सुदूरपश्चिम विश्वविद्यालय

मानविकी तथा सामाजिक शास्त्र सङ्काय

स्नातक अनिवार्य नेपाली

पाठ्यांश शीर्षकः वर्णविन्यास, व्याकरण र लेखनकला/ लेखनसिप

विषय संकेत नं. BCAN 111 | पूर्णाङ्क :
विषयको प्रकृति : सैद्धान्तिक | उत्तीर्णाङ्क :
तहः स्नातक | क्रेडिट आवरः ३
सत्र : पहिलो | जम्मा पाठघण्टा : ४८

१. पाठ्यांश परिचय

प्रस्तुत पाठ्यांश सुदूरपश्चिम विश्वविद्यालयअन्तर्गत स्नातक तहमा बी.सी.ए. कार्यक्रममा अध्ययनरत विद्यार्थीहरूमा नेपाली भाषाको मानक वर्णविन्यासको पहिचान र प्रयोग क्षमता विकास गर्न, नेपाली भाषाको व्याकरणसम्बन्धी आधारभूत ज्ञान अभिवृद्धि गरी त्यसका आधारमा लेखनसिपको विकास गर्न तयार पारिएको हो । यसमा नेपाली भाषाको लेख्य वर्णविन्यास, शब्द भण्डार, वाक्यतत्त्वपरक रचना, वाक्यान्तरण र वाक्यसंश्लेषण, पठनबोध, बुँदाटिपोट र सङ्क्षेपीकरण, व्यावहारिक लेखन, अनुच्छेद र प्रतिवेदनलेखन तथा सन्दर्भपूर्ण सूचनाको रूपान्तर क्षमताको विकास गर्ने पाठ्यसामग्रीहरू समावेश गरिएका छन् ।

२. उद्देश्य

प्रस्तुत पाठ्यांशको अध्ययनपश्चात् विद्यार्थीहरू निम्नलिखित कुरामा सक्षम हुनेछन्:

१. लेख्य नेपालीको स्वरूप पहिल्याई मानक वर्णविन्यासको प्रयोग गर्न
२. नेपाली भाषाका शब्दको स्रोत, वर्ग र निर्माण प्रक्रिया पहिल्याउन
३. वाक्यतत्त्वको पहिचान गरी वाक्यको रचना गर्न, तिनको वाक्यान्तरण गर्न र विभिन्न वाक्यमा व्यक्त छोटो सङ्कथनलाई एकल वाक्यमा संश्लेषण गर्न
४. नेपाली वाङ्मयका विविध क्षेत्रकागद्यांशहरु पढी तिनमा आधारित बोधप्रश्नहरूको उत्तर दिनका साथै सम्बद्ध गद्यांशको बुँदाटिपोट र सङ्क्षेपीकरण गर्न,
५. विभिन्न प्रयोजनका लागि निर्धारित ढाँचामा आधारित व्यावहारिक लेखन गर्न
६. विभिन्न शैलीमा अनुच्छेद रचना गर्न र विभिन्न प्रयोजनका प्रतिवेदन तयार गर्न,
७. तालिका, चित्राकृति (डायग्राम) रेखाचित्र (ग्राफ) र आरेखको सूचनालाई अनुच्छेदमा रूपान्तर गर्न तथा अनुच्छेदमा रहेका सूचनालाई तालिका, चित्राकृति आलेख र आरेखमा रूपान्तर गर्न ।

३. एकाइ विभाजन

एकाइ १. वर्णविन्यास
एकाइ २. शब्दभण्डार
एकाइ ३. वाक्यतत्त्वपरक रचना र वाक्यान्तरण र वाक्यसंश्लेषण
एकाइ ४. पठनबोध, बुँदाटिपोट, सङ्क्षेपीकरण

एकाइ ५. व्यावहारिक लेखन
एकाइ ६. अनुच्छेद र प्रतिवेदन-लेखन
एकाइ ७. सन्दर्भपूर्ण सूचनाको रूपान्तर

४. एकाइगत पाठ्यवस्तु विवरण

एकाइ १. नेपाली वर्णविन्यास **पाठघण्टा ५**

१.१. नेपाली शब्दहरूको वर्णविन्यास (ह्रस्वदीर्घ, श ष स, ब व, य ए, ज्ञ ग्यँ, क्ष छे, पदयोग, पदवियोग, चन्द्रखिन्दु, शिरविन्दु र पञ्चमर वर्ण, हलन्त र अजन्त तथा लेख्यचिह्नको प्रयोग र अभ्यास)

एकाइ २. नेपाली शब्दभण्डार **पाठघण्टा ८**

२.१. शब्दस्रोतगत वर्गीकरण .(तत्सम, तद्भव र आगन्तुक)
२.२. शब्दवर्ग नाम, सर्वनाम, विशेषण, क्रिया, नामयोगी, क्रियायोगी, संयोजक, निपात र विस्मयादिबोधक
२.३. शब्दनिर्माण प्रक्रिया: उपसर्ग, प्रत्यय, समास र द्वित्वद्वारा शब्दनिर्माण

एकाइ ३. वाक्यतत्त्वपरक रचना, वाक्यान्तरण र वाक्यसंश्लेषण **पाठघण्टा १२**

३.१ वाक्यतत्त्व: उद्देश्य र विधेय (उद्देश्यविस्तार तथा विधेयविस्तारसहित)
३.२ वाक्यतत्त्वपरक रचना: लिङ्ग, वचन, पुरुष, आदर, काल, पक्ष, भाव, वाच्य तथा करण अकरणका आधारमा सङ्गतिपूर्ण वाक्यरचनाको अभ्यास
३.३ वाक्यान्तरण: लिङ्ग, वचन, पुरुष, आदर, काल, पक्ष, भाव, वाच्य तथा कथन (प्रत्यक्ष-अप्रत्यक्ष)सँग सम्बद्ध वाक्यान्तरणको अभ्यास
३.४ विभिन्न वाक्यमा व्यक्त छोटो सङ्कथनलाई एकल वाक्यमा संश्लेषणको अभ्यास

एकाइ ४. पठनबोध, बुँदाटिपोट र सङ्क्षेपीकरण **पाठघण्टा ७**

४.१नेपाली वाङ्मयका विभिन्न क्षेत्रका दृष्टांश तथा अदृष्टांश पाठ्यांशको पठनबोध र बोधात्मक प्रश्नोत्तरको अभ्यास
४.२. निर्धारित अनुच्छेदबाट बुँदाटिपोटको अभ्यास
४.३. निर्धारित अनुच्छेदबाट सङ्क्षेपीकरणको अभ्यास

एकाइ ५. व्यावहारिक लेखन **पाठघण्टा ७**

५.१. कार्यालयीय तथा व्यापारिक चिठी, निवेदन तथा सम्पादकलाई चिठी लेखनको अभ्यास
५.२. विज्ञापन, शुभकामना, निमन्त्रणापत्र, बधाई तथा श्रद्धाञ्जलि लेखनको अभ्यास
५.३. व्यक्तिवृत्त(बायोडाटा) लेखनको अभ्यास

एकाइ ६. अनुच्छेद र प्रतिवेदन-लेखन **पाठघण्टा ४**

६.१. विभिन्न शैलीका संसक्तिमूलक अनुच्छेद रचनाको अभ्यास
६.२. विभिन्न घटना, समारोह, भ्रमण, निरीक्षणमा आधारित प्रतिवेदन लेखनको अभ्यास

एकाइ ७. सन्दर्भपूर्ण सूचनाको रूपान्तर **पाठघण्टा ५**

७.१. अनुच्छेदमा अभिव्यक्त सूचनालाई तालिकामा रूपान्तर र तालिकालाई अनुच्छेदमा रूपान्तर

७.२. अनुच्छेदमा अभिव्यक्त सूचनालाई चित्राकृति (वृत्तचित्र र स्तम्भचित्र)मा रूपान्तर र चित्राकृति (वृत्तचित्र र स्तम्भचित्र)लाई अनुच्छेदमा रूपान्तर

७.३. अनुच्छेदमा अभिव्यक्त सूचनालाई रेखाचित्रमा रूपान्तर र रेखाचित्रलाई अनुच्छेदमा रूपान्तर

७.४. अनुच्छेदमा अभिव्यक्त सूचनालाई आरेखमा रूपान्तर र आरेखलाई अनुच्छेदमा रूपान्तर

५. सन्दर्भग्रन्थ

अधिकारी, हेमाङ्ग राज (२०६७). *प्रयोगात्मक नेपाली व्याकरण.* साझा प्रकाशन ।

आचार्य, व्रतराज र गौतम, देवी प्रसाद (२०६१). *विशेष नेपाली : प्रयोजनपरक बोध र लेखन.* विद्यार्थी पुस्तक भण्डार ।

भट्टराई, रमेशप्रसाद र चौलागाईं, प्रेमप्रसाद (२०७७). *प्रज्ञा शैक्षणिक व्याकरण.* नेपाल प्रज्ञा प्रतिष्ठान ।

शर्मा, मेहनराज (२०७५). *नेपाली सन्दर्भपरक व्याकरण.* नेपाल प्रज्ञा प्रतिष्ठान ।

सुवेदी, लालानाथ (२०६१). *नेपाली बोध र रचनाकौशल.* हिमालय बुक स्टल ।

Far Western University
Bachelor in Computer Application (BCA)

Course Title: Basic Mathematic I
Course No.: BCAM112
Level: BCA
Semester: I

Credit Hour: 3
Nature of course: Theory
Total periods: 48

1. Course Introduction

This course is designed for the students of the BCA program as a foundational course to equip students with essential mathematical concepts and skills that are directly applicable to computing and problem-solving in computer science. The course integrates logical reasoning, algebraic structures, combinatorics, graph theory, and optimization techniques to build a robust analytical framework for future programming, algorithm design, data analysis, and system modeling. Beginning with propositional and predicate logic, students learn to construct and analyze logical statements critical for algorithmic thinking and formal proofs. The study of sets, relations, and functions introduces fundamental structures used in databases, programming, and discrete modeling. Boolean algebra and logic gates bridge mathematics and digital circuit design, reinforcing concepts essential for understanding computer architecture.

The course advances into linear algebra, where students perform matrix operations, solve systems of equations, and explore eigenvalues and eigenvectors—skills that are crucial in graphics, simulations, and data transformations. Combinatorics and recurrence relations further develop problem-solving abilities used in algorithm efficiency and dynamic programming. Graph theory introduces networks, connectivity, and traversal algorithms like DFS and BFS, laying the groundwork for applications in AI, social networks, and cybersecurity. This course introduces optimization techniques, including linear programming and gradient descent, which are key tools in decision-making, operations research, and machine learning.

Overall, this course provides students with a balanced foundation of theoretical knowledge and applied problem-solving skills, ensuring they are prepared for more advanced courses in computing and data science

2. Course Objectives

By the end of the course, the students are expected to achieve the following objectives:

- To apply the principles of propositional and predicate logic to analyze logical statements and construct valid arguments in computing contexts.
- **To perform** operations on sets, relations, and functions and distinguish among injective, surjective, and bijective mappings in mathematical models.
- To construct and evaluate truth tables and logic gate diagrams using Boolean algebra in relation to digital systems.
- **To execute** matrix operations including addition, multiplication, transpose, and compute determinants, inverses, and rank in solving linear algebra problems.
- **To solve** systems of linear equations using Gaussian elimination and matrix inversion techniques and interpret solutions involving eigenvalues and eigenvectors (2x2 cases).
- **To apply** the concepts of permutations, combinations, and recurrence relations in solving problems in combinatorics and algorithm design.

- **To analyze** the structure of graphs by identifying types, degrees, paths, and cycles, and represent them using adjacency matrices.
- **To implement** graph traversal algorithms such as DFS and BFS for problem-solving in computer science.
- **To demonstrate** understanding of vectors and their types and compute dot product, cross product, and vector triple product in geometric contexts.
- **To evaluate** vector spaces and subspaces, and determine basis, dimension, and orthogonality in abstract vector systems.
- **To formulate** and solve linear programming problems using graphical and simplex methods for optimal decision-making.
- **To interpret** the gradient descent algorithm conceptually and explain its role in optimization problems and machine learning applications.

3. Course Contents and Specific Objectives

Specific Objectives	Content
• To discuss and apply propositional and predicate logic. • To perform set operations and understand set theory. • To describe relations and functions with real-life computing examples. • To use Boolean algebra and logic gates in digital logic design.	**Unit 1: Logic and Discrete Structures (10 Hours)** 1.1 Propositional & Predicate Logic 1.2 Sets 1.3 Order pairs and Relations, 1.4 Functions (Injective/Surjective/Bijective) 1.5 Basics of Boolean Algebra (Logic Gates, Truth Tables)
• To perform matrix operations. • To apply concepts of determinants, inverses, and rank. • To solve systems of linear equations. • To comprehend the basics of eigenvalues and eigenvectors, and their importance in computer science.	**Unit II: Linear Algebra for Computer (10 Hours)** 2.1 Matrices introduction 2.2 Operations of matrix (Addition, Multiplication, Transpose) 2.3 Determinants 2.4 Inverses 2.5 Rank 2.6 Solving Linear Systems (Gaussian Elimination, Matrix Inversion) 2.7 Eigenvalues & Eigenvectors (2x2 cases, intuitive applications)
• To apply counting principles using permutations and combinations. • To solve basic combinatorial problems. • To comprehend and formulate recurrence relations. • To solve simple recurrence relations and explore their use in	**Unit III: Combinatorics and Recursion (6 hrs)** 3.1 Introduction 3.2 Permutations 3.3 Combinations

programming and algorithms.	3.4 Recurrence relations and applications
• To classify different types of graphs. • To represent graphs using adjacency matrices and lists. • To analyze paths, degrees, and cycles in graphs. • To implement and trace graph traversal algorithms like DFS and BFS. • To apply graph theory concepts in real-world computer applications.	**Unit IV: Graph Theory for Computer Applications (8 hrs)** 4.1 Introduction 4.2 Types of graphs 4.3 Degree, Paths, Cycles 4.4 Adjacency matrix 4.5 Graph Traversal (DFS/BFS)
• To define vectors and explain their geometric and algebraic representations. • To describe various types of vectors such as zero vector, unit vector, position vector, and co-initial vectors. • To represent vectors in two- and three-dimensional space with appropriate notation. • To compute the **dot product** of two vectors and interpret it in terms of projection and angle between vectors. • To compute the **cross** of vectors in 3D space and interpret it geometrically (perpendicular vector, area of parallelogram). • To define and evaluate the **scalar triple product** and **vector triple product** of vectors. • To define a **vector space and subspace** and verify whether a **product** given set forms a vector space. • To determine the **basis** and **dimension** of a vector space or subspace. • To check for **linear independence** and the span of a set of vectors.	**Unit V: Vector and Vector Space (8 Hours)** 5.1 Introduction 5.2 Types of vectors 5.3 Dot product and cross product 5.4 Vector triple product 5.5 Vector Spaces & Subspaces 5.6 Basis, Dimension, Orthogonality
• To describe the basics of linear programming. • To apply and solve graphical and simplex methods to solve	**Unit VI: Optimization (6 Hours)** 6.1 Linear Programming Introduction

optimization problems. • Grasp the concept of gradient descent and its role in optimization. • To recognize applications of optimization techniques in machine learning.	6.2 Graphical Method 6.3 Simplex method 6.4 Gradient Descent (Conceptual, no proofs) 6.5 Applications in Machine Learning

4. Methodology and Techniques

- Core concepts and principles will be delivered through interactive lectures using whiteboard explanations, slides, and real-life analogies relevant to computing and technology.
- Regular classwork and group activities will be organized to solve mathematical problems, especially from linear algebra, logic, and graph theory, encouraging analytical thinking.
- Use of graphing tools, matrix operations via calculators or software (e.g., Excel, MATLAB, Python), and visual aids for vector geometry and graph traversal to enhance conceptual understanding.
- Students will work in pairs or groups for selected topics (e.g., logic circuits, recursion problems, graph theory), promoting peer learning and communication.
- Numerical and matrix-based problems will include implementation using spreadsheets or programming to connect mathematics with computing applications.
- Emphasis will be placed on discussing alternative problem-solving strategies, encouraging students to ask questions and articulate their reasoning.
- Periodic assignments and small project tasks (e.g., constructing truth tables, solving real-world optimization problems) will help reinforce concepts practically.
- Frequent short reviews, quizzes, and feedback sessions will help assess progress and clarify misunderstandings before proceeding to new units.
- Where possible, online visualizations, videos, or simulation tools will be used to illustrate abstract mathematical ideas such as graph traversals or matrix transformations.

5. Evaluation Scheme

5.1 **Internal Evaluation (40%)**

Internal Evaluation will be conducted by the course teacher based on the following activities.

a) **Problem solving and Class presentation: 5*5 = 25 marks**

(problem solving on 5 questions of project type after certain units, followed by presentation and viva)

b) **Lab work: 5 marks**

(Perform practical exercises using tools like Python, Excel, or Logisim to implement logic gates, matrix operations, graph traversal algorithms, set operations, and basic optimization techniques, demonstrating the application of mathematical concepts in computing)

c) **Mid-term exam: 10 marks**

5.2 End Semester Examination Model (60%)

Nature of question	Total questions to be asked	Total questions to be answered	Total marks

Group A: Short answer type question			
Group B: Long answer type question/case studies			
Total			**60**

Recommended Textbooks/Resource Books

Deo, N. (2017). Discrete Mathematics and Graph Theory. Prentice-Hall India.

Lipschutz, S., & Lipson, M. (2018). Schaum's Outline of Linear Algebra (5th ed.). McGraw-Hill Education.

Lay, D. C., Lay, S. R., & McDonald, J. J. (2016). Linear Algebra and Its Applications (5th ed.). Pearson.

Rosen, K. H. (2019). Discrete Mathematics and Its Applications (8th ed.). McGraw-Hill Education.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). Introduction to Algorithms (4th ed.). MIT Press.

Taha, H. A. (2017). Operations Research: An Introduction (10th ed.). Pearson.

Kolman, B., Busby, R. C., & Ross, S. (2018). Discrete Mathematical Structures (6th ed.). Pearson.

Paudel, B. P. (2015). Discrete Mathematics for Computer Science. Kathmandu: Educational Publishing House.

Shrestha, S. (2016). Linear Algebra and Its Applications. Kathmandu: Buddha Academic Publishers and Distributors.

K.C., R. (2014). Mathematics for Computer Applications. Kathmandu: Ratna Pustak Bhandar.

Far Western University

Faculty of Humanities and Social Sciences

Course Title**: Introduction to Information Technology**

Course Code: **BCAC113** Nature of course: Theoretical + Practical

Level: Bachelor Credit hours: 3

Semester: First Teaching hours: 64 (32T + 32P)

1. Course Introduction

This course aims to provide the students with key areas of modern Information Technology, including essential software, applications, and digital skills for the IT field. It covers basic IT concepts, operating systems, and office tools such as word processors, spreadsheets, databases, and presentations. The course also explores telecommunications, computer networks, the internet, email, web technologies, AI and ethical considerations in IT.

2. Course General Objectives

Following are the general objective of this course:

- To familiarize the students with digital skills include computer fundaments.
- To use email, internet, and social media to communicate and collaborate with others.
- To create and use multimedia and digital resources to make learning more interesting and accessible.
- To explore AI tools to support their learning.
- To learn about cybersecurity and apply safe practices to protect their digital environment.

3. Specific Objectives and Contents

Specific Objectives	Contents
• Describe digital devices and computers system. • Know how computers have evolved over time. • Identify different types of computers and their uses. • Explain how computers process data using input and output. • Recognize key hardware parts like CPU,	**Unit 1 : Fundamentals of Computer (6T+4P)** 1.1 Introduction to Digital Devices and Computers 1.2 History and Evolution of Computers 1.3 Types of Computers (Desktop, Laptop, Tablet, Mobile) 1.4 Computer System: Input, Process and Output 1.5 Hardware Components (CPU, RAM, Storage) 1.6 Peripheral Devices and Their Functions 1.7 Software Types: System vs Application 1.8 Operating Systems Overview (Windows, macOS, Linux) 1.9 File Management and Storage Devices **Lab Work**

RAM, and storage. • List common peripheral devices and their functions. • Tell the difference between system and application software. • Describe the role of operating systems and name popular ones. • Organize files and understand types of storage devices.	• Create and organize folders and files on the desktop or mobile. • Customize desktop settings like wallpaper, theme, and taskbar. • Use system tools like Task Manager and Control Panel to view system info and adjust settings.
• Understand what the internet is and how it works. • Use web browsers and search engines. • Create and manage an email account. • Write emails using proper etiquette and clear communication. • Identify and use popular social media platforms. • Use online collaboration tools like Google Workspace and Microsoft 365. • Practice respectful and safe behavior online. • Store and share files using cloud computing services.	**Unit 2 : Internet and Social Media (5T+5P)** 2.1 Internet and how it works 2.2 Web Browsers and Search Engines 2.3 Creating and Managing Email Accounts 2.4 Email Etiquette and Communication 2.5 Social Media Platforms 2.6 Online Collaboration Tools (Google Workspace, Microsoft 365) 2.7 Netiquette and Responsible Online Behavior 2.8 Cloud Computing and File Sharing **Lab work** • Create and send an email with a subject, message, and attachment to your circle. • Search for information using Google or Bing and bookmark a useful website. • Join a Google Meet or Microsoft Teams session and share a document during the meeting. • Post a short message on a social media platform (or a mock platform) following netiquette rules. • Upload and share a file using Google Drive or OneDrive with proper access settings.
• Identify the different application packages • Demonstrate word processing	**Unit 3: Computer Applications and Office Packages (2T+8P)** **3.1** Introduction to Office Packages: An Overview of Office

skills, including creating and editing complex documents, using consistent formatting styles, and developing collaborative skills like tracking changes and sharing with peers. • Demonstrate to create and format spreadsheets, manipulate data, apply basic formulas and functions, and interpret data through visualization techniques, creating meaningful charts and graphs. • Create engaging presentations using slide transitions and animations and practice presentation techniques to enhance classroom learning experiences.	Packages (Microsoft Office, Google Office Workspace, LibreOffice), Installation and Setup or Subscription process. **3.2 Word Processor**: Create and edit documents, formatting documents, use styles, headers and footers, set page numbers, create section breaks, collaborating with documents (track changes, leave comments and share). **3.3 Spreadsheet**: Create and format spreadsheets, entering and editing data, basic Formulas and functions, data visualization (charts and graphs). **Lab work** • Perform with word processor software and prepare report as per teacher instruction using latest word processing tools. • Perform with spreadsheet application and work on data analysis and create report as per teacher instruction using latest spread sheet tools.
• Define the basic elements of multimedia like text, images, audio, and video. • Create simple presentations using presentation tools with interactive features. • Edit basic images using tools • Record and edit audio using basic audio tools. • Create and edit short videos using beginner-friendly software. • Use digital storytelling tools to share ideas creatively. • Add captions and alt text to make	**UNIT 4: Multimedia and Interaction (4T+8P)** 4.1 Introduction to Multimedia Elements 4.2 Creating Presentations and Interactive features 4.3 Basic Image Editing and Practices 4.4 Audio Recording and Editing Tools 4.5 Video Creation and Editing Basics 4.6 Using Digital Storytelling Tools 4.7 Accessibility in Multimedia (Captions, Alt Text) 4.8 Integrating Multimedia Social activities **Lab work** • Create a presentation using PowerPoint or Google Slides with text, images, and transitions. • Edit an image using Canva or Online tools by adding text and adjusting colors.

multimedia accessible. • Combine multimedia elements to social activities	• Record a short audio clip and edit it using a basic audio tool like Audacity. • Create a short video combining images, audio, and text using a simple video editor like CapCut. • Add captions and alt text to a multimedia project to improve accessibility.
• Define AI and describe how it works. • Identify common uses of AI in daily life. • Explore how chatbots and virtual assistants function. • Apply AI tools for writing and research. • Develop prompts and analyze generative AI outputs. • Create images or music using AI tools. • Evaluate ethical issues and recognize bias in AI.	**Unit 5 Artificial Intelligence tools (5T+5P)** 5.1 Artificial Intelligence and 5.2 Everyday Uses of AI 5.3 Introduction to Chatbots and Virtual Assistants 5.4 Using AI for Writing and Research 5.5 Prompt Engineering and Generative AI 5.6 AI Tools for Creativity (Image Generation, Music Creation) 5.7 Ethical Use of AI and Bias Awareness **Lab work** • Demonstrate to interact with an AI chatbot to ask questions and summarize answers. • Compose a paragraph and revise it using an AI writing assistant. • Design prompts and analyze how AI responds differently. • Generate creative content like images or music using AI tools. • Compare AI outputs and evaluate them for bias or ethical concerns.
• Describe cybercrime and important security • Identify common online threats • Practice safe browsing and protect personal data. • Use strong passwords and secure login methods.	**Unit 6: Cyber Security and Social behaviour (8T+4P)** 6.1 Concept of Cybercrime and information security 6.2 Common Threats: Viruses, Phishing, Malware 6.3 Safe Browsing Practices 6.4 Password Management and Authentication 6.5 Privacy Settings on Social Media

• Adjust privacy settings to stay safe on social media. • Recognize how online actions affect your digital reputation. • Report cyber incidents and respond appropriately. • Learn about cyber laws and follow responsible online behavior. • Demonstrate good digital citizenship in everyday online activities.	6.6 Digital Footprint and Online Reputation 6.7 Reporting and Responding to Cyber Incidents 6.8 Cyber Laws and Responsible Use 6.9 Digital Citizenship **Lab work** • Scan a device for viruses using antivirus software and review the results. • Create a strong password and test it using a password strength checker. • Adjust privacy settings on a social media account to limit public access. • Identify phishing emails from a sample set and explain why they are suspicious. • Review your digital footprint by searching your name online and reflecting on the results.

4. **Instructional Techniques**

Lecture preferably with the use of multi-media projector, demonstration, practical classes, discussion, and brain storming. The class can be made better with following techniques:

- Interactive Lectures with Multimedia
- Hands-On Practice and Lab Activities
- Visual Aids and Simulations
- Collaborative Learning Strategies
- Project-Based and Creative Challenges
- Formative Assessment and Scaffolding

5. Evaluation

Evaluation of students' performance is divided into parts: Internal assessment, practical examination and theoretical examinations. The distribution of points is given below:

Internal Assessment	Practical Exam/Viva	Semester Examination (Theoretical exam)	Total Points
40 Points	20 Points	40 Points	100 Points

***Note**: Students must pass separately in internal assessment, external practical exam and semester examination.*

5.1 Internal Assessment (40 Points)

Internal assessment will be conducted by subject teacher based on following criteria:

Attendance	
Internal Exam	
Readiness for learning	
Lab Work / Project work / Presentation	
Total	**40 points**

5.2 Semester Examination (40 Points)

Examination Controller's Office will conduct final examination at the end of semester.	
Short answer questions Long answer questions	
Total	**40 points**

5.3 Practical Examination (20 Points)

Internal Evaluation	External Evaluation			Total
	Lab Report	Lab Exam	VIVA	
5 points	5 points	5 points	5 points	20 points

6. Recommended Books and References Materials

Sinha, P. K., & Sinha, P. (2021). Computer Fundamentals: Concepts, Systems & Applications, 8th Edition (8th Edition). BPB Publications.

Alexis Leon & Mathews Leon (2009). Fundamentals of information technology, 2/e. New Delhi.Vikas Publishing House

Bowen, J. A., & Watson, C. E. (2024). *Teaching with AI: A practical guide to a new era of human learning*. Johns Hopkins University Press.

Khan, S. (2024). *Brave new words: How AI will revolutionize education (and why that's a good thing)*. Viking.

Cox, J., Lambert, J., & Frye, C. (2011). *Microsoft Office Professional 2010 step by step*. Redmond, Wash: Microsoft.

Melton, B. (Ed.). (2013). *Microsoft Office Professional 2013*. Sebastopol, Calif: O'Reilly Media.

Joan Lambert, Curtis Frye, (2013). Microsoft Office Step by Step (Office 2021 and Microsoft 365), Microsoft Press

Melton, Beth, Dodge, Mark. (2013). Microsoft Office Home and Student 2013 Step By Step. India: PHI

Patrice-Anne Rutledge. (2014), Office 2013 All-In-One Absolute Beginner's Guide ISBN:9789332539372, Pearson India

Far Western University

Faculty of Humanities and Social Sciences

Course Title: **Python Programming**

Course Code: **BCAC114** Nature of course: Theoretical + Practical

Level: Bachelor Credit hours: 3

Semester: First Teaching hours: 64 (32T + 32P)

1. Course Description:

This course provides an introduction to Python programming, focusing on core concepts such as syntax, control structures, functions, data structures, exception handling, file operations, and essential libraries like NumPy, Pandas, and Matplotlib. Students will learn to write efficient, readable code and solve real-world problems through hands-on lab exercises and projects. By the end, students will be equipped with the foundational skills needed for further study or entry-level programming roles.

2. Course General Objectives:

Upon successful completion of this course, students will be able to:

- To explain Python syntax and programming fundamentals.
- To apply control flow and looping structures.
- To develop and use functions and modules.
- To work with Python data structures effectively.
- To handle exceptions and perform file operations.
- To use Python libraries for data analysis and visualization.

3. Specific Objectives and Contents

Specific Objectives	Contents
• Understand basic programming concepts and Python's role. • Install Python and set up development environments. • Write and run simple Python scripts. • Apply correct syntax and indentation. • Use variables and data types effectively.	**Unit 1: Python Basics and Environment Setup (5T+5P)** 1.1 Introduction to Programming and Python 1.2 Installing Python and IDEs (VS Code, Jupyter) 1.3 Writing and Running Python Scripts 1.4 Python Syntax and Indentation 1.5 Variables and Data Types 1.6 Basic Input and Output

• Perform basic input and output operations. • Write readable code using comments and style guides. • Use Python REPL and manage virtual environments.	1.7 Comments and Code Style 1.8 Using Python REPL and Virtual Environments **Practical Works** • Install Python and VS Code/Jupyter • Write a script that takes user input and prints a formatted message. • Create a script that demonstrates different data types and type conversions. • Use comments to document a simple program
• Understand control flow and looping concepts in Python. • Apply Boolean logic and comparison operators. • Use conditional statements to control program behavior. • Implement while loops for repeated execution. • Use for loops to iterate over sequences. • Control loop execution using break, continue, and pass. • Write and manage nested loops effectively.	**Unit 2: Control Flow and Looping (5T+5P)** 2.1 Control Flow and Looping 2.2 Boolean Logic and Comparison Operators 2.3 Conditional Statements (if, elif, else) 2.4 while Loops 2.5 for Loops 2.6 Loop Control Statements (break, continue, pass) 2.7 Nested Loops **Practical Works** • Write a program demonsrate the control statement if. • Demonstrate the if-elif-else statements. • Demonstrate the while loop. • Demonstrate the nested for loops.
• Define and call functions in Python. • Use parameters and return values effectively. • Understand variable scope and lifetime. • Apply default and keyword arguments. • Implement basic recursion. • Use lambda functions. • Handle flexible arguments with *args and **kwargs. • Organize code using modules for better structure.	**Unit 3: Functions and Modular Programming (6P+ 6P hrs.)** 3.1 Defining and Calling Functions 3.2 Function Parameters and Return Values 3.3 Variable Scope and Lifetime 3.4 Default and Keyword Arguments 3.5 Recursion Basics 3.6 Lambda Functions 3.7 *args and **kwargs 3.8 Organizing Code with Modules **Practical Works** • Demontrate a functions.

	• Write a program using *args and **kwargs to handle variable arguments. • Organize multiple functions into a module and import them into another script.
• Create and manipulate lists using built-in methods. • Work with tuples and understand their immutability. • Use sets for unique data and perform set operations. • Create and manage dictionaries for key-value data. • Apply list and dictionary comprehensions. • Handle nested data structures effectively. .	**Unit 4: Data Structures (5T+ 5P Hrs.)** 4.1 Lists and List Operations 4.2 Tuples and Tuple Operations 4.3 Sets and Set Operations 4.4 Dictionaries and Dictionary Operations 4.5 List and Dictionary Comprehensions 4.6 Nested Data Structures 4.7 Iterating and Manipulating Structures **Practical Works** • Demonstare the create a list and perform basic operations (add, remove, sort). • Write a program that demonstrate a dictionary. • Build a nested dictionary to store and retrieve records.
• Define the purpose of exception handling and file operations. • Identify common errors and exceptions in Python. • Use try, except, and finally blocks to handle exceptions. • Define custom exceptions. • Read from and write to text files using Python. • Use the with statement for safe file handling. • Work with CSV files for structured data. • Read and write JSON files for data exchange.	**Unit 5: Exception Handling and File Operations (5T+ 5P Hrs.)** 5.1 Exception Handling and File Operations 5.2 Introduction to Errors and Exceptions 5.3 try, except, finally Blocks 5.4 Raising and Custom Exceptions 5.5 Working with Text Files (open, read, write) 5.6 Using with Statement for File Handling 5.7 Reading and Writing CSV Files 5.8 Working with JSON Files **Practical Works** • Demonstrate try-except in program. • Script that reads from and writes to a text file. • Parse a CSV file and display its contents in a formatted way. • Write a program that reads a JSON file and extracts specific

	data.
• Define the basics of NumPy arrays and their structure. • Perform array operations and indexing with NumPy. • Work with Pandas Series and DataFrames for data analysis. • Clean and manipulate data using Pandas. • Apply grouping and aggregation techniques in Pandas. • Use Matplotlib for basic data visualization. • Create line, bar, and scatter plots with Matplotlib. • Combine Pandas and Matplotlib to analyze and visualize data.	**Unit 6: Python Libraries (6T+6P Hrs.)** 6.1 Introduction to NumPy Arrays 6.2 Array Operations and Indexing 6.3 Introduction to Pandas Series and DataFrames 6.4 Data Cleaning and Manipulation 6.5 Grouping and Aggregation in Pandas 6.6 Introduction to Matplotlib 6.7 Plotting with Matplotlib (Line, Bar, Scatter) 6.8 Combining Pandas and Matplotlib for Analysis **Practical Works** • Create and manipulate NumPy arrays (reshape, slice, arithmetic). • Load a CSV file into a Pandas DataFrame and perform basic analysis. • Generate a bar chart and line plot using Matplotlib.

5 Instructional Techniques

Lecture preferably with the use of multi-media projector, demonstration, practical classes, discussion, and brain storming. Python class can be made better with following techniques:

- Active Learning and Problem-Based Instruction
- Hands-On Laboratory Work
- Visual and Simulation-Based Demonstrations
- Concept Mapping and Incremental Scaffolding
- Peer Instruction and Think-Pair-Share
- Mini-Projects and Design Challenges
- Flipped Classroom Approach
- Continuous Formative Assessment

6 Evaluation

Evaluation of students' performance is divided into parts: Internal assessment, practical examination and theoretical examinations. The distribution of points is given below:

Internal Assessment	Practical Exam/Viva	Semester Examination (Theoretical exam)	Total Points

40 Points	20 Points	40 Points	100 Points

***Note**: Students must pass separately in internal assessment, external practical exam and semester examination.*

6.1 Internal Assessment (40 Points)

Internal assessment will be conducted by subject teacher based on following criteria:

Attendance

Internal Exam

Readiness for learning

Lab Work / Project work / Presentation

Total	**40 points**

6.2 Semester Examination (40 Points)

Examination Division, Dean office will conduct final examination at the end of semester.

Short answer questions

Long answer questions

Total	**40 points**

6.3 **Practical Examination (20 Points)**

Internal Evaluation	**External Evaluation**			**Total**
	Lab Report	**Lab Exam**	**VIVA**	
5 points	5 points	5 points	5 points	20 points

7 Recommended Books and References Materials

Brown, M. C. (2018). *Python: The Complete Reference*. Fourth Edition, McGraw-Hill. ISBN:9789387572942

Barry, P. (2017). *Head first Python* (Second edition). O'Reilly Media.

Matthes, E. (2019). *Python crash course: A hands-on, project-based introduction to programming* (2nd edition).

Shaw, Z. A. (2017). *Learn Python 3 the hard way: A very simple introduction to the terrifyingly beautiful world of computers and code*. Addison-Wesley.

Beazley, D. M., & Jones, B. K. (2013). *Python cookbook: Recipes for mastering Python 3* (Third edition). O'Reilly.

Far Western University

Faculty of Humanities & Social Sciences

Course Title**: Computer Hardware and Troubleshooting**

Course Code: **BCAC115** Nature of course: Theoretical + Practical

Level: Bachelor Credit hours: 3

Semester: First Teaching hours: 64 (32T + 32P)

7. Course Description

This course aims to provides theory and skills about computer hardware, mobile repair, and system troubleshooting. Students will learn to assemble PCs, diagnose faults, repair mobile devices, and resolve operating system issues using real tools and simulators. Emphasis is placed on practical skills, safety, and problem-solving. Students will gain confidence in handling hardware, configuring systems, and using diagnostic software in real-world scenarios.

8. Course General Objectives

Following are the general objective of this course:

- To assemble and configure desktop computers using compatible components.
- To diagnose and repair hardware faults in PCs and mobile devices.
- To use BIOS/UEFI and diagnostic tools to troubleshoot system issues.
- To perform OS installation, recovery, and driver management.
- To apply preventive maintenance techniques to extend system life.
- To build repair procedures and communicate technical solutions clearly.

9. Specific Objectives and Contents

Specific Objectives	Contents
• Identify computer types and hardware components. • Label motherboard architecture accurately. • Install and test a CPU safely. • Install and troubleshoot RAM modules. • Connect storage devices and	**Unit 1: Fundamentals of Computer Hardware(6T+4P)** 1.1 Types of computers and hardware 1.2 Motherboard architecture 1.3 CPU types and installation 1.4 RAM types and troubleshooting 1.5 Storage devices (HDD, SSD, NVMe) 1.6 Power supply units and voltages 1.7 Cooling systems and thermal management

measure PSU voltages.	1.8 Safety and ESD precautions **Lab Work** • Identify and install components • Use software-based diagnostics using tools like Open Hardware Monitor and Speccy for real-time component monitoring. • Digital multimeter usage and include voltage range interpretation
• Describe the basic function of a microprocessor. • Identify instruction sets and registers. • Compare clock speeds and processor performance. • Distinguish between microprocessors and microcontrollers. • Locate and explain chipsets and buses. • Configure BIOS/UEFI settings. • Diagnose CPU and motherboard issues using POST and beep codes.	**Unit 2: Microprocessor and System Architecture (6T+4P)** 2.1 Microprocessor basics 2.2 Instruction sets and registers 2.3 Clock speed and performance 2.4 Microprocessor vs microcontroller 2.5 Chipsets and buses 2.6 BIOS/UEFI configuration 2.7 POST and beep codes 2.8 Troubleshooting CPU and motherboard **Lab work** • BIOS update • Firmware updates for GPU and SSD • Diagnose POST errors • Replace and test CPUs
• Plan a PC build based on user needs and budget. • Check compatibility between hardware components. • Organize case layout and manage cables efficiently. • Install operating systems and essential drivers. • Set up and configure peripheral devices. • Perform benchmarking and stress testing for performance.	**Unit 3: PC Assembly and Maintenance (2T+8P)** 1.9 PC build planning 1.10 Component compatibility 1.11 Case and cable management 1.12 Installing OS and drivers 1.13 Peripheral setup 1.14 Benchmarking and stress testing 1.15 Preventive maintenance 1.16 Troubleshooting assembled PCs **Lab work** • Assemble a PC from scratch • Install Windows/Linux

• Apply preventive maintenance and troubleshoot assembled PCs	• Perform stress tests • Use *VirtualBox* or *VMware* to simulate OS installation and system testing. • Automate driver installation using tools like *Snappy Driver Installer* or *Driver Booster or similar tools*
• Describe the basic architecture of mobile devices. • Identify and test battery and charging circuit issues. • Repair or replace mobile displays and touchscreens. • Diagnose and fix network and SIM-related problems. • Perform software flashing and system recovery. • Recognize common faults in Android and iOS devices. • Use mobile diagnostic tools safely and effectively.	**UNIT 4: Mobile Device Hardware and Repair (6T+6P)** 4.9 Mobile architecture overview 4.10 Battery and charging circuits 4.11 Display and touchscreen repair 4.12 Network and SIM issues 4.13 Software flashing and recovery 4.14 Common Android/iOS faults 4.15 Mobile diagnostic tools 4.16 Safety and handling 4.17 Battery calibration **Lab work** • Disassemble and reassemble phones • Replace screens and batteries • Flash firmware and recover bricked devices
• Explain the OS boot process and its stages. • Identify and resolve common OS errors like BSOD and boot loops. • Use safe mode and recovery tools for system repair. • Repair registry and system files to restore functionality. • Remove viruses and malware using security tools. • Detect and fix driver conflicts. • Perform backup, restore, and OS reinstallation or cloning.	**Unit 5: Operating System Troubleshooting (6T+4P)** 5.8 OS boot process 5.9 Common OS errors (BSOD, boot loops) 5.10 Safe mode and recovery tools 5.11 Registry and system file repair 5.12 Virus and malware removal 5.13 Driver conflicts 5.14 Backup and restore 5.15 OS reinstallation and cloning **Lab work** • Use recovery tools • Remove virus and malware from infected devices • Repair corrupted OS • Clone and restore systems

	• Use *Windows Event Viewer* and *Linux journaling* for system error analysis
• Use a multimeter to test power supply and voltage levels. • Apply POST diagnostic cards to identify hardware faults. • Utilize software tools for system diagnostics. • Recover lost data using appropriate recovery tools. • Troubleshoot issues in printers and scanners. • Identify and resolve laptop-specific hardware problems. • Perform component-level repair and maintain service documentation.	**Unit 6: Diagnostic Tools and Repair (6T+6P)** 6.10 Multimeter and power testing 6.11 POST diagnostic cards 6.12 Software tools (HWMonitor, MemTest86) 6.13 Data recovery tools 6.14 Printer and scanner troubleshooting 6.15 Laptop-specific issues 6.16 Component-level repair 6.17 Remote desktop tools: AnyDesk, TeamViewer 6.18 Customer service and documentation **Lab work** • Use diagnostic cards • Recover data from damaged drives • Troubleshoot laptops and peripherals • Remote support through AnyDesk, TeamViewer • Teach service documentation using digital ticketing systems like *Freshdesk* or *Zoho Desk*.

2 **Instructional Techniques**

Lecture preferably with the use of multi-media projector, demonstration, practical classes, discussion, and brain storming. The following tools are essential;

- Multimeter
- Screwdriver set
- POST diagnostic card
- Thermal paste and applicator
- Antistatic wrist strap
- Mobile opening toolkit
- Soldering iron
- USB bootable recovery drives
- Software like HWMonitor, CrystalDiskInfo, MemTest86, Rufus etc.

The class can be made better with following techniques:

- Hands-on lab first approach
- Visual and interactive tools

- Problem-based learning (PBL)
- Checklist and task-based assessment
- Role play and customer interaction

3 Evaluation

Evaluation of students' performance is divided into parts: Internal assessment, practical examination and theoretical examinations. The distribution of points is given below:

Internal Assessment	Practical Exam/Viva	Semester Examination (Theoretical exam)	Total Points
40 Points	20 Points	40 Points	100 Points

***Note**: Students must pass separately in internal assessment, external practical exam and semester examination.*

3.1 Internal Assessment (40 Points)

Internal assessment will be conducted by subject teacher based on following criteria:

Attendance

Internal Exam

Readiness for learning

Lab Work / Project work / Presentation

Total	**40 points**

3.2 Semester Examination (40 Points)

ExExamination Division, Dean office will conduct final examination at the end of semester. Short answer questions Long answer questions
Total **40 points**

3.3 Practical Examination (20 Points)

Internal Evaluation	External Evaluation			Total
	Lab Report	Lab Exam	VIVA	
5 points	5 points	5 points	5 points	20 points

4 Recommended Books and References Materials

Roberts, R. M. (2021). *Computer service and repair* (Fifth edition). The Goodheart-Willcox Company, Inc.

Felix, B. (2024). *The complete manual for mobile phone repair: Expert Guidance for Every Step*. Independently Published.

Mueller, S. (2015). *Upgrading and repairing PCs* (22nd edition). Que.

James, K. L. (2013). *Computer hardware: Installation, interfacing, troubleshooting and maintenance*. PHI Learning Private Limited.