Pokhara University
Faculty of Science and Technology

Course Code: CMP 344 (3 Credits)
Course title: **Computer Networks (3-1-2)**
Nature of the course: Theory & Practical
Level: Bachelor

Full marks: 100
Pass marks: 45
Total periods: 45
Program: BE

1. Course Description

This course is designed to familiarize the student with the basic taxonomy and terminology of the computer. It aims to provide an understanding about the operation of layer-wise network communication, various addressing mechanisms, routing algorithms, network management & security in the computer network and overview of server configuration for complete networking systems.

2. General Objectives

The course is designed with the following objectives:

- ➔ To acquaint the students with the computer networking concepts, including fundamental principles, terminology, and architecture.
- ➔ To make the students familiar with the various network models and protocols at different layers, understanding their roles, functions, and how they enable communication between devices.
- ➔ To expose the students to key concepts in network security, including strategies to protect data integrity, confidentiality, and availability, and to mitigate threats like hacking and data breaches.
- ➔ To equip the students with the practical skills to design, configure, manage, and troubleshoot networks, including the use of networking tools, hardware, and software.

3. Contents in Detail

Specific Objectives	**Contents**

The student will be able to understand the computer networking concepts, including fundamental principles, terminology, and architecture.	**Unit I: Introduction to Computer Network (3 hrs)** 1.1 Definition, merits, Demerits 1.2 Network Models 1.2.1 PAN, LAN, Campus Area Network (CAN), MAN, Country Area Network (CAN*), WAN, GAN 1.2.2 Topological Models (star, bus, distributed bus, mesh, tree, hybrid, ring) 1.2.3 Client/Server, Peer-to-Peer 1.3 ISPs, NSPs Overview and Backbone of Networking 1.4 Recent Trends in Telecom Technologies: 2G/3G/4G/5G.
Understand the layered approach to networking and the various network connecting devices	**Unit II: Reference Model (4 hrs)** 2.1 Protocols and Standards 2.2 Interfaces and Services 2.3 OSI Layers 2.4 TCP/IP Layers 2.5 Comparison between OSI and TCP/IP 2.6 Networking hardware: NIC, Hub, Repeater, Switches, Bridge, Router, Gateway
- Understand how the Physical Layer establishes the foundation for all subsequent layers of the networking model, ensuring that data can be physically transmitted between devices effectively and reliably. - Alongside, students will learn about the various networking parameters	**Unit III: Physical Layer (4 hrs)** 3.1 Guided Media: Copper, Fiber cabling and its capacity standards 3.2 Unguided Media: Bluetooth, Wi-Fi/Wireless LAN, Satellite Communication Basics (Microwaves, Radio waves) 3.3 Circuit/packet/message switching 3.4 ISDN signaling and Architecture 3.5 Network Performance: Bandwidth, Throughput, Latency, Bandwidth-Delay Product, Jitter

● In this chapter the student will learn how the data link layer provides reliable data transfer across a physical network link by handling error detection, frame synchronization, and flow control between directly connected devices.	**Unit IV: Data Link Layer (8 hrs)** 4.1 LLC and MAC sub-layer overview 4.2 Physical (MAC) addressing overview 4.3 Framing 4.4 Flow Control (stop and wait, go-back-N, selective-repeat-request) 4.5 Error Control Mechanism 4.5.1 Error Detection: Parity Check, CRC 4.5.2 Error Correction: Hamming Code 4.6 Channel Access 4.6.1 ALOHA Systems 4.6.2 CSMA, CSMA/CD 4.7 802.3 Ethernet, Fast Ethernet, Gigabit Ethernet 4.8 802.4 Token Bus, 802.5 Token Ring 4.9 Virtual Circuit Switching: Frame Relay, ATM and X.25
● Gain a good understanding of Internet Layer Protocol for ensuring that data packets are correctly routed and delivered across networks, using IP addresses.	**Unit V: Network Layer Protocols and Addressing (8hrs)** 5.1 Logical Addressing 5.1.1 IPV4 addressing, subnetting, supernetting, CIDR, VLSM 5.1.2 IPV6 addressing overview 5.1.3 IPV4 and IPV6 header protocol format 5.1.4 IPV4 and IPV6 feature comparison 5.2 Routing Algorithm Overview 5.2.1 Classful and Classless Routing 5.2.2 Adaptive and non-adaptive Routing 5.2.3 Distance vector and Link-state routing 5.2.4 Interior and exterior routing 5.2.5 Unicast and multicast routing 5.2.6 Routing Algorithms: RIP, OSPF, BGP 5.3 NAT
● Understand the concept of transport layer protocol to ensure reliable and efficient data transfer between devices by managing end-to-end communication.	**Unit VI: Transport Layer and Protocols (4 hrs)** 6.1 Port addressing overview 6.2 Process to process delivery: multiplexing and demultiplexing 6.3 TCP services, features, segment headers, well known ports & Handshaking 6.4 UDP services, features, segment headers, well known ports 6.5 Concept of socket programming: TCP and UDP socket

• In this chapter, the students will learn the traffic shaping algorithms used in computer networks to control the amount and rate of data transmission, helping to manage congestion and ensure QOS.	**Unit VII: Congestion Control and Quality of Services (3 hrs)** 7.1 Congestion Control: Open Loop and Closed Loop 7.2 Traffic Shaping (Leaky bucket and Token bucket) 7.3 TCP Congestion Control
• Learn how the Application Server Protocols facilitates communication between the application server and client devices, ensuring the efficient, secure, and reliable delivery of application services.	**Unit VIII: Application Layer, Servers and Protocols (4 hrs)** 8.1 Domain addressing, DNS server and Queries 8.2 HTTP, FTP & proxy server overview 8.3 DHCP Principles 8.4 Email Server Protocols: SMTP, POP, IMAP
• Here the student will learn how to protect the network and its data from unauthorized access, attacks, and breaches - ensuring confidentiality, integrity, and availability of information.	**Unit IX: Network Management and Security (7 hrs)** 9.1 Introduction to Network Management 9.2 Principles of Cryptography (Symmetric Key: DES, Asymmetric key: RSA) 9.3 Key Exchange Protocols (Diffie-Hellman, Kerberos) 9.4 VPN 9.5 Overview of IP Security 9.6 Firewall, Digital Certificate 9.7 Next Generation Network (NGN)

Note: The figures in the parentheses indicate the approximate periods for the respective units.

4. Methods of Instruction

Lecture, Tutorials, Discussions and Assignments

5. List of Tutorials

The following tutorial activities of 15 hours per group of maximum 24 students should be conducted to cover all the required contents of this course.

S.N.	Tutorials
1	Error Detection and Correction Methods, Parity ,CRC and Hamming code
2	Subnetting
3	Leaky Bucket and Token Bucket/ Queuing Delay Numericals
4	RSA and cryptography Numerical

6. Practical Works

S.N.	Practical works
1	Network commands testing: ping-pong, netstat, nslookup, ipconfig/ifconfig, tracert/traceroute.
2	Setting up Client/Server network system in Microsoft and Linux environment
3	UTP CAT6 cabling: Straight and Cross wiring, testing and verification
4	Internet Packet header analysis using TCPDUMP/WIRESHARK
5	Router Configuration use of packet tracer or other simulator software
6	OSPF configuration and practices
7	VLAN And Router on stick method
8	Web, Proxy, FTP server configuration
9	Implementation of Router ACL, Proxy Firewall, IPTables
10	Case Study: Network Design Standards (eg: building network design with servers including NCR

7. Evaluation system and Students' Responsibilities

Evaluation System

In addition to the formal exam(s) conducted by the Office of the Controller of Examination of Pokhara University, the internal evaluation of a student may consist of class attendance, class participation, quizzes, assignments, presentations, written exams, etc. The tabular presentation of the evaluation system is as follows.

Internal Evaluation	Weight	Marks	External Evaluation	Marks
Theory		**30**	**Semester End Examination**	**50**
Class attendance and participation	10%			
Assignments	20%			
Quizzes/ presentations	10%			
Internal Term Exam	60%			
Practical		**20**		
Attendance and class participation	10%			
Lab Report/Project Work	20%			

Practical Exam/Project Work	40%			
Viva	30%			
Total Internal		**50**		
Full Marks = 50 +50 =100				

Students' Responsibilities:
Each student must secure at least 45% marks in the internal evaluation with 80% attendance in the class to appear in the Semester End Examination. Failing to obtain such a score will be given NOT QUALIFIED (NQ) and the student will not be eligible to appear in the End-Term examinations. Students are advised to attend all the classes and complete all the assignments within the specified time period. Students are required to complete all the requirements defined for the completion of the course

8. Prescribed Books and References

Text Book

1. "Computer Networks", 4th Edition, A. S. Tanenbaum, Pearson Education.
2. "Data Communications and Networking", 5th Edition, Behrouz A. Forouzen, McGraw-Hills.

Reference Books

1. "Data & Computer Communications", 7th Edition, William Stallings, Pearson Education.
2. "Computer Networking: A Top-Down Approach", James F. Kurose, K.W. Rose, 6th Edition, Pearson Education.

Pokhara University
Faculty of Science and Technology

Course No.: CMP 338
Course title: **Simulation and Modeling**
Nature of the course: Theory and Practical

Level: Bachelor

Full marks: 100
Pass marks: 45
Time per period: 1 hour
Total periods: 45
Program: BE

1. Course Description

This course covers the various concepts system simulation. This course emphasizes on fundamental concept, principles and properties of continuous system and discrete system. It covers examples, solutions and programming language regarding continuous and discrete system. It also covers probability concepts and random number generation technique and testing. Output generated from the process is analyzed.

2. General Objective

The main objectives of the course are

- To provide basic knowledge of various systems.
- To study continuous and discrete system.
- To get the concept of probability concept and random numbers.

3. Methods of Instruction

3.1.General instructional Techniques: Lecture, discussion, readings.
3.2.Specific instructional Techniques: Lab works, Project works

4. Contents in Detail

Specific Objectives	Contents
• Familiarize and compare the various concept of system and its environment. • To explain why simulations are used in systems analysis and design, emphasizing their role in modeling complex systems.	**Unit 1 : Introduction to simulation and modeling (4 hrs)** 1.1System and its concept 1.2System Environment 1.3Types of System (continuous and discrete, static and dynamic, stochastic and deterministic) 1.4Steps of Simulation 1.5Advantage, disadvantage and application of simulation 1.6System Modeling and types of models 1.7Principles of Modeling 1.8Verification and validation of model

- To introduce simulation techniques such Monte Carlo simulation. - To provide knowledge on how to create accurate models that represent real-world systems.	**Unit 2 : System Simulation(8 hrs)** 2.1Monte Carlo Method 2.1.1 Problems regarding Monte Carlo method 2.2 Comparison of simulation and analytic solution 2.3 System simulation and its types 2.4 Real time simulation 2.5 Lag Models (Distributed lag Model, Cobweb Model) 2.6 Queuing system and its characteristics and notation 2.7 Single server queuing model 2.7.1 Arrival routine 2.7.2 Departure routine 2.7.3 Performance measure of SSQM 2.8 Time advance mechanism (next event oriented and fixed increment oriented)
- To represent continuous system using differential equations and other mathematical tools. - To analyze system dynamics, understand their stability, and predict future behavior. - To solve and implement continuous system using analog method and programming language.	**Unit 3 : Continuous System(8 hrs)** 3.1 Introduction to continuous system 3.2 Representation of continuous system using differential equation 3.3 Linear and nonlinear differential equations and its examples 3.4 Analog Computer (Components and examples) 3.5 Digital Analog Simulators 3.6 Hybrid Computers 3.7 CSSLs, CSMP III 3.7.1 Structural Statement 3.7.2 Data Statements 3.7.3 Control Statements 3.8 Feedback System with example 3.9 Interactive System
- For understanding and modeling processes where changes occur at distinct, separate points in time or involve discrete states. - To gather statistics white studying discrete system.	**Unit 4 : Discrete System(7 hrs)** 4.1 Introduction to discrete system 4.2 Components of discrete system 4.3 Representation of Time 4.4 Examples for discrete system 4.4.1Telephone call system as lost call and delayed call system 4.4.2 Bank Queue System 4.5 Simulation Programming Task 4.6 Steps of simulation programming task 4.7 Gathering Statistics 4.7.1 Counters and Summary Measures 4.7.2 Measuring Utilization and Occupancy 4.7.2 Recording Distribution and Summary Measures 4.8 Discrete System Simulation Languages

• To understand probability distributions and random variables, which can accurately represent random phenomena in simulations. • To generate random numbers using various generators and test their independence and uniformity property.	**Unit 5 : Probability Concept and Random Numbers(7 hrs)** 5.1 Stochastic System 5.2 Discrete and continuous probability function 5.3 Random numbers versus pseudo random numbers 5.4 Properties of random numbers 5.5 Random number generation Techniques 5.5.1 Linear Congruential Generator 5.5.2 Mixed Generator 5.5.3 Additive and Incremental Generator 5.6 Test for randomness 5.6.1 Uniformity Test - KS Test - Chi Square Test 5.6.2 Independence Test - Run Test (above and below, up and down, length of Runs - Test for Auto correlation - Gap Test - Poker Test
• To test different scenarios, identifying potential issues, and optimizing performance without the risks and costs associated with real-world trials.	**Unit 6 : Discrete System Languages (6 hrs)** 6.1 Simulation using GPSS 6.1.1 GPSS problems 6.2 Simulation using SIMSCRIPT 6.2.1 Organization of SIMSCRIPT 6.2.2 Programs of SIMSCRIPT 6.3 Other discrete simulation Languages
• To interpret, understand, and make decisions based on the results generated by a simulation model. • To validate the simulation results, optimizing system performance, and providing actionable insights.	**Unit 7 : Output Analysis Method(5 hrs)** 7.1 Nature of Problem 7.2 Estimation Method 7.3 Simulation Run Statistics 7.4 Replication of Runs 7.5 Elimination of Initial Bias

5. Laboratory work: (30 hrs)

1. Representing ohm's law and verifying its VI characteristics.
2. Generating value of pi using Monte Carlo method and check its accuracy level
3. Implementing various models in simulation

4. Generating random numbers and their testing
5. Implementing GPSS programs
6. Examples of continuous and discrete system
7. Develop a small project to simulate any mathematical model

6. **List of Tutorials:**
The various tutorial activities that suit this course should cover all the content of this course to give student a space to engage more actively with the course content in the presence of instructor. Students should submit tutorials as assignments or class works to the instructor for evaluation. The following tutorial activities of 15 hours should be conducted to cover all the content of course:

A. Discussion-based Tutorials: (6 hrs)
    1. Explain the concepts of system modeling, abstraction, and the simulation life cycle.
    2. Example of different models that can be simulated.
    3. Analyzing the output obtained from simulation.
    4. Continuous and discrete system example.
B. Problem solving-based Tutorials: (9 hrs)
    1. Examples using Monte Carlo simulation technique.
    2. Example questions for distributed lag model and cobweb model.
    3. Numerical to generate random numbers.
    4. Testing random number properties using various techniques.

7. **Evaluation system and Students' Responsibilities**

Internal Evaluation

In addition to the formal exam(s), the internal evaluation of a student may consist of quizzes, assignments, lab reports, projects, class participation, etc. The tabular presentation of the internal evaluation is as follows.

External Evaluation	Marks	Internal Evaluation	Weight	Marks
Semester-End examination	50	Assignments	12%	
		Attendance	6%	
		Unit test	14%	
		Assessment	28%	
		Practical	40%	
Total External	50	Total Internal	100%	50

Full Marks 50+50 = 100

Student Responsibilities:

Each student must secure at least 45% marks in internal evaluation with 80% attendance in the class in order to appear in the Semester-End Examination. Failing to get such score will be given NOT QUALIFIED (NQ) and the student will not be eligible to appear the Semester-End Examination. Students are advised to attend all the classes and complete all the assignments within the specified time period. If a student does not attend the class(es), it is his/her sole responsibility to cover the topic(s) taught during the period. If a student fails to attend a formal exam, test, etc. there won't be any provision for re-exam.

8. Prescribed Books and References

Text Book

1. G Gorden, **System Simulation**, Prentice Hall of India.
2. Jerry Banks, John S. Carson II, Barry L Nelson, David M. Nicol, **Discrete Event System Simulation,**

Reference Books

1. "Simulation Modeling and Analysis" by Averill M. Law and W. David Kelton
2. "System Simulation with Digital Computer" by N. W. McCormick
3. "Simulation and the Monte Carlo Method" by Reuven Y. Rubinstein and Dirk P. Kroese

Pokhara University
Faculty of Science and Technology

Course No.: CMP 364 (3 Credits)	Full marks: 100
Course title: Machine Learning (3-1-2)	Pass marks: 45
Nature of the course: Theory and Practical	Total Lectures: 45 hrs
Level: Bachelor	Program: BE (Computer)

1. Course Description

This course is designed to provide the fundamental principles and methodologies of machine learning. Students will learn to develop algorithms that can automatically learn from data, improve with experience, and make predictions or decisions. The course covers supervised, unsupervised machine learning alongside in-depth concepts of neural networks, and model evaluation and validation with a focus on both theoretical understanding and practical implementation.

2. General Objectives

- To provide the students with key concepts and principles of machine learning.
- To acquaint the students with the skills to develop and implement different machine learning algorithms.
- To develop the skills in students to use popular machine learning tools and frameworks and apply machine learning techniques to solve real-world problems.
- To acquaint the students with the knowledge of advanced topics in neural networks, including Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs).
- To provide the students with the knowledge to evaluate and interpret the performance of machine learning models.

3. Methods of Instruction

Lecture, Discussion, Readings, Practical works and Project works.

4. Contents in Detail

Specific Objectives	Contents

• Describe the machine learning process in detail.	**1. Introduction to Machine Learning (5 hrs)** **1.1.** Definition and Evolution of Machine Learning **1.2.** Types of Machine Learning **1.2.1.** Supervised Learning **1.2.2.** Unsupervised Learning **1.2.3.** Reinforcement Learning **1.2.4.** Active Learning **1.3.** Machine Learning Workflow **1.3.1.** Problem Definition **1.3.2.** Data Collection and Preprocessing **1.3.3.** Model Selection **1.3.4.** Model Evaluation and Validation **1.3.5.** Model Deployment **1.4.** Challenges in Machine Learning **1.4.1.** Data Quality Issues **1.4.2.** Computational Complexity **1.4.3.** Interpretability and Explainability **1.4.4.** Ethical Considerations
● Design and implement supervised learning algorithms to solve real world problems.	**2. Supervised Learning (10 hrs)** **2.1.** Types of Supervised Learning **2.1.1.** Regression **2.1.2.** Classification **2.2.** Regression **2.2.1.** Linear Regression **2.2.1.1.** Simple and multiple regression **2.2.1.2.** Polynomial Regression **2.2.2.** Regularization Techniques **2.2.2.1.** Ridge regression **2.2.2.2.** Lasso regression **2.2.2.3.** Bias-variance tradeoff **2.2.3.** Support Vector Regression **2.3.** Classification **2.3.1.** Logistic Regression **2.3.1.1.** Binary classification **2.3.1.2.** Multi-class classification **2.3.2.** K-Nearest Neighbors (KNN) **2.3.3.** Support Vector Machine (SVM) **2.3.3.1.** Hyperplane and Support Vectors **2.3.3.2.** Kernels and its Types: Linear, Polynomial, Radial Basis Function (RBF) **2.3.3.3.** SVM for Linear and Non-linear Classification **2.3.4.** Decision Trees **2.3.4.1.** Construction and pruning of decision trees **2.3.4.2.** Ensemble methods: Bagging, Random Forests

• Design and implement unsupervised learning algorithms to solve real world problems.	**3. Unsupervised Learning (10 hrs)** **3.1.** Basic Concept of Unsupervised Learning **3.2.** Types of Unsupervised Learning **3.2.1.** Clustering **3.2.2.** Dimensionality Reduction **3.2.3.** Association Rule Learning **3.3.** Clustering **3.3.1.** K-Means Clustering **3.3.2.** Hierarchical Clustering **3.3.2.1.** Agglomerative Clustering **3.3.2.2.** Divisive Clustering **3.3.3.** Density-based Clustering **3.3.3.1.** DBSCAN **3.4.** Dimensionality Reduction **3.4.1.** Principal Component Analysis (PCA) **3.4.2.** Linear Discriminant Analysis (LDA)
• Design and implement Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs).	4. **Artificial Neural Network (12 hrs)** 4.1. Introduction to Neural Network 4.1.1. Neural Network Architectures 4.1.1.1. Feedforward 4.1.1.2. Convolution 4.1.1.3. Recurrent 4.1.2. Perceptrons 4.1.2.1. Single layer perceptron 4.1.2.2. Multilayer Perceptron 4.1.2.3. Backpropagation 4.2. Training Neural Network 4.2.1. Forward and Backward Propagation 4.2.1.1. Forward Propagation 4.2.1.2. Backpropagation and Gradient Descent 4.2.2. Loss functions 4.2.2.1. Role of loss function 4.2.2.2. Mean Squared Error (MSE) 4.2.2.3. Cross-Entropy Loss 4.2.3. Regularization Techniques 4.2.3.1. Overfitting and underfitting 4.2.3.2. Regularization methods: L1, L2, Dropout, Batch Normalization 4.3. Advanced Neural Network Architecture 4.3.1. Convolution Neural Network (CNNs) 4.3.1.1. CNNs and their components 4.3.1.2. Convolution, Pooling and fully connected layers 4.3.1.3. Applications in image processing and computer vision 4.3.2. Recurrent Neural Networks (RNNs) 4.3.2.1. Basics of RNNs 4.3.2.2. Long Short-Term Memory (LSTM)

	4.3.2.3. Gradient Recurrent Units (GRU) 4.3.2.4. Applications in time-series prediction
● Apply the various techniques to evaluate and validate machine learning algorithms.	5.**Model Evaluation and Validation (8 hrs)** 5.1. Need of Model Evaluation in ML 5.2. Model Evaluation Metrics 5.2.1. Classification Metrics 5.2.1.1. Accuracy 5.2.1.2. Precision, Recall and F_β score 5.2.1.3. Confusion Matrix 5.2.1.4. ROC and PR-Curve 5.2.2. Regression Metrics 5.2.2.1. Mean Absolute Error (MAE) 5.2.2.2. Mean-Squared Error (MSE) 5.2.2.3. Root Mean-Squared Error (RMSE) 5.2.2.4. R-Squared 5.3. Model Validation Techniques 5.3.1. Train-Test Split 5.3.2. Cross-Validation 5.3.2.1. K-Fold Cross Validation 5.4. Hyperparameter Tuning 5.4.1. Grid Search 5.4.2. Random Search

5. Practical Works

Laboratory work of 30 hours per group of a maximum of 24 students must cover the following lab works:

SN	Implementation Description
1	Implement and evaluate a support vector machine.
2	Implement linear regression on a dataset (e.g., housing prices) and evaluate its performance. Apply ridge and lasso regression to prevent overfitting and compare results.
3	Implement k-means clustering and visualize the clusters on a dataset (e.g., customer segmentation) and apply PCA to reduce dimensionality and visualize data
4	Implement k-fold cross-validation on a classification or regression model
5	Build and train CNNs for image classification and RNNs for sequence prediction.

Students must submit a project work that uses all the knowledge obtained from this course to solve any problem they choose. The marks for the practical evaluation must be based on the project work submitted by students.

6. List of Tutorials

The various tutorial activities that suit your course should cover all the content of the course to give students a space to engage more actively with the course content in the presence of the instructor. Students should submit tutorials as assignments or class works to the instructor for evaluation. The following tutorial activities of 15 hours per group of maximum 24 students should be conducted to cover the content of this course:

A. Discussion-based Tutorials: (3 hrs)
   a. Evolution of Machine Learning (Class discussion).
   b. Group debate on the challenges in Machine Learning. (Oral Presentation).

B. Problem solving-based Tutorials: (6 hrs)
   a. Design CNNs for image classification.
   b. Design RNNs for sequence prediction.

C. Review and Question/Answer-based Tutorials: (6 hrs)
   a. A detailed case study on recent Tools and Frameworks for example TensorFlow, PyTorch and Python (Oral Presentation in class).
   b. Case study on model evaluation and validation.
   c. Students ask questions within the course content, assignments and review key course content in preparation for tests or exams.

7. Evaluation System and Students' Responsibilities

Evaluation System

The internal evaluation of a student may consist of assignments, attendance, internal assessment, lab reports, project works etc. The internal evaluation scheme for this course is as follows:

Internal Evaluation	Weight	Marks	External Evaluation	Marks
Theory		30		
Attendance & Class Participation	10%			
Assignments	20%			
Presentations/Quizzes	10%			
Internal Assessment	60%		Semester-End examination	50
Practical		20		
Attendance & Class Participation	10%			
Lab Report/Project Report	20%			
Practical Exam/Project Work	40%			
Viva	30%			
Total Internal		50		
Full Marks: 50 + 50 = 100				

Student Responsibilities

Each student must secure at least 45% marks separately in internal assessment and practical evaluation with 80% attendance in the class in order to appear in the Semester End Examination. Failing to get such a score will be given NOT QUALIFIED (NQ) to appear for the Semester-End Examinations. Students are advised to attend all the classes, formal exam, test, etc. and complete all the assignments within the specified time period. Students are required to complete all the requirements defined for the completion of the course.

8. Prescribed Books and References

Text Books

1. Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.
2. Murphy, K. P. (2012). *Machine learning: a probabilistic perspective.* MIT press.

References

1. Géron, A. (2022). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow.* O'Reilly Media, Inc.

2. Ian, G. (2016). *Deep learning/Ian Goodfellow, Yoshua Bengio and Aaron Courville*.MIT press.

Pokhara University
Faculty of Science and Technology

Course Code.: CMP 441
Course title:**Image Processing and Pattern Recognition (3-1-2)**
Nature of the course:Theory & Practice
Year, Semester:……………
Level: Bachelor

Full marks: 100
Pass marks: 45
Time per period: 1 hour
Total periods: 45
Program: BE

1. Course Description

This course covers essential image processing and pattern recognition techniques, including image enhancement, segmentation, and feature extraction. Students will use these methods to tackle real-world problems in fields such as medical imaging and computer vision, blending theory with practical, hands-on projects.

2. General Objectives

- To familiarize students with key techniques in image processing and pattern recognition.
- To equip students with skills for practical application in real-world scenarios.
- To develop competence in solving complex problems using advanced image processing methods.

3. Contents in Detail

Specific Objectives	Contents
Understand the foundational concepts, applications, and basic techniques involved in digital image processing.	**Unit 1: Introduction to Digital Image Processing [4 hrs]** 1.1 Fundamental Steps and Elements of DIP 1.2 Applications Areas of DIP 1.3 Elements of Visual Perception, Sampling and Quantization 1.4 Image and its types (Color Image, grayscale Image) 1.5 Relationship between Pixels (Neighbors, Path, Connectivity, Adjacency, Distances).
Learn practical methods to enhance images using spatial domain techniques like gray-level transformations and spatial filters.	**Unit2: Image Enhancement in Spatial Domain [7 hrs]** 2.1. Gray Level Transformations 2.1.1. Point operations 2.1.2. Contrast stretching, 2.1.3. Thresholding, 2.1.4. Digital negative, 2.1.5. Intensity level slicing 2.1.6. Bit Plane Slicing 2.2.Histogram Modeling, Histogram equalization, Histogram

	matching 2.3. Enhancement Using Arithmetic and Logic Operations 2.4. Spatial Filters 2.5. Smoothening and Sharpening Spatial Filters 2.5.1. Averaging 2.5.2. Median filtering 2.5.3. Spatial Low Pass 2.5.4. High pass filtering 2.5.5. Magnification by replication and interpolation
Master frequency domain techniques for image enhancement using Fourier Transform and frequency filters.	**Unit 3: Image Enhancement in the Frequency Domain [6 hrs]** 3.1. Introduction to Fourier Transform, DFT, FFT 3.2. Computing and Visualizing the 2D DFT 3.3. Smoothing Frequency Domain Filters 3.4. Sharpening Frequency Domain Filters, 3.5. Other Image Transforms 3.5.1. Hadamard transform 3.5.2. Haar transform 3.5.3. Discrete Cosine transform
Develop skills in restoring degraded images by applying noise reduction techniques in spatial and frequency domains.	**Unit 4: Image Restoration [4 hrs]** 4.1. A model of The Image Degradation / Restoration Process, 4.2. Noise Models Restoration in the Presence of Noise-Only Spatial Filtering 4.3 Types of noise (White noise, salt & pepper noise, Impulse noise, Gaussian noise, Rayleigh noise) 4.4. Periodic Noise Reduction by Frequency Domain Filtering
Acquire knowledge of image compression techniques to efficiently reduce file sizes while maintaining quality.	**Unit 5:Image Compression and Coding [7 hours]** 5.1 Need of Compression 5.2 Lossy & Lossless Compression, Issues of Compression 5.3 A generic model of compression 5.4 Element of Information Theory (Self Information, Entropy) 5.5 Data Redundancy, Coding Redundancy 5.6 Types of compression techniques 5.6.1 Entropy Encoding 5.6.1.1 Run Length Encoding (Interpixel Redundancy) 5.6.1.2 Huffman Encoding (Coding Redundancy) 5.6.1.3 LZW coding 5.6.2 Transform Coding 5.6.2.3 Predictive Coding
Learn to segment images into meaningful regions using various edge detection and thresholding techniques. Understand object	**Unit 6: Image Analysis [9 hrs]** 6.1 Introduction to Image Analysis 6.2 Feature Extraction & Types of Features, Detection of Discontinuities,

representation using descriptors for effective shape analysis in images.	6.3 Segmentation: Discontinuities-based segmentation (Point detection, line detection, Edge detection) 6.4 Similarities-based segmentation 6.4 .1 Feature Thresholding 6.4.1.1 Amplitude Thresholding 6.4.1.2.Thresholding based upon histogram statistics 6.4.1.3. Multi-level Thresholding 6.4.1.4. Local & Global Thresholding 6.4.1.5. Optimum Thresholding 6.4.2 Region growing based segmentation: seeded and unseeded 6.4.3 Region splitting& Merging 6.5 Region Description & representation 6.5.1Crack code & chain code 6.5.2 Polygon Approximation 6.5.3 Signatures 6.5.4 Shape Numbers 6.5.5 Fourier Descriptors
Gain expertise in object recognition and classification using pattern recognition techniques. Master the fundamentals of pattern recognition, focusing on feature extraction and classification algorithms.	**Unit 7: Pattern Recognition & Artificial Neural Network in Pattern Recognition [8 hours]** 7.1 Image pattern and its recognition 7.2 General steps of Pattern recognition 7.3 Boundary Preprocessing, Boundary Feature Descriptors, Region Feature Descriptors 7.4 Feature extraction: PCA 7.5 Scale-Invariant Feature Transform (SIFT) 7.6 Patterns and Pattern Classes 7.7 Pattern Classification by Prototype Matching 7.8 Optimum (Bayes) Statistical Classifiers 7.9 Artificial Neural Network 7.9.1 Perceptron 7.9.2 Hopfield Network

Note: The figures in the parentheses indicate the approximate periods for the respective units.

4. Methods of Instruction

General instructional Techniques: Lectures, discussion, Projects, tutorials, lab, assignments, quizzes.

5. List of Tutorials

The following tutorial activities of 15 hours per group of maximum 24 students should be conducted to cover all the required contents of this course.The various tutorial activities that suit your course should cover all the content of the course to give students a space to engage more actively with the course content in the presence of the instructor/professor. The tutorials section will cover the following portion mentioned below:

S.N.	Tutorials
1	Students will explore the origins and applications of digital image processing, including key steps like image sampling, quantization, and system components. They will also study pixel relationships (connectivity, distance measures) and the influence of visual perception on processing techniques.
2	Practical techniques like gray-level transformations (point operations, contrast stretching, thresholding), histogram processing, and spatial filtering (smoothing, sharpening). Students learn to combine these methods for optimal image enhancement.
3	Introduced to frequency domain enhancement through Fourier Transform and 2D DFT. Tutorials focus on applying frequency filters, exploring transforms like Hadamard and DCT, and using FFT for efficient processing.
4	Restoring images by understanding degradation models and noise reduction techniques. Students learn spatial filtering and frequency domain methods to restore image quality from various types of noise.
5	Explore image compression, focusing on reducing file size via coding techniques (Huffman, run-length), and understanding lossless/lossy methods like predictive coding. Tutorials emphasize practical applications in compression.
6	Introduce students to binary image processing with operations like dilation, erosion, and logical operations. Students learn to apply these techniques for noise removal and shape analysis through practical exercises.
7	Segmentation techniques including edge detection, thresholding (global, adaptive), and region-based methods. Tutorials focus on practical exercises to segment images into meaningful regions.
8	Methods for representing objects using descriptors like chain codes and Fourier descriptors. Students gain practical experience in analyzing and describing object shapes within images.
9	Pattern recognition and classification through tutorials on decision-theoretic methods and an introduction to neural networks. Practical exercises focus on object identification and classification.
10	Feature extraction and classification techniques, allowing students to apply various algorithms for pattern recognition and object classification in image datasets.

6. Practical Works

1. Every topic of the course content should be included for the lab.
2. Individual or group project work to develop a web application could be assigned. This should cover most of the technologies included in the course content.

7. Evaluation system and Students' Responsibilities

Evaluation System

In addition to the formal exam(s) conducted by the Office of the Controller of Examination of Pokhara University, the internal evaluation of a student may consist of class attendance, class participation, quizzes, assignments, presentations, written exams, etc. The tabular presentation of the evaluation system is as follows.

External Evaluation	Marks	Internal Evaluation	Weight	Marks
Semester-End examination	50	Theory		30
		Assignments	15%	
		Attendance/Class Participation	15%	
		Project/Presentation	20%	
		Term exam	50%	
		Practical		20
		Lab Report/Project Report	20%	
		Attendance	20%	
		Practical Exam/Project work	40%	
		Viva	20%	
	50	Internal Final	100%	50
Full Marks 50+50 = 100				

Students' Responsibilities:
Each student must secure at least 45% marks in the internal evaluation with 80% attendance in the class to appear in the semester-end examination. Failing to obtain such score will be given NOT QUALIFIED (NQ) and the student will not be eligible to appear in the End-Term examinations. Students are advised to attend all the classes and complete all the assignments within the specified time period. If a student does not attend the class(es), it is his/her sole responsibility to cover the topic(s) taught during the period. If a student fails to attend a formal exam, quiz, test, etc. there won't be any provision for a re-exam.

8. Prescribed Books and References

Text Book:

Rafael C. Gonzalez and Richard E. Woods, "Digital Image Processing", Prentice Hall of India Pvt. Ltd., 2010.

References:

1. I. Pitas, "Digital Image Processing Algorithms", Prentice Hall, 2009.
2. A. K. Jain, “Fundamental of Digital Image processing”, Prentice Hall of India Pvt. Ltd., 2011.
3. K. Castlemann, “Digital image processing”, Prentice Hall of India Pvt. Ltd., 2010.
4. R. C. Gonzalez and P. Wintz, “Digital Image Processing”, Addison-Wesley Publishing, 2009.
5. P. Monique and M. Dekker, “Fundamentals of Pattern recognition”, 2007.
6. M. James, “Pattern recognition”, BSP professional books, 2008.

Pokhara University
Faculty of Science and Technology

Course Code.: CMP 422 (2 Credits)
Course title: **Compiler Design (2-2-2)**
Nature of the course: Theory & Practice
Program: BE
Level: Bachelor

Full marks: 100
Pass marks: 45
Time per period: 1 hour
Total periods: 30

1. Course Description

The primary objective of this course is to introduce the major concepts of language translation and compiler design and impart the knowledge of practical skills necessary for constructing a compiler. The course includes phases of compiler, parsing, syntax-directed translation, type-checking use of symbol tables, code optimization techniques, intermediate code generation, code generation, and data flow analysis.

2. General Objectives

- Introduce students to the need for a compiler and various phases of a compiler.
- To familiarize students with the concept of designing a compiler for given a set of language features.
- Enhance the knowledge of patterns, tokens & regular expressions for lexical analysis.
- To help the students acquire skills in using lex tool & yacc tool for developing a scanner and parser.
- To enable the students to design and implement LL and LR parsers.
- To make the students able to design algorithms to do code optimization to improve the performance of a program in terms of space and time complexity.
- To familiarize the students with the concept of designing algorithms to generate machine code.

3. Contents in Detail

Specific Objectives	**Contents**

The structure of a compiler, the science of building a compiler, programming language basics.	**Unit I: Overview of Compilation (4 Hrs)**
	1.1 Compiler and Interpreter
	1.2 Compiler Structure
	1.3 Overview of Translation
	    1.3.1 The Front End
	    1.3.2 The Optimizer
	    1.3.3 The Backend

The Role of the Lexical Analyzer, Input Buffering, Recognition of Tokens, The Lexical-Analyzer Generator Lex, Finite Automata, From Regular Expressions to Automata, Design of a Lexical-Analyzer	**Unit II: Scanners (7 Hrs)**
	2.1 The Role of the Scanner
	2.2 Recognizing Words
	    2.2.1 A Formalism of Recognizer
	    2.2.2 Recognizing Complex Words
	2.3 Regular Expression
	    2.3.1 Formalizing the Notation

Generator, Optimization of DFA-Based Pattern Matchers.	2.3.2 Closure properties of REs 2.4 From Regular Expression to Scanner 2.4.1 Nondeterministic Finite Automata 2.4.2 Regular Expression to NFA: Thomson's Construction 2.4.3 NFA to DFA 2.4.4 Using a DFA as a Recognizer
Introduction, Context-Free Grammars, Writing a Grammar. Learning Regular Expressions. Top-Down Parsing, Bottom-Up Parsing.	**Unit III: Parsers (6 Hrs)** 3.1 Introduction to Syntax Analyzer 3.2 Context Free Grammar 3.3 Top Down Parsing 3.3.1 Transforming a Grammar for Top Down Parsing 3.3.2 Recursive Descent Parsing 3.3.3 Table Driver LL(1) Parsers 3.4 Bottom Up Parsing 3.4.1 The LR(1) Parsing Algorithm 3.4.2 Building LR(1) Table
Intermediate-Code Generation: Variants of Syntax Trees	**Unit IV: Intermediate Representations (5 Hrs)** 4.1 A Taxanomy of Intermediate Representations 4.2 Graphical IRs 4.2.1 Syntax Related Trees 4.2.2 Graphs 4.3 Linear IRs 4.4 Mapping Values to Names 4.5 Symbol Tables 4.5.1 Hash Tables 4.5.2 Building a Symbol Table
Understanding Code Generation Techniques	**Unit V: Code Generation (4 Hrs)** 5.1 Issues in Design of Code Generation 5.2 Basic Blocks and Flow Graphs & Optimization of Basic Blocks 5.3 A Simple Code Generator & Peephole Optimization 5.4 Register Allocation & Assignment, Dynamic Programming Code Generation
Optimizing the generated code.	**Unit 6: Machine Independent Optimization (4 Hrs)** 6.1 The Principal Sources of Optimization 6.2 Loops in Flow Graphs, Constant Propagation, Partial Redundancy Elimination 6.3 Introduction to Data-Flow Analysis 6.4 Solving Data-Flow Equations

5. List of Tutorials

The following tutorial activities of 15 hours per group of maximum 24 students should be conducted to cover all the required contents of this course.

S.N.	Tutorials
1	Describing various phases and structures of a compiler.
2	Understand Regular Expressions, NFA, DFA, Thomson's Construction
3	Course works on Bottom Up and Top Down Parsing. Building LR(1) and LL(1) tables.
4	Creating graphical and linear intermediate code generations. Building Hash tables.
5	Basic exercise on designing code generator and Peephole optimization
6	Designing Flow Graphs, Data Flow Analysis. Solving Data-Flow Equation

6. Practical Works

S.N.	Practical works
1	Programming Language Basics.
2	Using Lex Tools
3	Optimization of DFA Based Pattern Matcher
4	Writing Grammar and Implementing the Grammar
5	Desiging a Basic Recursive Descent Parser, Computing FIRST & FOLLOW Values, Predictive Parsing Table Construction & LL(1) Grammar
6	Simple LR Parser, Creating Parsing Table, Parser Generator
7	Control Flow, Switch-Statements, Intermediate Code for Procedures

7. Evaluation system and Students' Responsibilities

Evaluation System

In addition to the formal exam(s) conducted by the Office of the Controller of Examination of Pokhara University, the internal evaluation of a student may consist of class attendance, class participation, quizzes, assignments, presentations, written exams, etc. The tabular presentation of the evaluation system is as follows.

External Evaluation	Marks	Internal Evaluation	Marks
Semester-End Examination	50	Class attendance and participation	5
		Practical	20
		Quizzes/assignments and presentations	5
		Internal Term Exam	20
Total External	50	Total Internal	50
Full Marks 50+50 = 100			

Students' Responsibilities:

Each student must secure at least 45% marks in the internal evaluation with 80% attendance in the class to appear in the Semester End Examination. Failing to obtain such score will be given NOT QUALIFIED (NQ) and the student will not be eligible to appear in the End-Term examinations. Students are advised to attend all the classes and complete all the assignments within the specified time period. If a student does not attend the class(es), it is his/her sole responsibility to cover the topic(s) taught during the period. If a student fails to attend a formal exam, quiz, test, etc. there won't be any provision for a re-exam.

8. Prescribed Books and References

Text Book

Keith D. Cooper and Linda Torczon, *Engineering a Compiler*, 2nd Edition, Morgan Kaufman

Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffry D. Ullman, *Compilers: Principles, Techniques, and Tools*, 2nd Edition, Addison Wesley

Reference Books

Douglas Thain, *Introduction to Compilers and Language Design: Second Edition*, ISBN-13 : 979-8655180260

Pokhara University
Faculty of Science and Technology

Course Code.: **PRJ 360** (2 Credits)	Full marks: **100**
Course title: **Project I**	Pass marks: **45**
Nature of the course: **Practical** (0-0-2)	Time per period: **1 hour**
Year, Semester: **Year 3, Semester 6**	Total periods: **30**
Level: **Bachelor**	Program: **BECE, BEIT, BESE**

1. Course Description

This course is project work that is about involving in a team to design and produce tangible computer hardware and/or software and/or embedded product which can be executable in order to solve a real-world problem. In the due course, students are required to apply. theoretical knowledge obtained so far, and they are equally encouraged to learn and apply the tools and techniques prevailing in the industry at the time. As this is teamwork, students also learn and exhibit team building exercises.

This project work is recommended, but not compulsory, to be carried out in association with Project II. A larger framework (incorporating both the project works) may be conceptualized, and the first part may be done in this subject so that it can be extended to Project II.

2. General Objectives

The general objectives of the course are: -

- To provide practical knowledge of project undertaking by focusing on planning, requirements elicitation, design, development and implementation of a project.
- To provide the knowledge of tools and techniques currently used in the industry while developing a project.
- To make students able to work in a team, which also includes team building exercises.
- To help students develop necessary skills required to prepare project reports and that needed for oral presentation of their projects.

3. Working Procedure

The project course requires students to get themselves involved in a group consisting of generally 3-4 members and work jointly in the team, on a proposed task under the direct supervision of the faculty members assigned by their respective departments. The project may be selected by the department or project committee in consultation with the industries, and they shall be software and or electronic hardware based. The project may be done using any programming language or platform and it may be any type of application e.g. Scientific Applications, Information Systems, Web Applications, Games, Simulations etc. but it must find its practical usage in daily life, and it should be relevant, as possible, to the local industry environment and its demands.

4. Project Working Phases

The project must be started at the beginning of the semester, span throughout the semester and finished by the end of that very semester. The project work will be continuously assessed by a panel of examiners appointed by the college. Additionally, oral examination / viva-voce will be conducted by internal and external examiners appointed by the college.

The entire process consists of three phases – (1) Proposal, (2) Mid-term and (3) Final. The proposal phase shall occur in the beginning of the semester; the mid-term defense shall be organized in the middle of semester (at least 4 weeks after the Proposal Defense); and the final presentation shall be held at the end of the semester (at least 4 weeks after the Mid-term Defense). The marks distribution for the phases are 30%, 30% and 40% simultaneously.

4.1 Proposal Phase

The students are required to form a team and come up with a conceptual and implementational framework for their project work which must be documented in the form of a proposal report and presented in front of a panel of examiners in a formal presentation organized by the department or the project committee.

Supervisor must be assigned after the acceptance of the proposal. Supervisor may also be assigned in the very beginning or after finalizing the title with the approval of the department or the project committee.

Evaluation Criteria:

30% of the marks shall be based on the following criteria:

Task accomplished • Feasibility study- nature of the project, title, abstract etc. • Objective • Requirements analysis and specification • Project plan - cost estimation, timeline • Creativity, innovation • Teamwork	20%
Documentation • Report format and layout (refer to the Project Guideline)	10%

4.2 Mid-term Phase

Students are required to present the progress of the project work, and the amount of progress should in general be 60% or more. Students must have finished the design phase including the overall system/architectural design and validation scheme. The project must also be in the implementational phase, and the preliminary results must have been seen during this phase of project progress.

A mid-term defense shall be organized by the department or the project committee, where a panel of examiners will evaluate the project. Students must have obtained written consent of their supervisor for appearing in the mid-term defense.

Evaluation Criteria:
30% of total mark shall be based on the following criteria:

Task accomplished • Level of proposal-feedback incorporated • System/architectural design • Progress/depth of project work • Validation criteria • Group/team effort	20%
Documentation • Report organization • Completeness and consistency of the report • Organization and analysis of data and results	10%

4.2 Final Phase
All students must have finished all phases of their project work including requirements analysis, design, coding, testing by the time of the final project presentation. Students must come up with a visible output of the product that they have developed, and they should demonstrate them during the oral defense. A panel of examiners (comprised of an expert from industry) shall examine the project work.

Students must have obtained written consent of their supervisor in order to appear in the final defense.

Evaluation:
40% of total mark shall be based on the following criteria:

Task accomplished • Performance during presentation • Contribution in the entire work • Completeness of the work, • Analysis and design, tools and techniques used Viva-voce Project demonstration	30%
Documentation • Final project report – layout and format (see the Project Guidelines)	10%

5. Reference / Project Guideline

Students must follow the Project Guidelines provided by the University / College.