### Semester-V

<table>
<thead>
<tr>
<th>Core Courses</th>
<th>Title</th>
<th>No. of Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>US05CBCA01</td>
<td>Visual Programming Through VB.NET</td>
<td>4</td>
</tr>
<tr>
<td>US05CBCA02</td>
<td>Computer Graphics</td>
<td>4</td>
</tr>
<tr>
<td>US05CBCA03</td>
<td>Software Engineering</td>
<td>4</td>
</tr>
<tr>
<td>US05CBCA04</td>
<td>Practicals</td>
<td>6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Foundation Courses</th>
<th>Title</th>
<th>No. of Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>US05FBCA01</td>
<td>Operations Research</td>
<td>4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Elective</th>
<th>Title</th>
<th>No. of Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>US05EBCA01</td>
<td>Basics of UNIX Operating System</td>
<td>2</td>
</tr>
<tr>
<td>US05EBCA02</td>
<td>Software Project Management</td>
<td></td>
</tr>
</tbody>
</table>

Total Credits 24
BCA (V Semester)
Course : US05CBCA01
(Visual Programming Through VB.NET)

Credits : 4
Lectures per week : 4

All units carry equal weightage.

1. Introduction to .NET Framework and VB.NET
   - .NET Architecture, .NET Languages, Microsoft Intermediate Language (MSIL), The Just-In-Time (JIT) compiler, Working with Assemblies, The .NET framework class library
   - VB.NET - introduction, applications and types of project
   - Introduction to Visual Studio IDE
   - Creating simple Windows Application using VB.NET
   - Variables, data types, constants and operators
   - Type casting, Boxing and Unboxing,
   - Working with arrays and strings
   - Creating simple Windows Application using VB.NET

2. VB.NET Basics
   - Use of conditional statement (if), multibranaching statement (select) and With...EndWith statement,
   - Looping Statement: DO, FOR, FOR EACH..NEXT and WHILE, Working with EXIT, CONTINUE and WITH statements
   - Working with procedures – introduction, types, use of parameters, parameter passing, calling procedures
   - OOP concepts - Encapsulation, Inheritance, Interfaces and Polymorphism
   - Working with modules, classes (partial) and namespaces
   - Working with Windows Forms – introduction, life cycle, basic properties, methods and events, use of simple windows forms control.
   - Working with SDI and MDI forms

3. Developing Windows Forms, Exception Handling
   - Working with basic controls – Button, CheckBox, CheckedListBox, ComboBox, DateTimePicker, GroupBox, HScrollBar, RadioButton, VScrollBar, Label, ListBox, PictureBox, TextBox and Time controls.
   - Working with advanced controls – LinkLabel, RichTextBox, ColorDialog, FontDialog, TreeView, Working with modules, classes (partial) and namespaces
   - Error Handling: exception, structured exception using try...catch and final statement

4. Persisting Data Using Databases and Files
   - ADO.NET – introduction and applications
   - ADO.NET – architecture (connected and disconnected)
   - Database connectivity using ADO.NET
   - Use of Data sources, Server Explorer and working with DataSet
   - Populating data in a DataGridView,
   - Working with files
MAIN REFERENCE BOOKS:
1. Steven Holzner; VB.NET Black Book by Dreamtech publication
2. Francesco Balena: Programming Microsoft Visual Basic.NET, Microsoft Press
3. Bill Evjen, Billy Hollis, Bill Sheldon, Kent Sharkey and Tim McCarthy: Professional VB 2005 with .NET 3.0
1. **Introduction of Computer Graphics**
   - A survey of major applications of Computer Graphics
   - Overview of different video display Devices: CRT, Raster scan, Color Monitors, DVST, Flat Panels
   - Input Devices: Keyboard, mouse, Trackball, Spaceball, Joystick, Data Glove, Digitizers, Image Scanner, Touch Panel, Light pen & Voice system,
   - Hardcopy Devices: Printers and Plotters
   - Graphics Software & coordinate representation, Graphics functions, Software Standards

2. **Output Primitives and their attributes**
   - Output Primitives: Points, Lines, Circles
   - Line Drawing Algorithms (without program): Digital Differential Analyzer (DDA) and Bresenham.
   - Circle generating algorithm(without program): Midpoint Circle Algorithm
   - Filled area primitives
   - Inside – Outside tests: Odd even rule & Non-zero winding number rule
   - Boundary- fill algorithm (with procedure)
   - Flood-Fill Algorithm (with procedure), Character generation,
   - Attributes of output primitives

3. **Two – dimensional Geometric Transformations, Viewing & Clipping**
   - 2-D geometric Transformations : Translation, Rotation, Scaling, Reflection & Shear (with example)
   - Viewing Pipeline, Window-to-Viewport transformation
   - Point Clipping
   - Line clipping (without program)
   - Cohen Sutherland line clipping algorithm
   - Polygon Clipping(without program)
   - Text clipping

4. **Usage of a 2D Animation Package**
   - Timeline Window, Layers, Key Frame, Frames and Toolbox
   - Types of Symbols
   - Types of Animation – Frame By Frame and Tween (Motion and Shape)
   - Onion Skinning
   - Import and Export images
   - Publish settings, Guided Layer, Alpha Effect, Using Color properties (Brightness , Tint), Masking (Using Filled shape, Text), Motion guide and Movie Clip Mask
   - Introduction to Action scripting
   - Play & Stop, GetURL

**MAIN REFERENCE BOOKS :**
2. Macromedia Flash MX 2004 in 24 hours by Robert Renihardt and Snow Dowd
3. FLASH MX – Manual
Course : US05CBCA03
(Software Engineering)

Credits : 4
Lectures per week : 4

All units carry equal weightage.

1. **Introduction**
   - Introduction : Software and Software Engineering
   - General Characteristics of Software Process
   - Phases in Software development
   - Effort and Error Distribution
   - Process Models : Waterfall, Prototype, Iterative enhancement, spiral
   - Software metrics : introduction, product metrics, process metrics

2. **Requirement Specification and Software Project Planning**
   - Introduction : Software Requirement Specification (SRS) and Needs
   - Problem Analysis - Structuring Information
   - Introduction to UML
   - Software Requirement Specifications (SRS), Characteristics and Components of SRS
   - Specification language (**Structured English, Regular Expression and Decision Table**)
   - Structure of SRS, Validation of SRS
   - Introduction: Software Projects, Planning, Categories of Software projects
   - Overview of Cost estimation, Uncertainty in cost estimation, size estimation, COCOMO Model (with example)
   - Project Monitoring Plan : Time sheets, Reviews, Cost- schedule milestone and Earned value method
   - Software Quality Assurance Plans (SQAP)
   - **Overview of Risk Management**

3. **Software Design**
   - Introduction : System Design
   - Design Objectives and Design Principles
   - Design Concepts - Top down and Bottom up approach, Problem Partition, Abstraction, Modularity, Module Level concept, Coupling, Cohesion
   - Overview of structured design
   - Function v/s Object Oriented approach
   - Design Specification, Verification
   - Introduction: Detailed Design
   - Module Specification, Desirable properties, functional module specification, Data abstraction specification
   - PDL, Logic/ Algorithm Design
   - Design Verification – Design Walkthrough, Critical Design review, Consistency checkers
4. Coding and Testing
   - Introduction: Coding, Top Down and Bottom Up approach for coding
   - Structured programming, Information Hiding
   - Programming style, Internal documentation
   - Verification (code reading)
   - Introduction: Testing, Error, Fault, Failure & Reliability
   - Testing process, Top down and bottom up approach for testing
   - Levels of Testing
   - Functional Testing v/s. Structural testing

MAIN REFERENCE BOOKS:

BOOKS FOR ADDITIONAL READING:
Course: US05CBCA04  
(Practicals)

Credits: 6
No. of laboratory hours per week: 12

University examination duration: 4 Hours

Part-I: Weightage-70%
  • Practical based on VB.NET

Part-II: Weightage-30%
  • Practical based on Computer Graphics
Course : US05FBCA01
(Operations Research)

Credits : 4
Lectures per week : 4

All units carry equal weightage.

1. Introduction to Operations Research (OR):
   - History, meaning and scope of OR
   - Phases of OR study
   - Types of Models
   - Applications, advantages and limitations of OR

2. Linear Programming Problem (LPP)
   - Meaning
   - Advantages and limitations
   - Formulation of LPP
   - Graphical solution,
   - Simplex method – Big M method

3. Transportation Model and Assignment Model
   - Introduction
   - Mathematical model of Transportation problem
   - Initial basic feasible solution by North-west corner rule, Least-cost method, Vogel’s approximation method.
   - Optimum Solution by MODI method
   - Introduction to Traveling Salesman Problem.
   - Introduction to an Assignment Model
   - Mathematical model of Assignment problem
   - Solution by Hungarian method

4. Dynamic Programming, Sequencing problems and Project Scheduling in PERT-CPM
   - Introduction to Dynamic Programming.
   - Deterministic & Probabilistic Dynamic Programming.
   - Shortest Route problem.
   - Sequencing problems and Applications.
   - Introduction to PERT and CPM
   - Advantages and Assumptions
   - Rules for Network construction
   - Critical Path calculations, Total float, Free float

MAIN REFERENCE BOOKS:

BOOKS FOR ADDITIONAL READING:
Course: US05EBCA01  
(Basics of UNIX Operating System)

Credits: 2  
Lectures per week: 2

All units carry equal weightage.

1. Working with UNIX-like Systems
   - Brief history of UNIX and LINUX, strengths and weaknesses of UNIX-like operating systems
   - Basic concepts in UNIX-like systems: the kernel, shells, multiuser multitasking operation, remote access, file system, processes, environment and environment variables, the command line, online manual
   - Using the vi editor – modes of operation and switching between them, text navigation, editing text, saving and quitting, using buffers (cut-copy-paste), pattern searching and replacement, undoing and repeating commands
   - Basic commands related to handling files and the file system

2. The Bourne Again Shell (bash)
   - Prompts, the command line, quoting and escaping, internal and external commands, the path, shell variables, basic command line processing
   - Using the echo command
   - A quick introduction to basic filters – cat and cut
   - The building blocks approach
   - Input/output redirection
   - Command substitution

3. Introduction to Shell Scripting
   - Shell scripts
   - Fundamental shell programming constructs – conditional execution, loops, input and output, turning debugging on and off, etc.

4. Shell Scripting using Filters
   - Definition of a filter
   - Basic filters like the grep family, expr, sed, etc.
   - Processing the output of commands like ls, ps, who, etc.
   - Processing data in text files (fixed-width format and delimited format)

MAIN REFERENCE BOOKS:
Course : US05EBCA02
(Software Project Management)

Credits : 2
Lectures per week : 2

All units carry equal weightage.

1. Introduction
   - Definition of the project
   - Project specification and parameters
   - Principles of Project management
   - Project management life cycle
2. Software Project Planning
   - Project activities and Work Breakdown Structure (WBS)
   - Criteria for completeness in the WBS
   - Activity Resource Requirements and Cost
   - Joint project planning session
   - Project management plan
3. Project Economics and Risk Management
   - Project costing, empirical project estimation techniques, decomposition techniques, algorithmic methods, automated estimation tools
   - Risk concepts and identification, risk assessment and control, risk components and drivers, risk tracking and monitoring, risk mitigation and management
4. Project Scheduling and Tracking Techniques
   - Introduction to project scheduling and tracking
   - Effort estimation techniques
   - Task network and scheduling methods, monitoring and control progress
   - Graphical reporting tools

MAIN REFERENCE BOOKS:

BOOKS FOR ADDITIONAL READING: