quantification, Inference with FOPL: By converting into PL (existential and universal instantiation), Unification and lifting, Inference using resolution	**Class Lecture** **+** **Lab Session**	**14 hours**

4.5. Handling Uncertain Knowledge, Radom Variables, Prior and Posterior Probability, Inference using Full Joint Distribution, Bayes' Rule and its use, Bayesian Networks, Reasoning in Belief Networks 4.6. Fuzzy Logic: Fuzzy Sets, Membership in Fuzzy Set, Fuzzy Rulebase Systems		
Unit V: Machine Learning 5.1. Introduction to Machine Learning , Concepts of Learning, Supervised, Unsupervised and Reinforcement Learning 5.2. Statistical-based Learning: Naive Bayes Model *5.3.* Learning by Genetic Algorithms: Operators in Genetic Algorithm: Selection, Mutation, Crossover, Fitness Function, Genetic Algorithm 5.4. Learning with Neural Networks: Introduction, Biological Neural Networks Vs. Artificial Neural Networks (ANN), Mathematical Model of ANN, Activation Functions: Linear, Step Sigmoid, Types of ANN: Feed-forward, Recurrent, Single Layered, Multi-Layered, Application of Artificial Neural Networks, Learning by Training ANN, Supervised vs. Unsupervised Learning, Hebbian Learning, Perceptron Learning, Back-propagation Learning	**Class Lecture** **+** **Lab Session**	**9 Hours**
Unit VI: Applications of AI (6 Hrs) 6.1. Expert Systems, Components of Expert System: Knwoledge base, inference engine, user interface, working memory, Development of Expert Systems 6.2. Natural Language Processing: Natural Language Understanding and Natural Language Generation, Steps of Natural Language Processing: Lexical Analysis(Segmentation, Morphological Analysis), Syntatic Analysis, Semantic Analysis, Pragmatic Analysis, Machine Translation, 6.3. Machine Vision Concepts: Machine vision and its applications, Components of Machine Vision System	**Class Lecture** **+** **Lab Session**	**6 Hours**

6.4. Robotics: Robot Hardware (Sensors and Effectors) , Robotic Perceptions		

Text Book

1. **Stuart Russel and Peter Norvig**, *Artificial Intelligence A Modern Approach*, Pearson

Reference Books

1. **George F. Luger**, *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*, Benjamin/Cummings Publication
2. **E. Rich, K. Knight, Shivashankar B. Nair**, *Artificial Intelligence*, Tata McGraw Hill.
3. **D. W. Patterson**, *Artificial Intelligence and Expert Systems*, Prentice Hall.
4. **P. H. Winston**, *Artificial Intelligence*, Addison Wesley.

Laboratory Work Manual

Student should write programs and prepare lab sheet for most of the units in the syllabus. Majorly, students should practice design and implementation of intelligent agents and expert systems, searching techniques, knowledge representation systems and machine learning techniques. Students are also advised to implement Neural Networks for solving practical problems of AI. Students are advised to use LISP, PROLOG, and any other high level language like C, C++, Java, etc. The nature of programming can be decided by the instructor and student as per their comfort. The instructors have to prepare lab sheets for individual units covering the concept of the units as per the requirement. The sample lab sessions can be as following descriptions;

Unit II: Intelligent Agents (4 Hrs)

- Write programs for implementing simple intelligent agents.

Unit III: Problem Solving by Searching (12 Hrs)

- Write programs for illustrating the concepts of
    - Uninformed Search like DFS, BFS, etc.
    - Informed Search like Greedy Best First, A*, etc.
    - Game Search like MiniMax Search
- Write programs for constraint satisfaction problems like water jug, n-queen problem, cryptoarithmatic problem, etc.

Unit IV: Knowledge Representation (12 Hrs)

- Write programs for illustrating the concepts knowledge representation systems
    - rule based (program with if then rules)
    - predicate logic (using predicates like in Prolog)
    - frames (using concepts of class)
    - semantic nets (using concepts of graph)

Unit V: Machine Learning (10 Hrs)

- Write program for implementing Naive Bayes.
- Write program for implementing Neural Networks for realization of AND, OR gates.
- Write program for implementing Backpropagation Learning.

Unit VI: Applications of AI (7 Hrs)

- Write program for implementing expert systems like disease prediction, weather forecasting etc.
- Use library tools like NLTK to illustrate concepts of Natural Language Processing.

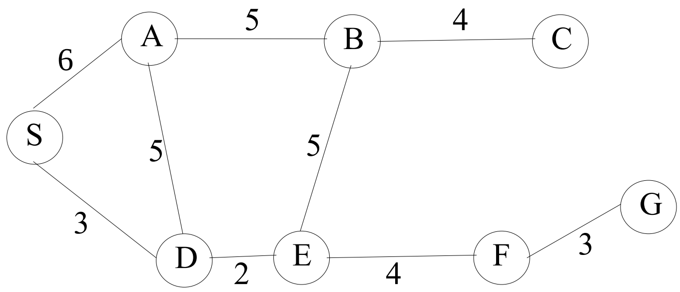
Model Question
Tribhuvan University
Institute of Science and Technology

Course Title: Artificial Intelligence | **Full Marks:** 60
Course No: CSC261 | **Pass Marks:** 24
Level: B. Sc CSIT Second Year/ Fourth Semester | **Time: 3 Hrs**

Section A
Long Answer Questions

Attempt any Two questions. *[2*10=20]*

1. What do you mean by heuristic search? Given following state space representation, show how greedy best first and A* search is used to find the goal state. [2+8] [Unit 3]



   S is the start state and G is the goal state. The heuristics of the states are h(S)= 12 , h(A)= 8, h(D)= 9, h(B)= 7, h(D)= 6, h(E)= 4, h(C)= 5 , h(F)= 2, h(G)= 0.

2. How resolution algorithm is used as a rule of inference in predicate logic? Convert following sentences into FOPL. [4+6] [Unit 4]
   All over smart person's are stupid
   Children's of all stupid persons are naughty
   Roney is Children of Harry
   Harry is over smart
   Prove that "Roney is naughty" using resolution algorithm.

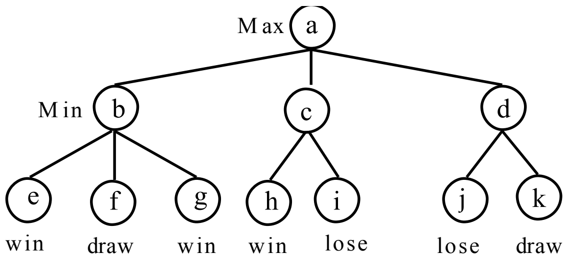
3. What is Artificial Neural Network? Define its mathematical model. Discuss how back propagation algorithm is used to train ANN? [1+2+6] [Unit 5]

Section B
Short Answer Questions

Attempt any Eight questions. *[8*5=40]*

4. Describe how Turing Test is used to define AI as acting humanly? [Unit 1]
5. Differentiate between model based and simple reflex agent with an example. [Unit 2]
6. What is Natural Language Processing? Discuss the steps of natural language processing. [1+4] [Unit 6]
7. How belief networks are constructed? Consider the probability of having cloudy is 50%. The probability that it will rain given the conditions it will be cloudy and if it is winter is 30%. The probability of being winter is 50%. The probability that it will be shiny is 70%. Now construct a belief network for this example. [2+3] [Unit 4]
8. What is expert system? Explain the major components of Expert System? [1+4] [Unit 6]

9. How mini-max algorithm is used in game search. For the following state space, show how mini-max algorithm finds path for the two players. [2.5+2.5][Unit 3]



10. How knowledge is represented using semantic networks? Illustrate with an example. [5] [Unit 4]
11. What is supervised learning? Discuss how Naïve Bayes model works? [Unit 5]
12. Construct PEAS framework for following intelligent agents. [Unit 2]
    a. Internet Shopping Assistant
    b. English Language Tutor