પરિપત્ર:

ભક્તકવિ નરસિંહ મહેતા યુનિવર્સિટીની કોમ્પ્યુટર સાયન્સ વિદ્યાશાખાનાં અભ્યાસક્રમ ચલાવતી તમામ સંલગ્ન કોલેજોનાં આચાર્યશ્રીઓને સવિનય જણાવવાનું કે કોમ્પ્યુટર સાયન્સ વિદ્યાશાખા હેઠળનો NEP-૨૦૨૦ અંતર્ગત બી.સી.એ વિષય સેમેસ્ટર-૬નો અભ્યાસક્રમ આ સાથે સામેલ છે.

માનનીય કુલપતિશ્રીની મંજુરી અનુસાર સદર અભ્યાસક્રમ શૈક્ષણિક વર્ષ જુન,૨૦૨૫ થી અમલવારી કરવાની રહે છે. કોમ્પ્યુટર સાયન્સ વિદ્યાશાખાનાં અભ્યાસક્રમ ચલાવતી તમામ સંલગ્ન કોલેજો ધ્વારા તેની અમલવારી કરવા જણાવવામાં આવે છે.

Bhakta Kavi Narsinh Mehta University Junagadh.

14/10/2025

ખાસ ફરજ પરના અધિકારી
(એકેડેમિક)

ક્રમાંક/બીકેએનએમયુ/ એકેડેમિક/૭૨૨/૨૦૨૫
ભક્તકવિ નરસિંહ મહેતા યુનિવર્સિટી,
સરકારી પોલીટેકનિક કેમ્પસ,
ભક્તકવિ નરસિંહ મહેતા યુનિવર્સિટી રોડ,
ખડીયા, જૂનાગઢ-૩૬૨૨૬૩
તા.૧૪/૧૦/૨૦૨૫

પ્રતિ,

- ભક્તકવિ નરસિંહ મહેતા યુનિવર્સિટી સંલગ્ન કોમ્પ્યુટર સાયન્સ વિદ્યાશાખાનાં અભ્યાસક્રમો ચલાવતી તમામ કોલેજોના આચાર્યશ્રીઓ તરફ....

નકલ સાદર રવાના:-

- માન.કુલપતિશ્રી/ કુલસચિવશ્રીનાં અંગત સચિવશ્રી.
- પરીક્ષા નિયામકશ્રી, ભક્તકવિ નરસિંહ મહેતા યુનિવર્સિટી, જુનાગઢ

નકલ રવાના જાણ તથા યોગ્ય કાર્યવાહી અર્થે:

- સીસ્ટમ મેનેજરશ્રી, આઇ.ટી.સેલ વિભાગ (વેબસાઇટ ઉપર પ્રસિદ્ધ થવા અર્થે.)

# BHAKTA KAVI NARSINH MEHTA UNIVERSITY

# JUNAGADH

## BOARD OF STUDIES

## FACULTY OF COMPUTER SCIENCE

## SYLLABUS FOR B.C.A (HONOURS)

## PROGRAMME (SEMESTER- VI)

## EFFECTIVE FROM JUNE, 2025

**BHAKTA KAVI NARSINH MEHTA UNIVERSITY**
**Syllabus of B.C.A (Honors) as per NEP-2020**
**Faculty of Computer Science**
**Effective from June 2025**
**Subject: Computer Science**
**SEMESTER- VI**

## SUMMARY OF THE SYLLABUS

| Sem No. | Sr. No. | Category of Course | Course Title | Course Level | Credit | Teaching Hrs. | SEE Marks | CCE Marks | Total Marks | Exam Duration |
|---|---|---|---|---|---|---|---|---|---|---|
| Sem-6 | 1 | Major-14 | Mobile Programming Using Android (Java) (Theory) | 5.5 | Theory- 04 | Theory- 60 | 50 | 50 | 100 | 2:00 Hrs. |
| | 2 | Major-15 | Mobile Programming Using Android (Java) (Practical) | 5.5 | Practical- 04 | Practical- 120 | 50 | 50 | 100 | 2:00 Hrs. |
| | 3 | Major-16 | Web Development Using React.js | 5.5 | Theory- 03<br>Practical- 01 | Theory- 45<br>Practical- 30 | 50 | 50 | 100 | 2:00 Hrs. |
| | 4 | Minor-6 | Advanced Python Programming for AI and Machine Learning | 5.5 | Theory- 03<br>Practical- 01 | Theory- 45<br>Practical- 30 | 50 | 50 | 100 | 2:00 Hrs. |
| | | | | **Total** | **16** | | | | | |

**BHAKTA KAVI NARSINH MEHTA UNIVERSITY**
**Syllabus of B.C.A (Honors) as per NEP-2020**
**Faculty of Computer Science**
**Effective from June 2025**
**Subject: Computer Science**
**SEMESTER- VI**

| | | | | |
|---|---|---|---|---|
| **Course Level** | 5.5 | | **Internal Marks** | 50 |
| **Programme** | Bachelor of Computer Application | | **External Marks** | 50 |
| **Semester** | VI | | **Practical Internal** | 0 |
| **Category of Course** | Major-14 | | **Practical External** | 0 |
| **Course Credit** | 04 | | **Prac. External Exam Duration** | - |
| **Teaching Hours** | Theory- 60<br>Practical-00 | | **Total** | 100 |
| **Course Code** | | | **Exam Duration** | 2:00 Hrs |
| **Course Title** | **Mobile Programming Using Android (Java) (Theory)** | | | |

**Course Objectives:**

- Understand the architecture and components of the Android platform and its development environment.
- Develop user-friendly Android applications using essential UI components and layouts.
- Utilize Android's data storage mechanisms, including Shared Preferences, SQLite, and Firebase.
- Implement location-based services and Google Maps integration for geospatial features.
- Integrate real-time cloud services using Firebase for authentication, data storage, and syncing.
- Handle Android networking using RESTful APIs and libraries like Retrofit for client-server communication.
- Apply Android system components such as services, intents, content providers, and broadcast receivers effectively.
- Design, test, and publish a complete Android application on the Google Play Store.

**Course Outcomes:**

- Explain the structure of Android applications and the role of the Android SDK, OHA, and Android Runtime.
- Develop interactive mobile apps using various Android UI elements and layout managers.
- Design responsive user interfaces with custom dialogs, animations, and fragments.
- Perform local data operations using SQLite and implement cloud storage using Firebase services.
- Use Location-Based Services including GPS, geocoding, Google Maps, and geo-fencing in real-time apps.
- Integrate external web services and APIs using Retrofit for data exchange with JSON and MySQL.
- Implement notifications, telephony features, and background task handling in Android apps.
- Design, develop, debug, and publish a professional-level mobile application on the Play Store.

**Course Contents:**

| UNIT | CONTENT | Hours |
|---|---|---|
| **Unit 1** Introduction to Android Development | • The Open Handset Alliance<br>• Android Platform and Android SDK<br>• Setting up Android Studio, Emulator, and Debugging Environment<br>• Building a Sample Android Application<br>• Anatomy of an Android Application<br>• Key Android Terminologies<br>• Application Context, Activities, Services, and Intents<br>• Activity Lifecycle and Fragment<br>• Receiving and Broadcasting Intents<br>• Android Manifest File and Common Settings<br>• Using Intent Filters and Permissions<br>• Managing Application Resources in a Hierarchy<br>• Working with Different Types of Resources | **15** |
| **Unit 2** User Interface and Graphics | • User Interface Screen Elements:<br>  ○ Button, EditText, TextView<br>  ○ DatePicker, TimePicker<br>  ○ ProgressBar, ListView, GridView<br>  ○ RadioGroup, ImageButton<br>  ○ Fragment<br>• Designing User Interfaces with Layouts:<br>  ○ RelativeLayout, LinearLayout<br>  ○ TableLayout, GridLayout<br>  ○ Dialogs<br>• Material Design Guidelines and Modern UI/UX Principles<br>• Drawing and Working with Animations:<br>  ○ Frame-by-Frame Animation<br>  ○ Tweened Animations: Fade In, Fade Out, Cross Fading, Blink, Zoom In/Out, Rotate, Move, Slide Up/Down, Bounce, Sequential, Together. | **15** |
| **Unit 3** Data Storage, Firebase, and Location-Based Services | • Using Android Data and Storage APIs<br>• Managing Data with SQLite (Insert, Update, Delete, Select)<br>• Sharing Data Between Applications with Content Providers<br>• Firebase Integration in Android Applications<br>  ○ Cloud Firestore, Realtime Database, Firebase Authentication, Firebase Storage<br>• Practical Implementation & Mini Projects | **15** |

| | | |
|---|---|---|
| | ○ (e.g., Notes App, Chat App, To-Do List, Health Tracker App)<br>• Introduction to Location-Based Services (LBS)<br>• Accessing Location Using GPS and Network Providers<br>• Geocoding and Reverse Geocoding Techniques<br>• Integrating Google Maps in Android Applications<br>• Working with Map Markers and Map Customization<br>• Real-Time Location Tracking and Sharing<br>• Geo-fencing and Proximity Alerts<br>• Background Location Updates and Foreground Services<br>• Firebase Integration for Real-Time Location Sync<br>• Exception Handling and Debugging Best Practices | |
| **Unit 4** Networking, APIs, Testing, and Deployment | • Networking in Android Using APIs<br>• Working with RESTful APIs and JSON<br>• Retrofit Library for Efficient API Communication<br>• Understanding and Using Android Web & Telephony APIs<br>• Creating and Customizing Android Notifications<br>• Using Vibrator, Lights, and Status Bar for User Feedback<br>• Android Services and Background Task Handling<br>• Application Development Using JSON and MySQL<br>• Basics of Testing in Android (Unit Testing, Instrumentation Testing)<br>• Version Control with Git and GitHub for Android Projects<br>• Publishing Android Applications on Google Play | **15** |

**Reference Books:**

1. Lauren Darcey and Shane Conder, Android Wireless Application Development, 2nd Edition, Pearson Education, 2011.
2. Reto Meier, Professional Android Application Development, Wiley India Pvt. Ltd, 2011.
3. Mark L. Murphy, Beginning Android, Wiley India Pvt. Ltd.
4. Jay A. Kreibich, Using SQLite, O'Reilly Media.
5. Mobile Computing using Android & iPhone

**Web Site References:**

1. https://www.tutorialspoint.com/android/
2. https://www.javatpoint.com/android-tutorial
3. https://developer.android.com/guide
4. https://developer.android.com/training/basics/firstapp
5. https://www.studytonight.com/android/

| | | | | |
|---|---|---|---|---|
| **Course Level** | 5.5 | | **Internal Marks** | 0 |
| **Programme** | Bachelor of Computer Application | | **External Marks** | 0 |
| **Semester** | VI | | **Practical Internal** | 50 |
| **Category of Course** | Major-15 | | **Practical External** | 50 |
| **Course Credit** | 04 | | **Prac. External Exam Duration** | 2:00 Hrs |
| **Teaching Hours** | Theory- 0<br>Practical-120 | | **Total** | 100 |
| **Course Code** | | | **Exam Duration** | - |
| **Course Title** | **Mobile Programming Using Android (Java) (Practical)** | | | |

**Course Objectives:**

- Understand the architecture and components of the Android platform and its development environment.
- Develop user-friendly Android applications using essential UI components and layouts.
- Utilize Android's data storage mechanisms, including Shared Preferences, SQLite, and Firebase.
- Implement location-based services and Google Maps integration for geospatial features.
- Integrate real-time cloud services using Firebase for authentication, data storage, and syncing.
- Handle Android networking using RESTful APIs and libraries like Retrofit for client-server communication.
- Apply Android system components such as services, intents, content providers, and broadcast receivers effectively.
- Design, test, and publish a complete Android application on the Google Play Store.

**Course Outcomes:**

- Explain the structure of Android applications and the role of the Android SDK, OHA, and Android Runtime.
- Develop interactive mobile apps using various Android UI elements and layout managers.
- Design responsive user interfaces with custom dialogs, animations, and fragments.
- Perform local data operations using SQLite and implement cloud storage using Firebase services.
- Use Location-Based Services including GPS, geocoding, Google Maps, and geo-fencing in real-time apps.
- Integrate external web services and APIs using Retrofit for data exchange with JSON and MySQL.
- Implement notifications, telephony features, and background task handling in Android apps.
- Design, develop, debug, and publish a professional-level mobile application on the Play Store.

**Course Contents:**

| UNIT | CONTENT | Hours |
|---|---|---|
| **Unit 1** Introduction to Android Development | • The Open Handset Alliance<br>• Android Platform and Android SDK<br>• Setting up Android Studio, Emulator, and Debugging Environment<br>• Building a Sample Android Application<br>• Anatomy of an Android Application<br>• Key Android Terminologies<br>• Application Context, Activities, Services, and Intents<br>• Activity Lifecycle and Fragment<br>• Receiving and Broadcasting Intents<br>• Android Manifest File and Common Settings<br>• Using Intent Filters and Permissions<br>• Managing Application Resources in a Hierarchy<br>• Working with Different Types of Resources | **15** |
| **Unit 2** User Interface and Graphics | • User Interface Screen Elements:<br>  ○ Button, EditText, TextView<br>  ○ DatePicker, TimePicker<br>  ○ ProgressBar, ListView, GridView<br>  ○ RadioGroup, ImageButton<br>  ○ Fragment<br>• Designing User Interfaces with Layouts:<br>  ○ RelativeLayout, LinearLayout<br>  ○ TableLayout, GridLayout, etc.<br>  ○ Dialogs<br>• Material Design Guidelines and Modern UI/UX Principles<br>• Drawing and Working with Animations:<br>  ○ Frame-by-Frame Animation<br>  ○ Tweened Animations: Fade In, Fade Out, Cross Fading, Blink, Zoom In/Out, Rotate, Move, Slide Up/Down, Bounce, Sequential, Together | **15** |
| **Unit 3** Data Storage, Firebase, and Location-Based Services | • Using Android Data and Storage APIs<br>• Managing Data with SQLite (Insert, Update, Delete, Select)<br>• Sharing Data Between Applications with Content Providers<br>• Firebase Integration in Android Applications<br>  ○ Cloud Firestore, Realtime Database, Firebase Authentication, Firebase Storage<br>• Practical Implementation & Mini Projects | **15** |

| | | |
|---|---|---|
| | ○ (e.g., Notes App, Chat App, To-Do List, Health Tracker App)<br>• Introduction to Location-Based Services (LBS)<br>• Accessing Location Using GPS and Network Providers<br>• Geocoding and Reverse Geocoding Techniques<br>• Integrating Google Maps in Android Applications<br>• Working with Map Markers and Map Customization<br>• Real-Time Location Tracking and Sharing<br>• Geo-fencing and Proximity Alerts<br>• Background Location Updates and Foreground Services<br>• Firebase Integration for Real-Time Location Sync<br>• Exception Handling and Debugging Best Practices | |
| **Unit 4** Networking, APIs, Testing, and Deployment | • Networking in Android Using APIs<br>• Working with RESTful APIs and JSON<br>• Retrofit Library for Efficient API Communication<br>• Understanding and Using Android Web & Telephony APIs<br>• Creating and Customizing Android Notifications<br>• Using Vibrator, Lights, and Status Bar for User Feedback<br>• Android Services and Background Task Handling<br>• Application Development Using JSON and MySQL<br>• Basics of Testing in Android (Unit Testing, Instrumentation Testing)<br>• Version Control with Git and GitHub for Android Projects<br>• Publishing Android Applications on Google Play | **15** |

**Reference Books:**

1. Lauren Darcey and Shane Conder, Android Wireless Application Development, 2nd Edition, Pearson Education, 2011.
2. Reto Meier, Professional Android Application Development, Wiley India Pvt. Ltd, 2011.
3. Mark L. Murphy, Beginning Android, Wiley India Pvt. Ltd.
4. Jay A. Kreibich, Using SQLite, O'Reilly Media.
5. Mobile Computing using Android & iPhone

**Web Site References:**

1. https://www.tutorialspoint.com/android/
2. https://www.javatpoint.com/android-tutorial
3. https://developer.android.com/guide
4. https://developer.android.com/training/basics/firstapp
5. https://www.studytonight.com/android/

| | | | | |
|---|---|---|---|---|
| **Course Level** | 5.5 | | **Internal Marks** | 25 |
| **Programme** | Bachelor of Computer Application | | **External Marks** | 50 |
| **Semester** | VI | | **Practical Internal** | 25 |
| **Category of Course** | Major-16 | | **Practical External** | 0 |
| **Course Credit** | 04 | | **Prac. External Exam Duration** | - |
| **Teaching Hours** | Theory- 45<br>Practical-30 | | **Total** | 100 |
| **Course Code** | | | **Exam Duration** | 2:00 Hrs |
| **Course Title** | **Web Development Using React.js** | | | |

**Course Objectives:**

- Learn the fundamentals of JavaScript and its role in web development.
- Understand variables, conditionals, loops, functions, arrays, and events in JavaScript.
- Gain knowledge of ES6 features such as classes, arrow functions, promises, and modern syntax.
- Get introduced to React JS, its history, and Single Page Applications (SPAs).
- Set up the React development environment using Node.js, npm, and project structure.
- Build a foundation in React core concepts: JSX, Babel, Virtual DOM, React DOM.
- Develop skills in creating React components: functional, class-based, pure, and higher-order.
- Work with props, state management, rendering, conditional rendering, and list rendering.
- Learn event handling and various binding techniques in React.
- Practice form handling in React applications.
- Master React Hooks (useState, useEffect, useReducer, useRef) for managing state and side effects in functional components.

**Course Outcomes:**

- Write and apply JavaScript (ES6) concepts including variables, functions, loops, arrays, events, classes, promises, and modern syntax.
- Set up and configure a React environment, create apps, and understand project structure.
- Build and manage different React components (functional, class, pure, higher-order) using props and state effectively.
- Implement rendering techniques, event handling, and form management for dynamic user interfaces.
- Utilize React advanced features like Fragments, Memo, Refs, and Hooks (useState, useEffect, useReducer, useRef) for state, side effects, and performance optimization.
- Develop a complete single-page React application that fetches API data, manages state efficiently, and delivers an interactive user experience.

**Course Contents:**

| UNIT | CONTENT | Hours |
|---|---|---|
| **Unit 1** Introduction to Java Script and React JS | • Introduction to Java Script<br>• Java Script Overview & Basics<br>• Variable, Conditional Statements, Loops in JS,<br>• Functions, Arrays & Events in JS<br>• ES6 Overview & Basics<br>• ES6 Classes, functions & Promises<br>• Introduction to React JS<br>• History and evolution of React<br>• Concept of Single Page Applications (SPA)<br>• React Environment Setup<br>• Install Node.js and npm<br>• Create a first app<br>• Explain folder structure<br>• React Fundamentals<br>• React JSX<br>• React JS Babel Introduction<br>• React JS Virtual DOM<br>• React JS React DOM | **10** |
| **Unit 2** React Components | • React Components<br>• Functional Component<br>• Class Component<br>• Pure Components<br>• Props in react<br>• Functional based props in react<br>• Class based props in react<br>• Destructuring props in react<br>• State in react<br>• State in functional component<br>• State in class component<br>• Destructuring state in react<br>• Rendering in react<br>• Conditional Rendering and List Rendering<br>• Higher Order Components | **10** |
| **Unit 3** Advanced Component Interaction & Event Handling in React | • Fragment in react<br>• Memo in react<br>• Refs in react<br>• Event Handling<br>• Event Handler with Functional Component<br>• Event Handler with Class Component<br>• Passing Arguments with functional Component | **15** |

| | | |
|---|---|---|
| | • Passing Arguments with Props<br>• Passing Arguments with Class component<br>• Binding event Handlers<br>• In constructor<br>• In Render method<br>• Using Arrow function | |
| **Unit 4** Introduction to Hooks in React | • Introduction to Hooks<br>• useRef<br>• useState<br>• useState with object<br>• useState with Array<br>• useEffect<br>• Fetching data from APIs<br>• Conditionally run effects<br>• useEffect with cleanup<br>• useReducer<br>• simple state and action<br>• Complex state and action<br>• Multiple useReducers<br>• Fetching data with useReducers<br>• useReducer VS useState | **10** |
| **Practical** | | **30** |

**Reference Books:**

- **Test-Driven React, Second Edition** by Trevor Burnham — published September 2024.
- The Road to React (2025 Edition) by Robin Wieruch — last updated May 2025.
- React Key Concepts – Second Edition: An in-depth guide to React's core features
- ReactJS Notes for Professional, GoalKicker, Website ebook.
- Learning React, Martin Bean, Kirupa Chinnathambi Pearson Addison Wesley
- Codevolution. "ReactJS Tutorial for Beginners."

**Web Site References:**

- https://react.dev
- https://reactrouter.com
- https://www.w3schools.com/react/
- https://developer.mozilla.org/
- https://www.freecodecamp.org/learn/
- https://scrimba.com/learn/learnreact

| | | | | |
|---|---|---|---|---|
| **Course Level** | 5.5 | | **Internal Marks** | 25 |
| **Programme** | Bachelor of Computer Application | | **External Marks** | 50 |
| **Semester** | VI | | **Practical Internal** | 25 |
| **Category of Course** | Minor – 6 | | **Practical External** | 0 |
| **Course Credit** | 04 | | **Prac. External Exam Duration** | - |
| **Teaching Hours** | Theory-45<br>Practical-30 | | **Total** | 100 |
| **Course Code** | | | **Exam Duration** | 2:00 Hrs |
| **Course Title** | **Advanced Python Programming for AI and Machine Learning** | | | |

**Course Objectives:**

- Understand Core Database Operations
- Gain Fundamental Knowledge of AI and ML
- Apply Machine Learning Techniques

**Course Outcomes:**

- Perform fundamental database operations
- Demonstrate understanding of AI principles and classification.
- Apply machine learning methods and tools.
- Evaluate ML models effectively.

**Course Contents:**

| Unit | Content | Hours |
|---|---|---|
| **Unit 1**<br>Basics of Database Connectivity in Python (with Tkinter GUI) | • Introduction **to Database Connectivity with Tkinter**<br>  ○ Overview of Tkinter for creating database-driven GUI applications<br>  ○ Importing the required libraries for MySQL, SQLite, and CSV<br>• CSV Operations with Tkinter<br>  ○ Reading data from CSV using csv module and pandas<br>  ○ Writing data into CSV files<br>  ○ Handling delimiters, headers, and encoding through GUI inputs<br>• MySQL **Operations with Tkinter**<br>  ○ Installing and setting up MySQL-connector-python<br>  ○ Designing Tkinter forms for database interaction<br>  ○ Executing SQL queries (SELECT, INSERT, UPDATE, DELETE) from GUI<br>  ○ Fetching results and displaying them in Tkinter widgets (e.g., labels, tables, listboxes) | **10** |

| | | |
|---|---|---|
| | • SQLite **Operations with Tkinter**<br>  ○ Creating and connecting to SQLite databases<br>  ○ Executing SQL queries (SELECT, INSERT, UPDATE, DELETE) via GUI<br>  ○ Displaying query results in Tkinter GUI components | |
| **Unit 2** Introduction to Artificial Intelligence and Machine Learning | • Overview of Artificial Intelligence (AI)<br>• Core concept of Artificial Intelligence (AI)<br>• Types of Artificial Intelligence (AI)<br>• Models of Artificial Intelligence (AI)<br>• Key features of Artificial Intelligence (AI)<br>• Real-world applications of AI<br>• Challenges of AI<br>• Introduction to Machine Learning (ML)<br>• Machine Learning Development Life Cycle (MLDLC)<br>• Artificial Intelligence using Machine Learning<br>• Categories of Machine Learning:<br>  ○ Supervised Learning<br>  ○ Unsupervised Learning<br>  ○ Reinforcement Learning<br>• **Case Study (Python):** Building a simple supervised learning model (predicting student grades based on study hours) | **10** |
| **Unit 3** Datasets and Data Preparation | • What is a dataset? Importance in ML<br>• Types of data: numerical, categorical, ordinal<br>• How to obtain, clean, and use datasets<br>• Data pre-processing techniques: handling missing values, scaling, normalization, encoding<br>• Data visualization techniques (scatter plots, histograms, box plots)<br>• Exploratory data analysis (EDA) basics<br>• **Case Study (Python):** Cleaning and visualizing a real-world dataset (e.g., Titanic survival dataset) | **10** |
| **Unit 4** Machine Learning Algorithms and Model Evaluation | • Regression Algorithms:<br>  ○ Simple Linear Regression<br>  ○ Multiple Linear Regression<br>• Classification Algorithms:<br>  ○ Logistic Regression<br>  ○ Decision Tree<br>  ○ Random Forest<br>  ○ K-Nearest Neighbours (K-NN)<br>  ○ Support Vector Machine (SVM)<br>• Clustering Algorithm:<br>  ○ K-Means Clustering<br>• Model training and testing techniques:<br>  ○ Train-Test Split<br>  ○ Cross-Validation<br>  ○ K-Fold Cross-Validation | **15** |

| | | |
|---|---|---|
| | • Model evaluation metrics:<br>  ○ For Regression: $R^2$ Score, MAE, MSE<br>  ○ For Classification: Accuracy, Precision, Recall, F1-Score, Confusion Matrix<br>• Model optimization basics: feature selection, hyperparameter tuning overview<br>• **Case Studies (Python):**<br>  ○ House price prediction using Linear Regression<br>  ○ Spam email classification using Logistic Regression<br>  ○ Comparing multiple models on the same dataset to evaluate performance | |
| **Practical** | | **30** |

**Reference Books:**

1. Python: The Complete Reference
2. Learning React, Martin Bean, Kirupa Chinnathambi Pearson Addison Wesley
3. AI 2025: The Definitive Guide to Artificial Intelligence, APIs, and Python Programming for the Future
4. Artificial Intelligence with Python by Teik Toe Teoh & Zheng Rong — Springer, 2022
5. Artificial Intelligence: A Modern Approach Authors: Stuart Russell, Peter Norvig
6. Machine Learning Author: Tom M. Mitchell Publisher: McGraw-Hill
7. Deep Learning Authors: Ian Goodfellow, Yoshua Bengio, Aaron Courville Publisher: MIT Press
8. Advanced Deep Learning with Python (O'Reilly)

**Web Site References:**

1. https://www.tutorialspoint.com/python
2. https://www.w3schools.com/python
3. https://www.tutorialspoint.com/artificial_intelligence/index.htm
4. https://www.geeksforgeeks.org/artificial-intelligence/
5. https://www.ibm.com/cloud/learn/what-is-artificial-intelligence

**Reference Tools:** Free Machine Learning Tools and Platforms

1. **Google Colab** – Free Jupiter notebook environment with GPU support.
2. **Kaggle** – Datasets, kernels, and free notebook environment for ML projects.
3. **Scikit-learn** – Open-source Python library for ML algorithms.
4. **TensorFlow** (Community Version) – Free ML library for deep learning and neural networks.
5. **Weka** – Free GUI-based ML software for beginners.
6. **Orange3** – Open-source data visualization and ML tool with drag-and-drop interface.

| INTERNAL EVALUATION SCHEME | | |
|---|---|---|
| **NO** | **Particulars** | **Marks** |
| 1 | **Mid Semester Exam/ Internal Practical Exam (Mandatory)** | **25** |
| 2 | Class Test | 05 |
| 3 | Open book exam/test | 05 |
| 4 | Open note exam/test | 05 |
| 5 | Self-test/ Online test | 05 |
| 6 | Essay/Article writing | 05 |
| 7 | Quizzes/Objective test | 05 |
| 8 | Class assignment | 05 |
| 9 | Home assignment | 05 |
| 10 | Reports Writing | 05 |
| 11 | Research/Dissertation | 05 |
| 12 | Case Studies | 05 |
| 13 | Viva/Oral exam | 05 |
| 14 | Group Discussion | 05 |
| 15 | Role Play | 05 |
| 16 | Paper presentation/Seminar | 05 |
| 17 | Language Lab work | 05 |
| 18 | Interview | 05 |
| 19 | Craft work | 05 |
| 20 | Co-curricular work | 05 |
| 21 | Field Assignment | 05 |
| 22 | Poster Presentation | 05 |
| 23 | Attendance | 05 |
| 24 | Project Work | 05 |
| | **Total** | **50** |

**Note: Sr.No.1 is mandatory. Select any five from Sr.No.2 to 24. Each Contains five marks. Student should secure 18 Marks for passing in internal Exam.**

**Paper Style:**

| Ques. No. | Particulars | From which Unit | Marks |
|---|---|---|---|
| 1 | Questions (Any Two Out Of Four) | 1 | 10 |
| 2 | Questions (Any Two Out Of Four) | 2 | 10 |
| 3 | Questions (Any Two Out Of Four) | 3 | 10 |
| 4 | Questions (Any Two Out Of Four) | 4 | 10 |
| 5 | Questions (Any Two Out Of Four) | From Each Unit | 10 |
| | | Total Marks | 50 |

**BHAKTA KAVI NARSINH MEHTA UNIVERSITY**
**Ability Enhancement Skill**
**Syllabus of B.C. A & B.Sc. (IT) (Honors) as per NEP-2020**
**Faculty of Computer Science**
**Effective from June 2025**
**Subject: Computer Science**
**SEMESTER- VI**

| | | | |
|---|---|---|---|
| **Course Level** | 5.5 | **Internal Marks** | 25 |
| **Programme** | BCA And B.Sc. (IT) | **External Marks** | 25 |
| **Semester** | VI | **Practical Internal** | 0 |
| **Category of Course** | AEC-5 | **Practical External** | 0 |
| **Course Credit** | 02 | **Prac. External Exam Duration** | - |
| **Teaching Hours** | Theory- 30<br>Practical-00 | **Total** | 50 |
| **Course Code** | | **Exam Duration** | 1:00 Hrs. |
| **Course Title** | **Introduction to Cyber Security** | | |

**Course Objectives:**

- Provide foundational knowledge of information security and cybersecurity concepts.
- Familiarize students with the roles of hackers, penetration testers, and cybersecurity professionals.
- Introduce ethical hacking methodologies and common cybersecurity terminology.
- Explain key cryptographic techniques and their importance in securing data.
- Develop understanding of hashing algorithms, integrity verification, and password security.
- Demonstrate practical tools such as CyberChef and Wireshark for real-world cybersecurity applications.
- Provide networking fundamentals essential for analyzing and securing communication systems.
- Enable students to monitor, analyze, and interpret network traffic using packet analysis techniques.

**Course Learning Outcomes**:

- Differentiate between information security and cybersecurity, explaining their scope and applications.
- Identify types of hackers, major attack methods, and common cybersecurity terms (payload, exploit, malware, etc.).
- Apply ethical hacking processes and penetration testing principles in simulated environments.
- Demonstrate cryptographic methods (AES, DES, RSA) and evaluate their role in data protection.
- Analyze network traffic and detect suspicious/malicious activities using Wireshark.
- Relate learned concepts to potential cybersecurity career roles and certifications.

**BHAKTA KAVI NARSINH MEHTA UNIVERSITY**
**Ability Enhancement Skill**
**Syllabus of B.C. A & B.Sc. (IT) (Honors) as per NEP-2020**
**Faculty of Computer Science**
**Effective from June 2025**
**Subject: Computer Science**
**SEMESTER- VI**

| Sem | UNIT | CONTENT | Hours |
|---|---|---|---|
| 6 | 1 | **1. Introduction to Information and Cybersecurity**<br>a. Definition of Information Security<br>b. Definition of Cybersecurity<br>Key objectives of security: Confidentiality, Integrity, Availability (CIA Triad)<br>Difference between Information Security and Cybersecurity<br>c. Importance and real-world applications of cybersecurity<br>**2. Hackers and Cybersecurity Roles**<br>**a.** Definition of a Hacker<br>b. Types of Hackers<br>c. Overview of Cybersecurity Roles<br>**3. Ethical Hacking Process and Penetration Testing**<br>a. Stages of Ethical Hacking<br>b. Penetration Testing<br>**4. Common Cybersecurity Terms**<br>a. Payload<br>b. Malware<br>c. Exploit<br>d. Zero-day Exploit<br>e. Vulnerability | **15** |
| | 2 | **1. Introduction to Cryptography**<br>a. Importance in cybersecurity<br>b. Symmetric vs Asymmetric Encryption<br>c. Demonstration of AES, DES, RSA using CyberChef<br>**2. Hashing & Integrity**<br>a. Definition and purpose<br>b. Common algorithms: MD5, SHA family<br>c. File integrity verification with hashes<br>**3. Packet Analysis with Wireshark**<br>a. Introduction to Wireshark & packet capture<br>b. Monitoring traffic and analyzing protocols (TCP, HTTP, ICMP) | **15** |

**Suggested Practice**:

1. Find the IP address of testphp.vulnweb.com with Nslookup.
2. Send ICMP echo requests (ping) from your machine to any IP address and use Wireshark to capture and analyze the ICMP traffic.
3. Visit "http://testphp.vulnweb.com/" from your browser and use Wireshark to capture and analyze the TCP three-way handshake between your machine and the web server.
4. Access testphp.vulnweb.com use Wireshark to capture the HTTP traffic. Analyze the server's response headers to determine which server version is running on the web server.
5. Visit google.com in your browser and use Wireshark to capture and analyze the DNS query and response generated by your system.
6. AES Encryption and Decryption demo using CyberChef.

**Note**: **The above practices are supported across Windows, macOS and Linux environments**.

**Tools**:

**1) Wireshark 2) CyberChef**

1) **Wireshark**:

1. Download: Visit https://www.wireshark.org/download.html
2. Choose Your OS: Select the installer for Windows, macOS, or Linux.
3. Install: Run the installer and follow the default prompts.
   - Windows: Accept installing Npcap when prompted (needed for packet capture).
   - macOS: You may need to allow permissions in System Settings → Privacy & Security.
   - Linux: Use your package manager (e.g., sudo apt install wireshark on Debian/Ubuntu).
4. Verify: Open Wireshark and ensure your network interfaces are listed for capture.

2) **CyberChef**:

1. Access Online (Recommended): Go to https://gchq.github.io/CyberChef/ —no installation required.

2. Run a Test: Drag-and-drop data or paste text, select an operation (e.g., Base64 Decode), and view the output instantly.

**Reference Books:**

1. William Stallings, Network Security Essentials: Applications and Standards, Pearson, 2017.
2. Nina Godbole and Sunit Belapure, Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Wiley, 2011.
3. Charles J. Brooks, Christopher Grow, Philip Craig, and Donald Short, Cybersecurity Essentials, Wiley, 2018.
4. Mark Ciampa, Security+ Guide to Network Security Fundamentals, Cengage Learning, 2018.
5. Behrouz A. Forouzan, Cryptography and Network Security, McGraw Hill, 2016.

**Web site References:**

1. CyberChef: https://cyberchef.io/
2. Wireshark: https://www.wireshark.org/docs/wsug_html_chunked/

**BHAKTA KAVI NARSINH MEHTA UNIVERSITY**
**Ability Enhancement Skill**
**Syllabus of B.C. A & B.Sc. (IT) (Honors) as per NEP-2020**
**Faculty of Computer Science**
**Effective from June 2025**
**Subject: Computer Science**
**SEMESTER- VI**

| INTERNAL EVALUATION SCHEME | | |
|---|---|---|
| **NO** | **Particulars** | **Marks** |
| 1 | **Mid Semester Exam (Mandatory)** | **13** |
| 2 | Class Test | 03 |
| 3 | Open book exam/test | 03 |
| 4 | Open note exam/test | 03 |
| 5 | Self-test/ Online test | 03 |
| 6 | Essay/Article writing | 03 |
| 7 | Quizzes/Objective test | 03 |
| 8 | Class assignment | 03 |
| 9 | Home assignment | 03 |
| 10 | Reports Writing | 03 |
| 11 | Research/Dissertation | 03 |
| 12 | Case Studies | 03 |
| 13 | Viva/Oral exam | 03 |
| 14 | Group Discussion | 03 |
| 15 | Role Play | 03 |
| 16 | Paper presentation/Seminar | 03 |
| 17 | Language Lab work | 03 |
| 18 | Interview | 03 |
| 19 | Craft work | 03 |
| 20 | Co-curricular work | 03 |
| 21 | Field Assignment | 03 |
| 22 | Poster Presentation | 03 |
| 23 | Attendance | 03 |
| 24 | Project Work | 03 |
| | **Total** | **25** |

**Note: Sr.No.1 is mandatory. Select any Four from Sr.No.2 to 24. Each Contains three marks. Student should secure 09 Marks for passing in internal Exam.**

**Paper Style:**

| **Ques. No.** | **Particulars** | **From which Unit** | **Marks** |
|---|---|---|---|
| **1** | **Questions (Any Two Out Of Four)** | **1** | **10** |
| **2** | **Questions (Any Two Out Of Four)** | **2** | **10** |
| **3** | **Questions (Any One Out Of Two)** | **From Each Unit** | **5** |
| | | **Total Marks** | **25** |