Kanpur Institute of Technology

*(*An Autonomous Institute of Dr. A.P.J. Abdul Kalam Technical University, Lucknow)

Accredited grade 'A' by NAAC Approved

by AICTE, New Delhi

Autonomous status approved by UGC, New Delhi

Evaluation Scheme & Syllabus

For

B.Tech. First Year

- **Computer Science & Engineering**
- **Information Technology**
- **Computer Science and Engineering (Artificial Intelligence & Machine Learning)**

(Effective from the Session: 2024-25)

Department of Computer Science & Engineering

___

A-1, UPSIDC Industrial Area, Rooma, Kanpur 208001(U.P.) India

Website: www.kit.ac.in/kit, E-Mail: info@kit.ac.in

Semester II

S. NO	COURSE CODE	COURSE TITLE	TYPE	L	T	P	FA	SA	Total	CREDIT
1	AH12010	ENGINEERING MATHEMATICS-II	TH	2	1	0	70	30	100	3
2	AH12020	TECHNICAL COMMUNICATION	TH	3	0	0	70	30	100	3
3	CS12010	BASICS OF PYTHON PROGRAMMING	TH	2	1	0	70	30	100	3
4	CS12020	OOPS WITH C++	TH	3	1	0	70	30	100	4
5	MO12010	MOOC-01 (ONE COURSE ON C/PYTHON/C++) FROM SWAYAM PORTAL)	TH	2	0	0	70	30	100	2
6	CS12030	BASICS OF PYTHON PROGRAMMING LAB	PR	0	0	4	70	30	100	2
7	CS12040	OOPS WITH C++ LAB	PR	0	0	4	70	30	100	2
8	CA12010	CO-CURRICULAR ACTIVITIES	PR	0	0	0	100	-	100	0.5
9	GP12010	GENERAL PROFICIENCY	PR	0	0	0	100	-	100	0.5
		Total ->		12	3	8	690	210	900	20

Abbreviation used:

L: Lecture	**FA**: **F**ormative **A**ssessment
T: Tutorial	**SA**: **S**ummative **A**ssessment
P: Practical	**TH**: **Th**eory, **PR**: **Pr**actical

B.TECH. FIRST YEAR (SEMESTER-II)					
Course Code	AH12010	**L**	**T**	**P**	**Credit**
Course Title	**ENGINEERING MATHMATICS II**	**2**	**1**	**0**	**3**

Course Objectives:

The objective of this course is to familiarize the prospective engineers with techniques in sequences, multivariate integration, ordinary and partial differential equations and complex variables. It aims to equip the students to deal with advanced level of mathematics and applications that would be essential for their disciplines.
The students will learn:
- The effective mathematical tools for the solutions of differential equations that model physical processes
- To applying basis concept of linear algebra.
- The tool of Fourier series for learning advanced Engineering Mathematics.
- The tools of differentiation of functions of complex variables that are used in various techniques dealing with engineering problems.
- The tools of integration of functions of complex variables that are used in various techniques dealing with engineering problems.

Pre-requisites: To effectively follow this syllabus, students should have a solid foundation in calculus, including differentiation and integration, as well as a basic understanding of first-order differential equations. Familiarity with linear algebra concepts such as matrices, determinants, vector spaces, and linear transformations is essential. Additionally, knowledge of sequences and series, particularly convergence tests, is important. For the units on complex variables, students should have a basic grasp of complex numbers, limits, continuity, and analytic functions. These prerequisites will enable students to understand and master the higher-order concepts covered in the syllabus.

Course Contents / Syllabus		
UNIT-I	**Ordinary Differential Equation of Higher Order**	**6 hours**
Linear differential equation of nth order with constant coefficients, Simultaneous linear differential equations, Second order linear differential equations with variable coefficients, Solution by changing independent variable, Reduction of order, Normal form, Method of variation of parameters, Cauchy-Euler equation.		
UNIT-II	**Linear Algebra**	**6 hours**
Linear dependent & independent, Basis and dimension, dimension theorem, linear transformation, RANK- nullity (statement only), and computation of rank and Nullity of Linear transformation, solution of linear simultaneous Algebraic equation.		
UNIT-III	**Sequences and Series**	**6 hours**
Definition of Sequence and series with examples, Convergence of sequence and series, Tests for convergence of series, (D'Alembert's test (Ratio test), Raabe's test). Fourier series, Half range Fourier sine and cosine series.		
UNIT-IV	**Complex Variable–Differentiation**	**6 hours**
Limit, Continuity and differentiability, Functions of complex variable, Analytic functions, Cauchy- Riemann equations (Cartesian and Polar form), Harmonic function, Method to find Analytic functions, Conformal mapping, Mobius transformation and their properties. Milne's Thomson method		
UNIT-V	**Complex Variable–Integration**	**6 hours**
Complex integrals, Contour integrals, Cauchy- Integral theorem, Cauchy integral formula, Taylor's and Laurent's series (without proof), Singularities, Classification of Singularities, zeros of analytic functions, Residues, Methods of finding residues, Cauchy Residue theorem		

Course Outcome (CO)		Knowledge Level
At the end of course, the student will be able to:		
CO1	Understand the concept of differentiation and apply for solving differential equations.	K_2 & K_3
CO2	Remember the concept of basic linear Algebra.	K_1, K_3 & K_5
CO3	Understand the concept of convergence of sequence and series. Also Evaluate Fourier series	K_2 & K_5
CO4	Illustrate the working methods of complex functions and apply for finding Analytic functions.	K_3
CO5	Apply the concept of complex functions for finding Taylor's series, Laurent's series and evaluation of definite integrals.	K_3 & K_5

TEXT BOOKS:
1. B.V. Ramana, Higher Engineering Mathematics, Tata McGraw-Hill Publishing Company Ltd., 2008.
2. B.S. Grewal, Higher Engineering Mathematics, Khanna Publisher,2005.
3. R.K. Jain & S.R.K. Iyenger, Advance Engineering Mathematics, Narosa Publishing-House, 2002
REFERENCE BOOKS:
1. E.Kreyszig, Advance Engineering Mathematics, John Wiley & Sons,2005.
2. Peter V.O'Neil, Advance Engineering Mathematics, Thomson(Cengage)Learning, 2007.
3. Maurice D.Weir, JoelHass, FrankR.Giordano, Thomas, Calculus, EleventhEdition, Pearson.
4. G.BThomas,R LFinney, Calculusand Analytical Geometry, Ninth Edition Pearson,2002.
5. James Ward Brown and Ruel VChurchill, Fourier Series and Boundary Value Problems, 8th Edition-McGraw-Hill
6. D.Poole, LinearAlgebra: AModernIntroduction, 2nd Edition, Brooks/Cole, 2005.
7. VeerarajanT.,Engineering Mathematics for first year,McGraw-Hill,NewDelhi, 2008.
8. Charles E Roberts Jr, Ordinary Diffrential Equations, Application, Model and Computing, CRC Press T&F Group.
9. RayWylie Cand LouisCBarret, Advanced Engineering Mathematics, 6thEdition, McGraw-Hill.
10. James Ward Brownand RuelV Churchill, Complex Variable and Applications, 8thEdition, McGraw-Hill.
11.P.Sivaramakrishna DasandC.Vijayakumari, Engineering Mathematics, 1stEdition, Pearson India Education Services Pvt. Ltd.
12. Advanced Engineering Mathematics By Chandrika Prasad, Reena Garg Khanna Publishing House, Delhi
13.E.Kreyszig, Advance Engineering Mathematics, JohnWiley&Sons,2005.
14. PeterV.O'Neil, Advance Engineering Mathematics, Thomson(Cengage)Learning, 2007.
15. MauriceD.Weir,JoelHass, FrankR.Giordano, Thomas, Calculus, Eleventh Edition,

Pearson.
16.G.BThomas,R LFinney,Calculusand Analytical Geometry,Ninth Edition Pearson,2002.
17. James Ward Brown andRuel VChurchill,Fourier Series and Boundary Value Problems, 8th Edition-McGraw-Hill
18.D.Poole,LinearAlgebra: AModernIntroduction,2nd Edition, Brooks/Cole, 2005.
19. VeerarajanT., Engineering Mathematics for first year, McGraw-Hill, NewDelhi, 2008.
SUGGESTED VIDEO LINKS:

1. https://youtu.be/tHqx1qxA8q4?feature=shared
2. https://youtu.be/Y8CF01gbd_U?feature=shared
3. https://www.youtube.com/watch?v=1HKn9IvYiEM&list=PLU6SqdYcYsfI5CvOHMVOxxMWdAKWG67Xq
4. https://youtu.be/K1LPzOCIwY8?feature=shared
5. https://youtu.be/J7kxzE2vqwg?feature=shared
6. https://youtu.be/bTeWAkfPbYY?feature=shared
7. https://youtu.be/1HKn9IvYiEM?feature=shared
8. https://youtu.be/rDrQH9ma82o?feature=shared
9. https://youtu.be/Pc2dWW3aSrk?feature=shared
10. https://youtu.be/1fLxcBJB_t8?feature=shared
11. https://youtu.be/HnFFuXDi9p8?feature=shared
12. https://youtu.be/Gkit1hUTsX8?feature=shared
13. https://www.youtube.com/playlist?list=PLU6SqdYcYsfIAbLwyyWUiWJPV6qFbPC13
14. https://www.youtube.com/watch?v=vfg1mmsSMMM&list=PLU6SqdYcYsfJqbZvQECrwnlQrp4fg6isX
15. https://youtu.be/9sLsX9DV5Fs?feature=shared
16. https://www.youtube.com/watch?v=CDeT2yV_CbM&list=PLU6SqdYcYsfJfRHH0K1CgNkheUKk2fBb6
17. https://www.youtube.com/watch?v=Sq2FhCxcyI8&pp=ygUqaGFsZiByYW5nZSBmb3VyaWVyIHNpbmUgYW5kIGNvc2luZSBzZXJpZXMg
18. https://www.youtube.com/watch?v=0RkcZoCaF5Q&list=PLVCBPCYGv7bCFGhx7shED8m33ZWHXlWch
19. https://www.youtube.com/watch?v=t9xW7UaZwZ0&list=PLU6SqdYcYsfI3sh-ho_iiTkCGsTbVh_Sw
20. https://youtu.be/OQQqbV32b78?feature=shared
21. https://youtu.be/ABNW9FcZ5zQ?feature=shared
22. https://youtu.be/xgnQTqMc6A4?feature=shared
23. https://youtu.be/xaKTNENeJdA?feature=shared
24. https://youtu.be/fSoQxuVdKIs?feature=shared
25. https://youtu.be/xls_5Ly7VA4?feature=shared
26. https://www.youtube.com/watch?v=G1Wc5gGAZ8A&list=PLU6SqdYcYsfIBWXp-5lybgsBc9My0ejma
27. https://youtu.be/4yC4IXcMKJg?feature=shared

B.TECH. FIRST YEAR (SEMESTER-II)					
Course Code	**AH12020**	**L**	**T**	**P**	**Credit**
Course Title	**TECHNICAL COMMUNICATION**	3	0	0	3

Course Objectives:

1: Students will establish a strong foundation in English grammar by focusing on core and advanced parts of speech, which are crucial for effective communication. They will gain insights into the importance of communication in the professional world, exploring various forms such as written, verbal, and non-verbal communication. Emphasis will be placed on understanding communication etiquette and overcoming barriers to ensure clear and impactful interactions.

2: Students will refine their business writing skills, covering essential areas such as email etiquette, memo, report, and letter writing, while mastering core grammar concepts like sentence structure and subject-verb agreement. They will develop a professional writing style, enhance their editing and proofreading abilities, and improve active listening skills to communicate effectively in a business environment.

3: Students will advance their grammatical proficiency by mastering complex concepts, including various tenses, and simultaneously develop their public speaking skills. They will learn to structure and deliver impactful presentations, manage public speaking anxiety, and utilize body language and visual aids effectively. The course will also focus on preparing students to handle Q&A sessions confidently and address challenging questions with composure.

4: To elevate students' grammatical skills, particularly in direct and indirect speech, modal verbs, and question tags. It will deepen their understanding of non-verbal communication by interpreting body language, facial expressions, and cultural differences in non-verbal cues. Additionally, students will learn to build strong professional relationships through effective networking and team communication strategies.

5: Students will develop advanced grammar skills, focusing on voice mechanics and sentence transformation for precise communication. They will also gain cross-cultural communication competencies, enabling them to navigate and respect cultural differences in diverse workplaces. The course will cover professional etiquette, including business and dining protocols, preparing students for effective interactions in global business contexts.

Pre-requisites:
Students should have basic English proficiency and introductory grammar knowledge, including core parts of speech and sentence structure. Familiarity with fundamental communication principles and elementary writing skills is also important. Additionally, some experience with public speaking will help in mastering advanced topics in the syllabus.

Course Contents/Syllabus		
UNIT-I	**Foundations of Professional Communication and English Grammar Essentials**	**8 hours**

Fundamentals of English Grammar: Core Parts of Speech: Nouns, Pronouns, and Verbs, Advanced Parts of Speech: Adjectives, Adverbs, Prepositions, Conjunctions, and Interjections

Key Aspects of Professional Communication and Etiquette: Understanding the

Importance of Communication in the Professional World, Overview of communication skills, Impact of effective communication on career success, Dressing Etiquettes, Types of Professional Communication, Written communication, Verbal communication, Non-verbal communication, Barriers to Effective Communication

UNIT-II	Comprehensive Business Communication and Core Grammar Skills	8 hours

Core Grammar Concepts: Sentences – Meaning and Types Subject Verb Agreement
Business Writing and Effective Communication Techniques: Basics of Business Writing, Email etiquette, Memo and report writing, Crafting Effective Business Letters, Structure and format, Tone and language, Professional Document Editing and Proofreading, Editing techniques, Importance of proofreading, Developing a Writing Style, Clarity and conciseness, Tone and voice in professional writing, Understanding Effective Listening, Importance of active listening, Techniques for improving listening skills

UNIT-III	Public Speaking Excellence and Grammar Precision	8 hours

Advanced Grammar Concepts: Fundamentals of Tenses: Simple and Continuous Forms, Advanced Tenses: Perfect and Perfect Continuous Forms.

Verbal Communication and Public Speaking: Public Speaking Basics, Overcoming public speaking anxiety, Structuring a presentation, Presentation Delivery, Body language and gestures, Using visual aids effectively, Handling Q&A Sessions, Responding to questions confidently, Dealing with challenging questions

UNIT-IV	Advanced Communication Skills: Grammar, Non-Verbal Cues, and Professional Relationships	8 hours

Advanced Grammar Mastery: Direct and Indirect Speech: Rules, Conversions and Applications, Modal Verbs and Question Tag Usage.
Advanced Non-Verbal Communication and Professional Relationship Building: Understanding Non-Verbal Cues, Body language, Facial expressions and gestures, Interpreting Non-Verbal Signals, Recognizing hidden meanings, Cultural differences in non-verbal communication, Building Effective Professional Relationships, Networking skills, Team communication.

UNIT-V	Cross-Cultural Competence and Grammar Mastery	8 hours

Advanced Grammar: Voice and Transformation of Sentences: Voice Mechanics: Active and Passive Forms, Transformation of Sentences: Definition, Rules and Conversions.
Cross-Cultural Communication and Professional Etiquette: Cross-Cultural Communication, Understanding and respecting cultural differences, Communicating in diverse workplaces, Business Etiquettes, Dining Etiquettes.

Course Outcome (CO)		Knowledge Level
At the end of course, the student will be able to:		
CO1	**Foundations of Professional Communication and English Grammar Essentials**: Students will gain a solid understanding of English grammar basics and core parts of speech, develop effective professional communication skills across various mediums, and apply appropriate etiquette in both written and verbal interactions to enhance their overall professional effectiveness.	K2
CO2	**Comprehensive Business Communication and Core Grammar Skills**: Students will refine their business writing and communication techniques, mastering essential grammar concepts and business writing formats, while enhancing their active listening skills and ability to convey clear and concise messages in a professional setting.	K3

CO3	**Public Speaking Excellence and Grammar Precision**: Students will demonstrate advanced grammatical proficiency and effective public speaking skills, including overcoming anxiety, structuring and delivering presentations, and handling Q&A sessions with confidence, utilizing body language and visual aids to engage their audience effectively.	K3
CO4	**Advanced Communication Skills: Grammar, Non-Verbal Cues, and Professional Relationships**: Students will master advanced grammar topics such as direct and indirect speech, interpret and apply non-verbal communication cues effectively, and build strong professional relationships through improved networking skills and understanding of non-verbal signals in diverse work environments.	K4
CO5	**Cross-Cultural Competence and Grammar Mastery**: Students will apply advanced grammar concepts, including the use of modal verbs and voice mechanics, effectively in both written and spoken communication, and navigate cross-cultural communication challenges and professional etiquette to enhance their adaptability in global business contexts.	K5

TEXT BOOKS:
1. "Business Communication: Process and Product" by Mary Ellen Guffey and Dana Loewy
2. "Technical Communication: A Reader-Centered Approach" by Paul V. Anderson
REFERENCE BOOKS:
1. "The Gregg Reference Manual: A Manual of Style, Grammar, Usage, and Formatting" by William A. Sabin
2. "Effective Business Communication" by Herta A. Murphy, Herbert W. Hildebrandt, and Jane P. Thomas
3. "Communicating in Business" by Scot Ober
4. "Intercultural Communication in the Global Workplace" by Linda Beamer and Iris Varner
5. "The Handbook of Communication Skills" edited by Owen Hargie
SUGGESTED VIDEO LINKS:
1. https://www.youtube.com/watch?v=W-4IcNJIyM8&list=PLvbKJaHKFw3ZYTp2Fc9cj2LwZtIbOd5ux
2. https://www.youtube.com/watch?v=fyzmCU931QE&list=PLnyJK8kcdCduqkQ7rWxHyIic6_nOeUS5W&index=1
3. https://www.youtube.com/results?search_query=Public+Speaking+Excellence+and+Grammar+Precision
4. https://www.youtube.com/watch?v=akfatVK5h3Y
5. https://www.youtube.com/watch?v=In7LHa4fcAg

B.TECH. FIRST YEAR (SEMESTER-II)					
Course Code	CS12010	L	T	P	Credit
Course Title	BASICS OF PYTHON PROGRAMMING	2	1	0	3

Course Objectives:

The Python programming language course for B.Tech 1st-year students aims to provide a comprehensive introduction to programming concepts using Python. Through this course, students will develop a solid understanding of Python syntax and semantics, enabling them to write clear and concise code. Emphasis will be placed on problem-solving skills, with students learning to implement algorithms and data structures in Python to solve real-world problems. Practical application development will be a key focus, empowering students to create various projects such as games, web applications, and automation tools. Additionally, students will gain exposure to collaborative coding practices using version control systems like Git and will be introduced to popular Python libraries and frameworks. Throughout the course, ethical considerations in software development and professional practices such as documentation and testing will be emphasized, preparing students for future advanced courses and careers in computer science and engineering.

Pre-requisites:

The Python programming language course for B.Tech 1st-year students requires a basic understanding of computer literacy, including file management and navigating operating systems, as well as foundational mathematical knowledge such as arithmetic operations and logic. Critical thinking skills are essential for problem-solving and logical reasoning, while a basic level of English proficiency aids in understanding programming concepts and documentation. Previous programming experience is not necessary, as the course is designed to accommodate beginners, but students with prior exposure may find it advantageous. These prerequisites collectively ensure that students are equipped with the necessary skills and mindset to engage effectively with Python programming concepts and problem-solving techniques.

Course Contents/Syllabus		
UNIT-I	Foundations of Python Programming: From History to Setup and Syntax	8 hours
	Overview of Python: History and Features, Python 2 vs Python 3, Setting up the Python Environment: Installing Python, IDEs and Text Editors, Basic Syntax and Data Types: Variables and Data Types, Operators, Input/Output.	
UNIT-II	Control Flow and Functions	8 hours
	Control Flow: Conditional Statements (if, elif, else), Looping Constructs (for, while), Exception Handling, Functions: Defining Functions, Function Parameters and Return Values, Scope and Lifetime of Variables, File Handling: Reading and Writing to Files, File Modes and Operations	
UNIT-III	Data Structures in Python	8 hours
	Operations and Methods, Slicing and Indexing, Creating and Accessing Dictionaries, Dictionary Methods, Sets, Creating and Manipulating Sets, Set Operations	
UNIT-IV	Object-Oriented Programming (OOP) in Python	10 hours
	Introduction to OOP: Classes and Objects, Inheritance and Polymorphism, Advanced OOP Concepts: Encapsulation and Abstraction, Class Methods and Static Methods	
UNIT-V	Advanced Topics and Applications	8 hours
	Modules and Packages: Creating and Using Modules, Organizing Code into Packages, Regular Expressions: Pattern Matching with re module, String Manipulation, Introduction to GUI Programming: Basics of Tkinter. Web Development with Flask (optional): Basics of Web Development	

Course Outcome (CO)		Knowledge Level
At the end of course, the student will be able to:		
CO1	Students will demonstrate a comprehensive understanding of Python programming fundamentals, including its history, features, and differences between Python 2 and Python 3, enabling them to make informed choices in selecting the appropriate version for their projects.	K2,K3
CO2	Upon completion of the course, students will proficiently set up Python environments, install necessary software, and effectively utilize integrated development environments (IDEs) and text editors, facilitating a conducive coding environment for efficient programming tasks.	K3
CO3	Learners will exhibit competency in basic Python syntax and data types, including variables, operators, and input/output operations, thereby enabling them to write clear, concise, and functional code to solve a variety of computational problems.	K3
CO4	Upon mastering control flow mechanisms such as conditional statements, looping constructs, and exception handling, students will demonstrate the ability to develop robust Python programs that effectively manage program flow and handle errors gracefully, ensuring the reliability and stability of their applications.	K3
CO5	By the conclusion of the course, students will possess advanced skills in object-oriented programming (OOP) concepts, including class and object creation, inheritance, polymorphism, encapsulation, and abstraction, equipping them to design and implement sophisticated software solutions using OOP principles in Python.	K4

TEXTBOOKS
(1) Python Crash Course" by Eric Matthes
(2) Learning Python" by Mark Lutz
(3) Automate the Boring Stuff with Python" by Al Sweigart
REFERENCEBOOKS:
(1) Python Programming: An Introduction to Computer Science" by John Zelle
(2) Effective Python: 90 Specific Ways to Write Better Python" by Brett Slatkin
(3) Fluent Python: Clear, Concise, and Effective Programming" by Luciano Ramalho
(4) Python Cookbook" by David Beazley and Brian K. Jones
SUGGESTED VIDEO LINK:

1. https://www.youtube.com/watch?v=9MmC_uGjBsM&list=PLyqSpQzTE6M_Fu6l8irVwXkUyC9Gwqr6_
2. https://www.youtube.com/watch?v=Bc___4fV94lE&list=PLyqSpQzTE6M_Fu6l8irVwXkUyC9Gwqr6_&index=2
3. https://www.youtube.com/watch?v=TURlDFwVeEs&list=PLyqSpQzTE6M_Fu6l8irVwXkUyC9Gwqr6_&index=3
4. https://www.youtube.com/watch?v=Muzbm6FFS90&list=PLyqSpQzTE6M_Fu6l8irVwXkUyC9Gwqr6_&index=4
5. https://www.youtube.com/watch?v=CeHWTBQpiFE&list=PLyqSpQzTE6M_Fu6l8irVwXkUyC9Gwqr6_&index=5
6. https://www.youtube.com/watch?v=I7zOYKF4AHc&list=PLyqSpQzTE6M_Fu6l8irVwXkUyC9Gwqr6_&index=6
   https://www.youtube.com/watch?v=ffMgNo17Ork&list=PLyqSpQzTE6M_Fu6l8irVwXkUyC9Gwqr

6_&index=7

7. https://www.youtube.com/watch?v=ffMgNo17Ork&list=PLyqSpQzTE6M_Fu6l8irVwXkUyC9Gwqr6_&index=7
8. https://www.youtube.com/watch?v=oe6iF3yzMo8&list=PLyqSpQzTE6M_Fu6l8irVwXkUyC9Gwqr6_&index=8
9. https://www.youtube.com/watch?v=oe6iF3yzMo8&list=PLyqSpQzTE6M_Fu6l8irVwXkUyC9Gwqr6_&index=8
10. https://www.youtube.com/watch?v=J_Cw3G5v460&list=PLyqSpQzTE6M_Fu6l8irVwXkUyC9Gwqr6_&index=10
11. https://www.youtube.com/watch?v=lOhshXnJAJw&list=PLyqSpQzTE6M_Fu6l8irVwXkUyC9Gwqr6_&index=12
12. https://www.youtube.com/watch?v=0y5HOotxpys&list=PLyqSpQzTE6M_Fu6l8irVwXkUyC9Gwqr6_&index=14
13. https://www.youtube.com/watch?v=rP9XZRGPvdQ&list=PLyqSpQzTE6M_Fu6l8irVwXkUyC9Gwqr6_&index=34
14. https://www.youtube.com/watch?v=rP9XZRGPvdQ&list=PLyqSpQzTE6M_Fu6l8irVwXkUyC9Gwqr6_&index=34
15. https://www.youtube.com/watch?v=D __ znhgteJ0&list=PLyqSpQzTE6M_Fu6l8irVwXkUyC9Gwqr6_&index=37

B.TECH. FIRST YEAR (SEMESTER-II)					
Course Code	CS12020	L	T	P	Credit
Course Title	OOPS WITH C++	3	1	0	4

Course Objectives:
The objective of the Object-Oriented Programming (OOP) course with C++ is to equip students with a comprehensive understanding of OOP principles and proficiency in C++ syntax, enabling them to design and implement classes effectively while emphasizing encapsulation, inheritance, polymorphism, and abstraction. Through practical exercises and projects, students will learn memory management techniques, exception handling, file I/O operations, and how to leverage the Standard Template Library (STL) for efficient data structure and algorithm implementations. Additionally, they will gain familiarity with object-oriented design patterns, testing methodologies, and collaboration workflows, ensuring they can develop robust, maintainable software solutions that adhere to best practices and coding standards.

Pre-requisites: Basic knowledge of C

Course Contents / Syllabus		
UNIT-I	**Introduction to C++ and OOP and Elements of C++ Language**	**8 hours**
Introduction to C++ and OOP: Basic concept of OOP, Comparison of Procedural Programming and OOP, Benefits of OOP, C++ compilation, Abstraction, Encapsulation, Inheritance, Polymorphism, Difference between C and C++. **Elements of C++ Language :** Tokens and identifiers: Character set and symbols, Keywords, C++ identifiers; Variables and Constants: Integer, character and symbolic constants; Dynamic initialization of variables,		
Reference variables, Basic data types in C++, Streams in C++.		
UNIT-II	**Operators and Manipulators, Decision and Control Structures and Function**	**8 hours**
Operators and Manipulators: Operators, Types of operators in C++, Precedence and associativity of operators, Manipulators. Decision and Control Structures: if statement, if-else statement, switch statement, Loop: while, do-while, for; Jump statements: break, continue, go to. Function: main() function, components of function: prototype, function call, definition, parameter;		
passing arguments; types of function, inline function, function overloading.		
UNIT-III	**OOP's Concept**	**8 hours**
Introduction to Classes and Objects: Classes in C++, class declaration, declaring objects, Defining Member functions, Inline member function, Array of objects, Objects as function argument, Static data member and member function, Friend function and friend class. Constructors and Destructors: Constructors, Instantiation of objects, Default constructor, Parameterized constructor, Copy constructor and its use, Destructors. Operator Overloading: Operator keyword, arguments and return value; overloading unary and binary operators: arithmetic operators, manipulation of strings using operators; Type conversions.		
Inheritance Derived class and base class: Defining a derived class, Accessing the base class member, Inheritance: multilevel, multiple, hierarchical, hybrid; Virtual base class, Abstract class.		
UNIT-IV	**Virtual Functions and Polymorphism and File Handling**	**8 hours**

Virtual Functions and Polymorphism: Virtual functions, pure virtual functions; Polymorphism, Categorization of polymorphism techniques: Compile time polymorphism, Run time polymorphism.		
File Handling File: File classes, Opening and Closing a file, File modes, Manipulation of file pointers, Functions for I/O operations.		
UNIT-V	**Templates and The Standard Template**	**8 hours**
Introduction to STL: Overview of the STL Components of the STL (containers, iterators, algorithms) Advantages of using the STL Basic usage examples		
Containers: Vector, List, Dequeue, Set, Map, Stack and Queue.		
Iterators: Introduction to iterators Types of iterators (input, output, forward, bidirectional, random access) Iterator operations and Iterator adapters.		
Advanced Containers: Array, Forward List, Unordered Set, Unordered Map and Priority Queue		
Advanced Topics: Function and function objects, Lambda expressions, Smart pointers and their relationship with STL containers and Exception safety and strong exception guarantee in STL.		

Course Outcome (CO)		**Knowledge Level**
At the end of course, the student will be able to:		
CO1	Able to understand and apply various object-oriented features.	K2
CO2	Implement C++ application program using basic concept of C++ and proper program structuring.	K3
CO3	Implement C++ application program using OPPs Principle and proper program structuring.	K3
CO4	Understand the concepts Virtual Functions and Polymorphism and File Handling.	K3
CO5	Boost your C++ skills with our guide on the Standard Template Library (STL). Explore advanced concepts and elevate your coding.	K4

TEXT BOOKS/ REFERENCE BOOKS:
Primary Text: "C++ Primer" by Stanley B. Lippman, Josée Lajoie, and Barbara E. Moo
Supplementary Text: "The C++ Programming Language" by Bjarne Stroustrup
SUGGESTED VIDEO LINK:

1. https://youtu.be/mlIUKyZIUUU?si=TmgMUjViwFSomxvn
2. https://youtu.be/s0g4ty29Xgg?si=Lo6odf16ahFN1GTt
3. https://youtu.be/imNlJohlLPk?si=MLrDK6DQePU1IUiZ
4. https://youtu.be/yGB9jhsEsr8?si=ql_2CZNtwBof0dK1
5. https://youtu.be/i3a-G6Ebh9E?si=0YtZFIddw0T629LW
6. https://youtu.be/AY96XFqb934?si=4dgyh1X4r1iNXuaQ
7. https://youtu.be/83KtncYgHYQ?si=evFOJvRXXwamNNul
8. https://youtu.be/mlIUKyZIUUU?si=byzTF6J91KkrDH1m
9. https://youtu.be/nGJTWaaFdjc?si=PobrmW5H4JyJFnq9
10. https://youtu.be/oRBK0Mh_gG0?si=5ZqNMgt0k2YoAV6e
11. https://youtu.be/BO2KagRMS3M?si=Rv9e1hgpXdZmZWMB
12. https://youtu.be/RO1ZYW9NAzg?si=0UHaB_FPM4Xlrq_A
13. https://youtu.be/fB3JHNnlRfI?si=QnX6RKmGRrxQ7tIp
14. https://youtu.be/F5PbnZkHqmg?si=P4PnltEirlHmprbQ
15. https://youtu.be/LS1zjr1wog4?si=exhamTfVYkj4r5di
16. https://youtu.be/c9iREsYpayk?si=tYzK3fPjZ6D77O-Z
17. https://youtu.be/m0gnToak2-g?si=rYCiDFxWC48z0J6t
18. https://youtu.be/SgcHcbQ0RCQ?si=9MH_0XyUEnqiHjs5
19. https://youtu.be/okLflHtlCuk?si=Deqwt0mp_dDqS_AW

B.TECH. FIRST YEAR (SEMESTER-II)					
Lab Code	CS12030	L	T	P	Credit
Lab Title	BASICS OF PYTHON PROGRAMMING LAB	0	0	2	2

Course Outcome (CO)	
At the end of course, the student will be able to:	
CO 1	Write simple python programs.	K2, K3
CO 2	Implement python programs using decision control statements	K3, K5
CO 3	Writing python programs using user defined functions and modules	K2
CO 4	Implement programs using python data structures –lists, tuples, set, dictionaries	K3
CO 5	Write programs to perform input/output operations on files	K3, K4

List of Programs/ Experiment:

S.N.	Program/Experiment Title	CO
1	Python Program to print "Hello Python"	CO1
2	Python Program to read and print values of variables of different data types.	CO1
3	Python Program to perform arithmetic operations on two integer numbers	CO1
4	Python Program to Swap two numbers	CO1
5	Python Program to convert degree Fahrenheit into degree Celsius	CO1
6	Python Program to demonstrate the use of relational operators.	CO2
7	Python Program to understand the working of bitwise and logical operators.	CO2
8	Python Program to calculate roots of a quadratic equation.	CO2
9	Python Program to check whether a year is leap year or not.	CO2
10	Python Program to find smallest number among three numbers.	CO2
11	Write a Python program that takes the radius of a circle as input and computes the area of the circle.	CO3
12	Write a Python program to compute the sum of digits in a given integer.	CO3
13	Write a Python program to generate the Fibonacci sequence up to n terms using a user-defined function.	CO3
14	Write a Python program to check if a given string is a palindrome using a user-defined function.	CO3
15	Write a Python program to calculate the sum of elements in a list using a user-defined function.	CO3
16	Write a Python program that finds common elements between two lists.	CO4
17	Write a Python program to perform operations on tuples, such as concatenation and repetition.	CO4
18	Write a Python program to perform set operations like union, intersection, and difference.	CO4
19	Write a Python program to perform operations on dictionaries, such as adding elements and iterating through keys and values.	CO4
20	Write a Python program to merge two lists and sort them in ascending order.	CO4
21	Write a Python program to write data to a text file.	CO5
22	Write a Python program to append new data to an existing text file.	CO5

23	Write a Python program that reads a text file and counts the number of words in it.	CO5
24	Write a Python program to copy the contents of one text file to another.	CO5
25	Write a Python program that reads a text file and counts the number of words in every sentence.	CO5

SUGGESTED VIDEO LINK:
1. https://www.youtube.com/watch?v=kcL5WWJjmIU
2. https://www.youtube.com/watch?v=RgK39AWivGM
3. https://www.youtube.com/watch?v=q1IMxGvpibc

Note: Faculty may delete/add practical according to needs.

B.TECH. FIRST YEAR (SEMESTER-II)					
Lab Code	**CS12040**	**L**	**T**	**P**	**Credit**
Lab Title	**OOPS WITH C++ LAB**	0	0	2	2

Course Outcome (CO)		**Knowledge Level**
At the end of course, the student will be able to:		
CO1	Able to understand and apply various object-oriented features.	**K2**
CO2	Construct logical expressions to implement sophisticated decision-making logic in their programs.	**K5**
CO3	Its analysis covers a range of cognitive levels from remembering the syntax of classes to creating a structured program that applies OOP principles effectively.	**K4**
CO4	They will assess the design choices involving inheritance and polymorphism for maintaining code readability and reusability.	**K3**
CO5	Boost your C++ skills with our guide on the Standard Template Library (STL). Explore advanced concepts and elevate your coding.	**K4**

List of Programs/ Experiment:

1	C++ Program to Demonstrate the correct size of various data types on your computer.	CO1
2	Write a C++ program that calculates the area of a rectangle given its length and width. Use variables for length and width, and define a symbolic constant for the value of Pi. The program should dynamically initialize the variables for length and width based on user input, and then compute and output the area of the rectangle.	CO1
3	C++ Program to Swap Two Numbers	CO1
4	C++ Program to find ASCII Value of a Character	CO1
5	C++ Program to Find Size of int float double and char in your system	CO1
6	C++ Program to demonstrate the Bitwise Operators, Logical Operators and Conditional Operators.	CO2
7	C++ Program to check whether an integer is positive or negative	CO2
8	C++ Program to implement prefix and postfix.	CO2
9	C++ Program to check leap year using if-else	CO2
10	C++ Program to print whether a character is vowel or not	CO2
11	C++ Program to demonstrate implementation of Inheritance	CO3
12	CPP Program to demonstrate parameterized constructors	CO3
13	C++ Program to demonstrate implementation of Inheritance	CO3
14	C++ Program to demonstrate parameterized constructors	CO3
15	C++ program to overload the binary operator + This program adds two complex numbers	CO3
16	Writing Data to the Text File in a Program	CO4
17	Write a C++ program to count the number of lines in a text file.	CO4
18	Write a C++ program to copy the contents of one text file to another.	CO4
19	C++ program to demonstrate the use of virtual function	CO4
20	Write C++ Program to implement Abstract Class	CO4
21	C++ program to implement map container.	CO5
22	C++ Program to Implement LexicoGraphical_Compare() in STL	CO5
23	C++ Program to Implement List in STL	CO5

24	C++ Program to Implement List in STL	CO5
25	Write a program C++ STL Sorting Algorithm	CO5

SUGGESTED VIDEO LINK:

1. https://www.youtube.com/playlist?list=PLBlnK6fEyqRhqQV_MzlT8xsPQnsGcMdIo
2. https://youtu.be/GdR7BVrXTOk?feature=shared
3. https://youtu.be/XOSblnOEHKY?feature=shared
4. https://youtu.be/7ZfEp71tIec?feature=shared
5. https://youtu.be/prl3iiwEER4?feature=shared
6. https://youtu.be/eGfjbjai9Qg?feature=shared
7. https://youtu.be/KwS-Vbsha1k?feature=shared
8. https://youtu.be/MeVAqqodoEs?feature=shared
9. https://youtu.be/k5js3EH4oe8?feature=shared
10. https://youtu.be/vs9d_bcbN6g?feature=shared

Note: Faculty may delete/add practical according to needs.