Draft Copy

Pokhara University
Faculty of Science and Technology

Course Code: MTH 150	Full Marks: 100
Course title: Algebra and Geometry (3-2-0)	Pass Marks: 45
Nature of the course: Theory	Total Lectures: 45 hours
Level: Bachelor	Program: BE

1. Course Description

The course covers linear algebra, vector algebra, linear programing, two-dimensional and three-dimensional analytical geometry.

2. General Objectives

To provide the sound knowledge of Algebra and Geometry of two and three dimensions.

3. Methods of Instruction

Lecture, Discussion, and Class Work

4. Contents in Detail

Specific Objectives	Contents
• Solve system of linear equations and find rank of a matrix	**Unit I: Matrix and System of linear equations (4 hrs.)** 1.1 System of linear equations, classification, matrix and vector representations, solution by Cramer's and Gauss elimination methods. 1.2 Rank of a matrix, elementary transformation on matrices, equivalent matrices and their ranks, computation of rank of matrix by reducing into echelon (triangular) form and canonical (normal) form, rank of coefficient matrix and consistency of system of equations.
• Solve the problems related to vector space and determine eigenvalues and eigenvectors.	**Unit II: Vector Space (7 hrs.)** 2.1 Vectors and vector spaces, $\mathbb{R}^2 and\ \mathbb{R}^3$ as vector spaces and their subspaces, Basis. 2.2 Linear dependence and independence, rank of matrix in terms of linearly independent column (row)vectors. 2.3 Linear transformation and transformation by matrix multiplication as linear transformation, orthogonal matrix and transformation by orthogonal matrix. 2.4 Eigenvalues and eigenvectors, characteristic equation and computation of Eigenvalues and eigenvectors, properties of eigenvalues. 2.5 Cayley-Hamilton theorem (statement and verification only), application to compute inverse of a matrix (up to 3×3 matrix). 2.6 Similar matrices and diagonalization of matrix (up to 2×2 matrices only).

• Solve linear programing problems by Simple method.	**Unit III: Linear Programing Problems (5 hrs.)** 3.1 Introduction, Model Formulation, Standard Form, Solution by Simplex Method 3.2 Duality, Dual Simplex Method 3.3 Simplex method for mixed inequalities (Big-M method)
• Solve the problems related to product of three and four vectors and	**Unit IV: Vector Algebra** (**6 hrs.)** 4.1 Review of vectors in terms of coordinates, scalar and vector product of two vectors. 4.2 Vector and scalar product of three and four vectors with physical interpretations. 4.3 Reciprocal system of vectors, properties of reciprocal system of vectors and related problems.
• Test the convergence and divergence of the series	**Unit V: Infinite Series (5 hrs.)** 5.1 Infinite sequence and concept of convergence, infinite series, convergence of series, invariance of convergence (by addition or removal of certain finite numbers or terms, by multiplication by any finite constant, sum and product of convergent series), necessary condition for an infinite series to be convergent (divergent test), convergence of infinite geometric series. 5.2 Series with positive terms (or all terms negative) and different convergence tests (theorems statement with illustrations, proof not required). The hyper-harmonic series (p-series) and its convergence, Comparison test, ratio test, root test. 5.3 Alternating series (Series with negative and positive terms alternatively) Leibnitz test, absolute convergence, power series, interval of convergence and radius of convergence
• Solve the problems related coordinate transformation, ellipse, hyperbola, and conic section	**Unit VI: Two-dimensional Geometry (6 hrs.)** 6.1 Transformation of coordinates: by transformation of origin, by transformation of axes by changing direction of axes, combined transformation and use of transformation to reduce complex equation into standard equations. 6.2 Ellipse: Standard equation of ellipse with derivation, equation of tangent and normal condition for tangency. 6.3 Hyperbola: Standard equation of a hyperbola with derivation, rectangular hyperbola, conjugate hyperbola, equation of tangent and normal condition for tangency. 6.4 General equation of Conic section in Cartesian and Polar form
• Solve the problems related straight lines sphere, cone, and cylinder	**Unit VII: Three-dimensional Geometry (12 hrs.)** 7.1 Review coordinate in space and plane. 7.2 Straight line: Introduction, line in symmetrical form, line passing through two given points, reduction of general equation of a line into symmetrical form, angle between a plane and a line, conditions for a line to lie on a plane, length of a perpendicular

	from a given point to the line, coplanar lines, condition for coplanarity of two lines, shortest distance its length and equation. 7.3 Sphere: Equation of a sphere, condition for a general equation of second degree to represent a sphere, equation in diameter form, plane section of a sphere, circle as intersection of a plane and a sphere, intersection of two spheres, equation of a tangent plane, condition of tangency and related problem. 7.4 Cone: Equation of a cone with given vertex and generator intersecting given conic related problems, Equation of right circular cone and related problems 7.5 Cylinder: Equation of a cylinder whose generator intersecting a given conic and parallel to the line $\frac{x}{l} = \frac{y}{m} = \frac{z}{m}$ and related problems, Equation of a right circular cylinder and related problems

5. List of Tutorials

The following tutorial activities of 30 hours per group of maximum 24 students shall be conducted to cover all the required contents of this course. This will enable the students to complete the related mathematical problems under the supervision of the subject teacher.

a) Problems on solution of system of linear equations by Gauss method (1 hr)
b) Determining rank of a matrix and test the consistency then solve the linear equations (2 hrs)
c) Problems on vector space and subspace (1 hr)
d) Linear dependence and independence, Linear transformation (1 hr)
e) Eigenvalues, eigenvectors and diagonalization, verifying Caley-Hamilton Theorem and its application in finding the inverse (2 hrs)
f) Simplex method for standard problems (2 hr)
g) Duality in LPP (1 hr)
h) Simplex method for mixed inequalities (Big-M method) (1 hr)
i) Dot and vector product of 3 and 4 vectors with geometrical interpretation (2 hrs)
j) Reciprocal system of vectors (1 hr)
k) To test for convergence of a series by different tests (3 hrs)
l) Finding centre of convergence, radius of convergence and interval of convergence (2 hrs)
m) To solve the problems on transformation of coordinates (1 hr)
n) Problems on ellipse and hyperbola (3 hrs)
o) Problems on straight lines (3 hrs)
p) Problems on sphere (2 hrs)
q) Problems related to cone and cylinder (2 hrs)

6. Evaluation System and Students' Responsibilities

Evaluation System

The internal evaluation of a student may consist of assignments, attendance, term-exams, and project works etc. The internal evaluation scheme for this course is as follows:

Internal Evaluation	**Weight**	**Marks**	**External Evaluation**	**Marks**
Theory		50		

Attendance & Class Participation	10%		Semester-End examination	50
Assignments	20%			
Presentations/Quizzes	10%			
Internal Assessment	60%			
Total Internal		50		
Full Marks: 50 + 50 = 100				

Students' Responsibilities

Each student must secure at least 45% marks in internal assessment 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) to appear 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.

7. Prescribed Books and References

Text Books

1. Kreyszig, E. *Advanced Engineering Mathematics*. New Delhi: John Wiley and Sons Inc.
2. Thomas, G. & Finney, R. *Calculus and Analytical Geometry*. New Delhi: Narosa Publishing House.

References

1. Swokoswski, E.W. *Calculus with Analytic Geometry*. Prindle, Weber and Schmidi.
2. Narayan, S. *Analytical Solid Geometry*. S. Chand and company.
3. Prasad, C. *Algebra and Theory of Equations*. Pothishala Pvt. Ltd.
4. Cheney, W. & Kincaid, D. *Linear Algebra: Theory and applications*. Jones and Bartlett Publisher.
5. Vittal, P. R. *Analytical Geometry 2D and 3D*, Delhi: Pearson India.

Pokhara University
Faculty of Science and Technology

Course Code: MEC 116
Course Title: Basic Engineering Drawing (0-0-3)
Nature of the Course: Practical
Level: Bachelor

Full Marks: 100
Pass Marks: 45
Total Duration: 45 hours
Program: BE

Course Description
This course is designed to provide students the knowledge and skills to draw, visualize and represent objects manually as well as with the application of computer aided techniques. The course will be delivered using tutorials and self-learnings by the students.

1. General Objectives
The general objectives of this course are:
- To enhance knowledge and skills to draw and visualize geometrical shapes of objects
- To enable students to draw, visualize and representation objects using Computer aided techniques.

2. Methods of Instruction
Lecture, discussions, demonstration, tutorials and assignments

3. Contents in Detail

Specific Objectives	Contents
Recognize the drawing instruments, drawing sheets, lettering and dimensioning.	**Unit I: Introduction to engineering drawing (3 hrs)** 1.1 Manual drawing instruments, drafting machines, drawing paper and materials, preparation for drawing, cautions in use of instruments, drawing sheets-their layout and planning 1.2 Technical lettering and dimensioning: Single-stroke letters, capital and lowercase letters, vertical and slant lettering, vertical and inclined numerals 1.3 Procedure for lettering, dimensioning terms and notations, theory of dimensioning, system of dimensioning, use of scales, units and general rules of dimensioning
Draw basic geometrical shapes	**Unit II: Geometrical Constructions (7 hrs)** 2.1 Construction involving lines and angles, bisecting and trisecting lines and angles, division of lines, proportional division of lines 2.2 Construction of polygons, constructions using tangents circles and arcs, open and cross belt tangents 2.3 Construction of conic and engineering curves: ellipse (four center method), parabola (Tangent method), hyperbola (eccentricity method), cycloid, involute, Archimedean spiral, helix.
Visualize objects through	**Unit III: Multi-view Drawings & Sectional Views (15 hrs)**

orthographic projections	3.1 Introduction to projection (point, line plane), orthographic projection 3.2 Selection of views, ways for making a multi-view drawing 3.3 Introduction of sectional views, half and full sectional views, offset sectional view, hatching lines
Illustrate the surface development	**Unit IV: Developments of surfaces (4 hrs)** 4.1 Introduction of surface development 4.2 Complete developments of truncated right solids (Cylinder, Cone, Pyramid and Prism), frustums of right solids (Cone & Pyramid)
Visualize and draw objects through isometric, oblique and perspective drawings	**Unit V: Isometric, Oblique and Perspective Projections (6 hrs)** 5.1 Introduction of Axonometric projection, isometric projection, methods and procedure for making an simple isometric drawing 5.2 Introduction of oblique projection and oblique drawing 5.3 Perspective projection, position of object, construction of one-point
Recognize symbols and use computer aided techniques to visualize and draw objects	**Unit VI: Symbols and Computer Aided Drawing (10 hrs)** 6.1 Electrical and Electronics symbols 6.2 Introduction to AutoCAD, Basic commands for 2D drawing like: Line, Circle, Polyline, Rectangle, Hatch, Fillet, Chamfer, Trim, Extend, Offset, Dim style, etc. 6.3 Basics of 3D drawings (In computer laboratory with only demonstration and practices)

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

4. Laboratory Work (45 hrs for a group of maximum 24 students)
Lettering and use of drawing instruments; Dimensioning; Geometrical and Projection drawing; Multiview drawings; Sectional views, Development of surfaces; Axonometric projection; Symbols and AutoCAD Drawing

5. Evaluation System and Students' Responsibilities

Evaluation System
The evaluation of a student may consist of attendance, assignments, term-exams, projects etc. The final examination will be held by the PU Examination Controller's Office. The internal and external evaluation detail is given in the table below:

Internal Evaluation	**Weight**	**Marks**	**External Evaluation**	**Marks**
Practical				
Attendance and Class Participation	10%		Semester-End examination	50
Drawing Sheets Evaluation	30%			
Assignment	10%			
Internal Assessment	50%			
Total Internal		50		
Full Marks: 50 + 50 = 100				

Students' 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 score will be given NOT QUALIFIED (NQ) to appear 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.

6. Prescribed Books and References

Text Book

1. Luzadder, W. J. & Duff, J. F (2015). *Fundamentals of Engineering Drawing with an introduction to Interactive Computer Graphics for Design and Production*, Pearson India Education Services.

References

1. Gill, P. S. (2009). *Engineering Drawing*, Seagull Books Pvt Ltd. India.
2. Dhawan, R. K. (2019). *A Textbook of Engineering Drawing*, S. Chand Publishing.
3. Omura, G. (2012). *Mastering AutoCAD 2013 and AutoCAD LT 2013*, John Wiley & Sons.

Draft Copy

Pokhara University
Faculty of Science and Technology

Course No.: CMP 160 (3 Credits)	Full marks: 100
Course title: Data Structure and Algorithms (3-1-3)	Pass marks: 45
Nature of the course: Theory and Practical	Total Lectures: 45 hrs
Level: Bachelor	Program: BE (Computer/Software/IT/Electronics and Communication)

1. Course Description

This course is designed to encompass the concepts of basic data structures- stack, queue, linked list, tree, graph etc., basic algorithm design techniques- divide and conquer, greedy algorithms etc. and algorithm analysis techniques to determine the cost of algorithms. It presents the various search and sorting algorithms that follow the divide and conquer and greedy strategy to solve the problems. This course also introduces the advance data structures such as hash table and B tree. After completion of this course, students can design and choose an appropriate data structure and efficient algorithm to achieve better performance.

2. General Objectives

- To acquaint the students with basic concepts of basic data structures such as stack, queue, linked list, tree and graph.
- To acquaint the students with concepts of sorting and searching algorithms.
- To acquaint the students with the knowledge of algorithms design techniques and algorithm analysis techniques.
- To develop the skills in students to choose the appropriate data structure and algorithm design technique for a specified application..
- To acquaint the students with the knowledge of the recursion, a popular problem solving technique, to solve the real world complex problems.

3. Methods of Instruction

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

4. Contents in Detail

Specific Objectives	Contents

• Understand the data structure, ADTs and algorithm design techniques. • Analyze the cost of algorithms.	**Unit 1: Introduction (5 hrs)** 1. Philosophy of Data Structures 1.1. Need of Data Structures 1.2. Characteristics and Types 2. Abstract Data Type (ADT) and Data Structures 3. Algorithm Design Techniques 3.1. Divide and Conquer 3.2. Greedy Algorithms 3.3. Backtracking 4. Algorithm Analysis: 4.1. Best, Worst and Average Case Analysis 4.2. Rate of Growth 4.3. Asymptotic Notations- Big Oh, Big Omega and Big Theta
• Implement the stack to solve various problems like expression evaluation and conversion. • Use the recursion to solve recursive problems.	**Unit 2: Stack and Recursion (7 hrs)** 1. Stack 1.1. Definition and Stack Operations 1.2. Stack ADT and its Array Implementation 1.3. Expression Evaluation: Infix and Postfix 1.4. Expression Conversion: Infix to Postfix and Postfix to Infix 2. Recursion 2.1. Recursion- A problem Solving Technique 2.2. Principle of Recursion 2.3. Recursive Algorithms- Greatest Common Divisor, Sum of Natural Numbers, Factorial of a Positive integer, Fibonacci Series and Tower of Hanoi 2.4. Recursion and Stack 2.5. Recursion vs Iteration 2.6. Recursive Data Structures 2.7. Types of Recursion 2.8. Applications of Recursion

Implement the queue and linked list to solve various problems.	**Unit 3: Queue and Linked List (10 hrs)** 1. Queue 1.1. Definition and Queue Operations 1.2. Queue ADT and its Array Implementation 1.3. Circular Queue and its Array Implementation 1.4. Double Ended Queue and Priority Queue 2. Linked List 2.1. List- Definition and List Operations 2.2. List ADT and its Array Implementation 2.3. Linked List- Definition and its Operations 2.4. Singly Linked List- Basic Operations, Singly Linked List ADT and Implementation of Singly Linked List 2.5. Doubly Linked List and Circular Linked List 2.6. Linked Implementation of Stack and Queue
• Understand the use and applications of Tree. • Construct the binary search tree, AVL trees and B trees.	**Unit 4: Tree (7 hrs)** 1. Definition and Tree Terminologies 2. General Trees 2.1. Definition and their Applications 2.2. Game Tree 3. Binary Trees 3.1. Definition and Types 3.2. Array and Linked List Representation 3.3. Traversal Algorithms: pre-order, in-order and post-order traversal 3.4. Application of Full Binary Tree: Huffman algorithm 4. Binary Search Tree: 4.1. Definition and Operations on Binary Search Tree: insertion, deletion, searching and traversing 4.2. Construction of Binary Search Tree 5. Balanced Binary Tree 5.1. Problem with unbalanced binary trees 5.2. Balanced Binary Search Tree 6. AVL tree 6.1. Definition and Need of AVL Tree 6.2. Construction of AVL tree: Insertion, Deletion on AVL tree and Rotation Operations 7. B Tree: Definition, Need and Application

• Understand and implement the various internal and external sorting algorithms.	**Unit 5: Sorting Algorithms (5 hrs)** 1. Internal/external Sort, Stable/Unstable Sort 2. Insertion and selection Sort 3. Bubble and Exchange Sort 4. Quick Sort and Merge Sort 5. Radix Sort 6. Shell Sort 7. Heap Sort as priority queue
• Understand and implement the sequential and binary search algorithms. • Design and implement the hash system for storing and searching data in hash table.	**Unit 6: Searching Algorithms and Hashing (5 hrs)** 1. Sequential Search 2. Binary Search 3. Hashing 3.1. Hash Function 3.2. Hash Table 3.3. Hashing as a Data Structure and a Search Technique 4. Collision in Hash Table 5. Collision Resolution Techniques 5.1. Open Hashing: Separate Chaining 5.2. Closed Hashing: Linear Probing, Quadratic Probing and Double Hashing 6. Load Factor and Rehashing
• Understand the concept of graph to represent real world problems and use it for finding minimum cost solution.	**Unit 7: Graphs (6 hrs)** 1. Definition, Terminologies and Types of Graphs 2. Representation of Graphs: Adjacency Matrix, Incidence Matrix and Adjacency list 3. Transitive Closure and Warshall's Algorithm 4. Graph Traversals: Breadth-First Search, Depth-First Search and Topological Sort 5. Minimum Spanning Tree: Kruskal's Algorithm and Prim's Algorithm 6. Shortest-Paths Problems: Types, Single-Source Shortest Path Problem- Dijkstra's Algorithm

5. Practical Works

Laboratory work of 45 hours per group of maximum 24 students should cover implementation of basic data structures, sorting algorithms and searching algorithms using C language or C++ language. Students should complete the following implementations in laboratory:

SN	Implementation Description
1	Implementation of stack using array.
2	Implementations of linear queue and circular queue using array.
3	Implementation of recursive algorithms- Greatest Common Divisor, Sum of Natural Numbers and Tower of Hanoi
4	Implementation of linked list: singly and doubly linked lists.
5	Implementation of stack and queue using linked list.
6	Implementation of in-order, pre-order and post-order tree traversals.
7	Implementation of insertion sort, bubble sort and quick sort.
8	Implementation of sequential, binary search and hash system.
9	Implementation of breadth-first search to traverse a graph and Kruskal's Algorithm to find the minimum spanning tree of a graph.
10	Implementation of Dijkstra's Algorithm.

Students should submit a project work that uses all the knowledge obtained from this course to solve any problem chosen by themselves. 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. Philosophy of data structure- Parking problem in narrow garage, Word Reversing Problem, and Need of Data structure (Class discussion)
   b. Algorithm Design Techniques (Class discussion).
   c. Need of Algorithm Analysis (Oral Presentation).

B. Problem solving-based Tutorials: (6 hrs)
   a. Design a system to generate the Huffman code for characters in a given text.
   b. Design a hash system that implements simple hash function, hash table and resolution solution techniques to minimize collisions.
   c. Suppose you are given a task to design a network system of transportation link or communication link or electricity transmission line (or choose any problem) in your city. Use the graph data structure to represent the problem and find the solution that has minimum cost to implement the system.

C. Review and Question/Answer-based Tutorials: (6 hrs)
   a. Case study on history of Fibonacci numbers and Tower of Hanoi and their recursive solutions. (Oral Presentation in class).

b. Case study on "Amount of resource demand of common growth rate functions" and "Comparison of algorithms using growth rates".
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	Semester-End examination	50
Attendance & Class Participation	10%			
Assignments	20%			
Presentations/Quizzes	10%			
Internal Assessment	60%			
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. Langsam, Y., Augenstein, M. J., & Tenenbaum, A. M. (1996). *Data Structures using C and C++*. Prentice Hall Press.
2. Rowe, G. W. (1997). I*ntroduction to data structures and algorithms with C++*. Prentice-Hall, India.

3. Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). *Introduction to algorithms*. MIT press.

References

1. Kruse, R. L., & Ryba, A. J. (1998). *Data structures and program design in C++*. Prentice Hall, India..
2. Brassard, G., & Bratley, P. (1996). *Fundamentals of algorithmics*. Prentice-Hall, India.

Draft Copy

Pokhara University
Faculty of Science and Technology

Course Code: ELE 172	Full Marks: 100
Course Title: Instrumentation (3-1-2)	Pass Marks: 45
Nature of the course: Theory & Practical	Total Lectures: 45 hours
Level: Bachelor	Program: BE

1. Course Description

This course is designed to provide students with basic concepts of Instrumentation and Measurements. After completion of this course the students will understand the fundamental concept of Analog and Digital instrumentation with signal conditioning, wave shaping and filtering. The theory part shall provide instruction on instrumentation and the practical part would be validation of most of the analyses and calculations covered in theory.

2. General Objectives

The course is designed with the following general objectives:

- To provide comprehensive idea about Analog and Digital Instrumentation.
- To provide comprehensive idea about signal conditioning and Data Acquisition System.

3. Methods of Instruction

3.1 **General Instructional Techniques:** Lectures, Tutorials, discussion, question-answer, brain storming, etc.

3.2 **Specific Instructional Techniques:** All the units of the course are practical oriented. Both the theory and practical classes are to be synchronized and a practical work shall be assigned for every theory unit.

4. Contents in Detail

Specific Objectives	Contents
Explain the Basic of Instrumentation, Bridge Measurement and Transducers	**Unit 1: Introduction to Instrumentation System (10 hrs)** 1.1 Typical applications of Instrument systems 1.2 Functional elements of Instrumentation and Measuring systems i.e., Input elements (Transducers and Electrodes), intermediate elements (signal conditioning) and output elements (Data display and storage). 1.3 Errors and uncertainties in Measurements and Static performance characteristics of instruments: 1.3.1 Introduction to errors and uncertainties in the measurement of performance parameters of instruments. 1.3.2 Static performance parameters: Accuracy, Precision, Resolution, Threshold, Sensitivity,

	Linearity, Hysteresis, Dead band, Backlash, Drift, Span 1.3.3 Impedance loading and matching 1.3.4 Errors: Statistical analysis of error in measurement 1.3.5 Standards of measurement 1.4 Bridge Measurement: 1.4.1 DC bridges- Wheat-stone bridge 1.4.2 AC bridges – Kelvin, Hay, Maxwell, Schering and Wien bridges 1.4.3 Wagner ground Connection 1.5 Physical Variable and Transducer 1.5.1 Physical Variable and their types (Electrical, Mechanical, Process and Biophysical) 1.5.2 Transducer principle and operation 1.5.3 Input and output characteristics and application of transducers 1.5.3.1 Resistive 1.5.3.2 Capacitive 1.5.3.3 Inductive 1.6 Measurement of mechanical variables, displacement, strain. velocity. acceleration and vibration 1.7 Measurement of process variables - temperature pressure, level, fluid flow, chemical constituents in gases or liquids, pH and humidity 1.8 Measurement of bio-physical variables blood pressure and myoelectric potentials 1.9 Calibration and error in transducers 1.10 Measurement of voltage & current (moving coil & moving iron instruments 1.11 Measurement of low, high & medium resistances
Explain the basis of Analog instruments and Principle of equipment used in measurement of electrical quantities	**Unit II: Principle of Analog Instruments (7 hrs)** 2.1 Review of DC/AC voltmeter and Ammeter: The D' Arsonval Principle 2.2 DC Multirange Ammeters and Extending Ammeter ranges 2.3 DC Multirange Voltmeters and Extending Voltmeters ranges 2.4 AC voltmeter and multi range voltmeter 2.5 Ohm Meter and Multirange 2.6 Electronic Multimeter 2.7 Multimeter as a micro ammeter and dc ammeter Types pf voltmeter: Differential type and True rms 2.8 Wattmeter: Types and Working principles 2.9 Energy Meter: Types and Working Principle 2.10 Power Factor Meter 2.11 Instrument Transformer

Explain about the Signal conditioning and transmission system	**Unit III: Electrical Signal Processing and Data Acquisition (7 hrs)** 3.1 Basic Op-amp characteristics 3.2 Instrumentation amplifier 3.3 Signal amplification, attenuation, integration, differentiation, network isolation, wave shaping 3.4 Effect of noise, analog filtering, digital filtering 3.5 Data Acquisition System 3.5.1 Analog Data Acquisition System 3.5.2 Digital Data Acquisition system 3.5.3 Single channel Data Acquisition system: 3.5.4 Multi-channel Data Acquisition system 3.5.5 PC based Data acquisition system 3.6 Series and Parallel transmission: 3.6.1 Features and application of RS232 cable 3.6.2 Features and application of IEEE 1248 B 3.7 Optical communication, fibre optics, electro-optic conversion devices
• Explain about the analog to Digital and Digital to Analog converter in depth	**Unit IV: Date Converter and Connectors (8 hrs)** 4.1 Analog to Digital Converter (ADC) and Digital to analog Converter (DAC): Principle and Specification 4.2 Quantization Error 4.3 Types of ADC 4.3.1 Flash type ADC 4.3.2 Counter type ADC 4.3.3 Successive Approximation Type ADC 4.3.4 Dual Slope ADC 4.3.5 Introduction to Delta-Sigma ADC 4.4 Types of DAC 4.4.1 Weighted Resistor DAC 4.4.2 R-2R Ladder DAC 4.4.3 PWM Type DAC 4.5 Probes and Connectors 4.5.1 Test Leads: Twisted pair unshielded test leads 4.5.2 Shielded Cables 4.5.3 Connectors 4.5.4 Low Capacitive Probes 4.5.5 High Voltage Probes 4.5.6 Current Probes
• Compare different types of wave analyzer and principle of Digital instrumentation.	**Unit V: Wave Analyzers and Digital Instruments (8 hrs)** 5.1 Wave Analyzer 5.1.1 Frequency Selective Wave Analyzer 5.1.2 Heterodyne Wave Analyzer 5.2 Spectrum Analyzer 5.2.1 Basic Spectrum Analyzer using Swept Receiver Design 5.2.2 IRF Spectrum Analyzer 5.3 Distortion Analyzer: Harmonic Distortion Analyzer-Fundamental Suppression Type

	5.4 Measurements of Frequency and Time: Decimal Count Assembles 5.5 Frequency Counter 5.6 Period Counter 5.7 Error: Counter Error and Signal Related Error 5.8 Digital Voltmeter 5.8.1 Ramp type digital voltmeter 5.8.2 Integrating type digital voltmeter 5.8.3 Servo Potentiometer type digital Voltmeter 5.8.4 Successive Approximation type digital Voltmeter 5.9 Vector Voltmeter 5.10 Digital Multimeter 5.11 Computer Based Digital Instruments: IEEE 488 GPIB Instrument
Differentiate different types of output devices used in instrumentation	**Unit VI: Recorders, Displays and Storage Devices (5 hrs)** 6.1 Oscilloscopes: 6.1.1 Cathode Ray Tube, Vertical and Horizontal Deflection Systems, Delay lines, Probes and Transducers, 6.1.2 Specification of an Oscilloscope 6.1.3 Oscilloscope measurement Techniques 6.2 Special Oscilloscopes – Storage Oscilloscope, Sampling Oscilloscope 6.3 Recorders Basic recording systems. Strip chart recorders. Galvanometer and Potentiometer type recorders (direct and null type) 6.4 Indicators and display Devices - Nixie, LED, LCD and seven segment and dot matrix displays. 6.5 Magnetic tape and disc recorders 6.6 Data loggers, Dot matrix and laser printers 6.7 Compact disc/Optical disc recorders

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:

Unit	Tutorial
I	Error Calculation Calculation of Dynamic behavior of instrumentation Calculation of Different Bridge Measurements Calculation of Transducers
II	Torque calculation of Wattmeter and Energy Meter Range Extension of Analog Voltmeter and Ammeter
III	Calculation and Design of Signal conditioning circuits Calculation and design of wave shaping circuits and filters
IV	Calculation of Resolution and Quantization error Calculation for ADC and DAC
V	Calculation on Counter Error and Signal Related Error

6. Laboratory Works

1. Accuracy test in Analog Meter
2. Operational amplifier in circuits: Use of Op-amp as
3. summer, inverter, integrator and differentiator
4. Use of Capacitive, inductive transducer to measure displacement
5. Use strain gauge transducer to measure force
6. Study of Various transducer for measurement of angular displacement. angular Velocity, pressure and flow
7. Use optical, Hall effect and inductive transducer to measure angular displacement
8. Use tacho-generator to measure angular velocity
9. Use RTD transducers to measure pressure and flow
10. Digital to Analog Conversion to Perform static testing of D/A converter
11. Analog to Digital Conversion to Perform static testing of A/D converter

7. Evaluation System and Students' Responsibilities

Evaluation System

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.

Internal Evaluation	Weight	Marks	External Evaluation	Marks
Theory		30	Semester-End examination	50
Attendance & Class Participation	10%			
Assignments	20%			
Presentations/Quizzes	10%			
Internal Assessment	60%			
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 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 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 Book

Helfrick, A. D. & Cooper, W. D. *Modern Electronic Instrumentation and Measurements Techniques*. Prentice Hall of India

Reference Books

1. Nihal Kularatna, Digital and Analogue Instrumentation testing and measurement, The Institution of Engineering and Technology, London, United Kingdom n First published 2003 Reprinted 2008
2. A K. Sawhney, A Course in Elec. & Electronics Measurements & Instrumentation, Dhanpat Rai and Sons India, 1998
3. Joshph J.Carr,Elements of Electronics Instrumentation and Measurement-3rd Edition by.Pearson Education.

Draft Copy

Pokhara University
Faculty of Science and Technology

Course No.: CMP 162 (3 Credits)	Full Marks: 100
Course Title: Object Oriented Programming in C++ (3-1-3)	Pass Marks: 45
Nature of the Course: Theory and Practical	Total Lectures: 45 hrs
Level: Bachelor	Program: BE (Computer / Software/ Information Technology)

1. Course Description

This course is designed to encompass the concept of Object Oriented Programming to implement its important features such as data abstraction, encapsulation, inheritance, polymorphism, generic programming, exception handling and file handling using the object-oriented programming language called the C++ language.

2. General Objectives

- To familiarize the students with the basic concepts of Object Oriented Programming.
- To acquaint the students with the knowledge of features of C++.
- To develop the skills in students to solve the problems using Object Oriented concepts using C++.

3. Methods of Instruction

Lectures, Discussions, Readings, Practical works, and Project works.

4. Contents in Detail

Specific Objectives	Contents
● Understand the basic concepts of Object Oriented Analysis and Design.	**Unit 1: Object Oriented Concepts (8hrs)** 1. Object Oriented Programming Paradigm 2. A way of viewing World Agent 3. Procedure Oriented vs Object-Oriented Programming 4. Features of Object Oriented Programming: Class and Object, Data Abstraction, Encapsulation, Inheritance, Polymorphism, Message passing 5. Computation as Simulation, Coping with Complexity and Abstraction Mechanisms 6. Object Oriented Analysis and Design: Introduction, Responsibility Driven Design (RDD), Component Responsibility and Collaborator (CRC) Cards, Responsibility Implies Non-Interference, Programming in Small and Programming in Large
● Implement the use of class, object, and method, Data Abstraction, Encapsulation,	**Unit 2: Classes and Objects (8hrs)** 1. Introduction to C++: Origin of C++, Basic C++ Program Structure, Console Input/output Streams and Manipulators 2. Structure in C and C++ 3. Classes and Objects

message passing, data hiding in C++. ● Understand and implement the concept of constructor, destructor, memory allocation and advanced functions in C++.	4. Array of Objects 5. Class Diagram and Object Diagram 6. Access Specifiers and Visibility Mode 7. State and Behavior, Methods and Responsibilities 8. Implementation of Data Abstraction, Encapsulation, Message Passing and Data Hiding 9. Memory Allocation for Objects 10. Constructor: Default Constructor, Parameterized Constructor, Copy Constructor 11. Constructor Overloading 12. Destructors 13. Dynamic Memory Allocation: new and delete. 14. Dynamic Constructor 15. Functions: Inline function, Default argument, Passing and Returning by Value, Pointer and Reference, Static Data Member and Static Member Function 16. Friend Function and Friend Class
● Understand and implement the concept of software reusability using inheritance in C++.	**Unit 3: Inheritance (8hrs)** 1. Introduction to Inheritance 2. Inheritance Relationship Diagram 3. Inheritance Mode: Public, Private & Protected 4. Types of Inheritance: Single, Multilevel, Hierarchical, Multiple and Hybrid 5. Ambiguity Resolution 6. Multipath Inheritance and Virtual Base Class 7. Constructor and Destructor in Derived Class 8. Subclass, Subtype and Principle of Substitutability 9. Composition and its Implementation 10. Composition Relationship Diagram 11. Software Reusability
● Understand and implement the concept of polymorphism in C++.	**Unit 4: Polymorphism (9 hrs)** 1. Introduction to Polymorphism 2. Types of Polymorphism: Compile Time Polymorphism: Function Overloading, Operator Overloading Runtime Polymorphism: Virtual Function 3. Overloading Unary and Binary Operators 4. Function Overriding 5. this Pointer and Object Pointer 6. Pure Virtual Function, Abstract Class 7. Virtual Destructor 8. Type Conversion: Basic to User-Defined, User-Defined to Basic, User-defined to User-Defined
● Understand and implement the basic concept of generic programming and Standard Template Library (STL) in C++.	**Unit 5: Templates (5 hrs)** 1. Generic Programming 2. Class Template and Function Template 3. Standard Template Library (STL): Container, Algorithm, Iterator

● Understand and implement file stream operations and exception handling in C++.	**Unit 6: Exception Handling and Stream I/O (7 hrs)** 1. Exception Handling: Error and Exception, Exception Handling Mechanism (try, throw, and catch), Multiple Exception Handling 2. File Handling: Stream Class Hierarchy, Opening and Closing a File, Reading and Writing Object

5. Practical Works

Laboratory works of 45 hours per group of maximum 24 students should cover all the concepts of Object Oriented Programming in C++ language studied in the lectures. Students should submit a final project that uses all the constructs and features of Object Oriented Programming language using C++. The evaluation of the practical works should also be based on project work.

6. List of Tutorials

The various tutorial activities that suit this course should cover all the contents of this 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 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 (3hrs)
   1. Evolution of Object Oriented Programming Paradigm.
   2. The way of viewing the world in Object Oriented Programming.
   3. The features of Object Oriented Programming.
   4. Object Oriented Analysis and Design.

B. Problem solving base tutorials (8 hrs)
   1. Write a program to illustrate class, access specifiers, objects, encapsulation, data hiding and data abstraction in C++.
   2. Write a C++ program to illustrate the array of objects.
   3. Write a C++ program to illustrate dynamic memory allocation.
   4. Write a C++ program to illustrate the use of static data members and static member functions.
   5. Write a C++ program to illustrate the use of friend function and friend class.
   6. Write a C++ program to illustrate the use of virtual base class.
   7. Write a C++ program to illustrate the overloading of unary and binary operators using and without using friend function.
   8. Write a C++ program to illustrate the use of STL-Vector and its operations.
   9. Write a C++ program to illustrate how to catch all the exceptions.
   10. Write a C++ program to illustrate reading and writing into multiple files.

C. Review and Question/Answer-based Tutorials (4hrs)

   1. Case Study on origin of C++ languages.
   2. Case study on “Responsibility Driven Design- Interactive Intelligent Kitchen Helper”.
   3. 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	Semester-End examination	50
Attendance & Class Participation	10%			
Assignments	20%			
Presentations/Quizzes	10%			
Internal Assessment	60%			
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.

Text Books:

1. Lafore, R. (2001). *Object-oriented programming in Turbo C++*. Galgotia publications.
2. Budd, T. (2008). *Introduction to object-oriented programming*. Pearson Education India.
3. Balagurusamy, E., Balagurusamy, E., & Balagurusamy, E. (2008). *Object oriented programming with C++* (Vol. 4). Tata McGraw-Hill.

References:

1. Parsons, D. (2001). *Object-oriented Programming with C++*. Cengage Learning EMEA.
2. Schildt, H. (2003). *C++: The complete reference*. McGraw-Hill.