Faculty of Science & Technology

Savitribai Phule Pune University
Pune, Maharashtra, India

Syllabus for

Third Year
Bachelor of Computer Applications
(2019 Pattern)
(with effect from A.Y. 2021-22)

Prologue

On behalf of Board of Studies in Computer Applications, it is a great pleasure and honor for me to present the syllabi for TYBCA (Third Year Computer Applications - 2019 Pattern).

While revising syllabus, every efforts were made to design the syllabus for Computer Applications program as per the guidelines and objectives set out by the UGC and our University. The draft was prepared by taking into account the recent technological developments and expectations of employers.

The revision of Syllabus was carried out with dedicated efforts, active participation and suggestions from stakeholders. All efforts have been made by the members of Board of Studies and faculty members from affiliated colleges to decide contents as per the guidelines and recommendations of faculty of Science & Technology, SPPU, Pune.

The BoS in Computer Applications had appointed a subcommittee for the preparation of the draft. Two of our BoS members, Dr. Pallawi Unmesh Bulakh and Dr. M N Shelar coordinated the task of preparation of drafts for Semesters V and VI respectively.

For each course, the draft suggests course objectives and outcomes and could be suitably modified, if needed, by faculty members from affiliated colleges teaching these courses.

I am thankful all the faculty members, members of subcommittee, Heads of the departments and all members of the BoS Prof. Dr. R M Sonar, Dr. M N Shelar, Dr. Pallavi Bulakh and Dr. Razak Sayyad for their help and support in preparation of this draft.

I appreciate various suggestions received through emails and during the online workshop to finalize the draft contents.

Regards,

Dr. Shirish S Sane
Chairman, Board of Studies (Computer Applications), SPPU, Pune

Faculty of Science & Technology

Savitribai Phule Pune University

Third Year of Computer Applications (2019 Pattern)

(With effect from Academic Year 2021-22)

Table of Contents

Savitribai Phule Pune University, Pune

Faculty of Science and Technology

Third Year Bachelor of Computer Application (TYBCA)

Pattern-2019 (With effect from A.Y. 2021-22)

Structure of TYBCA								
Course Code	Course Name	Teaching Scheme (Hours /week)		Examination Scheme			Credits	
		TH	PR	CE	ESE	Total	TH	PR
Semester-V								
BCA351	DSE I (Programming in Java)	5	--	30	70	100	4	--
BCA352	DSE II (Data Mining and Data Science)	5	--	30	70	100	4	--
BCA353	DSE III (Principles of Operating Systems)	5	--	30	70	100	4	--
BCA354	SEC I (Artificial Intelligence)	3	--	15	35	50	2	--
BCA355	SEC II (Cloud Computing)	3	--	15	35	50	2	--
BCA356	DSE I Lab (Programming in Java)	--	4*	15	35	50	--	2
BCA357	DSE II Lab (Data Mining)	--	4*	15	35	50	--	2
BCA358	DSE III Lab (Operating Systems and AI)	--	4*	15	35	50	--	2
		21	**12***	**165**	**385**	**550**	**16**	**6**
Semester-VI								
BCA361	DSE IV Android Programming	5	--	30	70	100	4	--
BCA362	DSE V Programming in GO	5	--	30	70	100	4	--
BCA363	DSE VI Software Project Management	5	--	30	70	100	4	--
BCA364	SEC III Management Information Systems	3	--	15	35	50	2	--
BCA365	SEC IV Internet of Things (IoT)	3	--	15	35	50	2	--
BCA366	DSE IV Lab (Android Programming)	--	4*	15	35	50	--	2
BCA367	DSE V Lab (Programming in GO and IoT)	--	4*	15	35	50	--	2
BCA368	DSE VI Project Lab	--	4*	15	35	50	--	2
		21	**12***	**165**	**385**	**550**	**16**	**6**

TH: Theory **PR**: Practical **CE**: Continuous Evaluation **ESE**: End Semester Examination

DSE: Discipline Specific Elective **SEC**: Skill Enhancement Course

* Laboratory sessions each of 4 hours 20 minutes duration to be conducted for each batch of 12 students in each week

Semester V

Semester-V

SEMESTER V

BCA 351: DSE I (Programming in Java)

Teaching Scheme:	Credits	Examination Scheme:
Theory: 05 Hrs./Week	**04**	**Continuous Evaluation: 30 Marks** **End-Semester : 70 Marks**

Pre-requisite Course:

- BCA 241 Object Oriented Programming and C++

Companion Course:

- BCA 356 DSE I Laboratory (Programming in Java)

Course Objectives:

- To learn implementation of object-oriented conceptswith Java.
- To understand collection classes and interfaces.
- To know the process of application development using Graphical User Interface (GUI).
- To acquire knowledge about handling databases using Java.
- To study web components for developing web applications.

Course Outcomes:

After successful completion of this course, learner will be able to:

- Identify classes, objects, class members and relationships for a given problem.
- Design end to end applications using object oriented constructs.
- Apply collection classes for storing java objects.
- Use Java APIs for program development.
- Handle abnormal termination of a program using exception handling.

	Course Contents	
Unit I	**Introducing Java, Classes and Objects**	**10 Hrs**
	Introducing Java 1.1 A Short History of Java 1.2 Features of Java 1.3 Java Environment – Compiler, Interpreter, JVM 1.4 Simple java program 1.5 Types of Comments 1.6 Declaring single and multi-dimensional arrays 1.7 Accepting input using Command line arguments 1.8 Accepting input from console (Using BufferedReader and Scanner class) **Classes and Objects** 1.9 Defining Your Own Classes 1.10 Access Specifiers (public, protected, private, default) 1.11 Array of Objects 1.12 Constructor, Overloading Constructors and use of „this‟ Keyword 1.13 static blocks, static Fields and static methods 1.14 Predefined classes – Object class methods (equals(), toString(),	

	hashcode()) 1.15 Garbage Collection (finalize() Method)	
Unit II	**Inheritance and Interface**	**10 Hrs**
	Inheritance 2.1 Inheritance Basics (extends Keyword) and Types of Inheritance 2.2 Superclass, Subclass and use of Super Keyword 2.3 Method Overriding and runtime polymorphism 2.4 Use of final keyword related to variable, method and class 2.5 Use of abstract class and abstract methods **Interface** 2.6 Defining and Implementing Interfaces 2.7 Runtime polymorphism using interface **Packages** 2.8 Creating, Accessing and using Packages	
Unit III	**Collections, Exception Handling and I/O**	**10 Hrs**
	Collections 3.1 Wrapper Classes 3.2 Introduction to the Collection framework 3.3 List – ArrayList, LinkedList and Vector 3.4 Set - HashSet, TreeSet, and LinkedHashSet 3.5 Map – HashMap, LinkedHashMap, Hashtable and TreeMap 3.6 Interfaces such as Iterators, ListIterators, Enumerations **Exception Handling** 3.7 Exception class, Checked and Unchecked exception 3.8 Catching exception and exception handling – try, catch, finally, throw and throws, multiple catch block 3.9 Creating user defined exception **I/O** 3.10String class(basic methods) , String Buffer class 3.11 File class 3.12 DataInputStream and DataOutputStream class	
Unit IV	**Swing**	**10 Hrs**
	4.1 What is Swing? 4.2 The MVC Architecture and Swing 4.3 Layout Manager and Layouts, The JComponent class 4.4 Components – JLabel, JButton, JText, JTextArea, JCheckBox, JRadioButton, JList, JComboBox, JMenu and JPopupMenu Class, JMenuItem 4.5 Dialogs (Message, confirmation, input), JFileChooser 4.6 Event Handling: Event sources, Listeners – ActionListener, ItemListener 4.7 Mouse and Keyboard Event Handling 4.8 Adapters – MouseAdapter, KeyAdapter 4.9 Anonymous inner class	
Unit V	**Database Programming**	**10 Hrs**

	5.1 The role of jdbc, jdbc configuration 5.2 Types of drivers 5.3 Connectivity with database 5.4 JDBC Statements – Statement, PreparedStatement, CallableStatement 5.5 Scrollable and updatable result sets - TYPE_FORWARD_ONLY, TYPE_SCROLL_INSENSITIVE, TYPE_SCROLL_SENSITIVE - CONCUR_READ_ONLY, CONCUR_UPDATABLE 5.6 Metadata – DatabaseMetadata, ResultSetMetadata (Database : PostgreSQL)	
Unit VI	**Servlets & JSP**	**10 Hrs**
	Servlets 6.1 Introduction to Servlets and Hierarchy of Servlets 6.2 Life cycle of a servlet 6.3 Tomcat configuration (Note: Only for Lab Demonstration) 6.4 Handing get and post request (HTTP) 6.5 Handling a data from HTML to a servlet 6.6 Session tracking – Cookies and Http Session **JSP** 6.7 Simple JSP program 6.8 Life cycle of a JSP 6.9 Implicit Objects 6.10 Scripting elements – Declarations, Expressions, Scriplets, Comments 6.11 JSP Directives – Page Directive, include directive 6.12 Mixing Scriplets and HTML	
Reference Books: 1) Core Java Volume I - Fundamentals By Cay S. Horstmann, 11th Edition, Prentice Hall, ISBN 978-0-13-516630-7 2) The Complete Reference By Herbert Shildt, 11th Edition, McGraw Hill Education, ISBN 978-260-44023-2 3) Java Beginners Guide By Herbert Shildt, 8th Edition, McGraw-Hill Education ISBN 978-1-260-44021-8 4) Core Java Volume II – Fundamentals By Cay S. Horstmann, 11th Edition, Prentice Hall, ISBN 978-013-516631-4 5) Java 2 Programming Black Book By Steven Holzner, DreamTech Press, ISBN 978-93-5119-953-4		
E-Books: 1) The Complete Reference By Herbert Shildt_ https://bbooks.info/viewmore/java-the-complete-reference-eleventh-edition 2) Java 2 Programming Black Book By Steven Holzner https://idoc.pub/documents/java-2-black-book-steven-holzner-vyly2rmq9v4m		
Online Courses: https://nptel.ac.in/courses/106/105/106105191/		

SEMESTER V

BCA 352: DSE II - Data Mining and Data Science

Teaching Scheme:	Credits	Examination Scheme:
Theory: 05 Hrs./Week	**04**	**Continuous Evaluation: 30 Marks** **End-Semester : 70 Marks**

Pre-requisite Course:

- BCA 117 Applied Mathematics Laboratory
- BCA 232 Database Management Systems-II

Companion Course:

- BCA 357 DSE- II Laboratory (Data Mining)

Course Objectives:

- To introduce students to the basic concepts and techniques of Data Mining and Data Science
- To study data mining algorithms for solving practical problems.
- To understand data visualization and exploratory analysis.

Course Outcomes:

After successful completion of this course, learner will be able to:

- Identify the key processes of data mining, data warehousing and knowledge discovery
- Design data warehouse with dimensional modeling and apply OLAP operations.
- Identify appropriate data mining algorithms to solve real world problems
- Compare and evaluate different data mining techniques like classification, prediction, clustering and association rule mining
- Choose an appropriate method to perform exploratory analysis.
- Interpret results by carrying out data visualization and formal inference procedures

	Course Contents	
Unit I	**Introduction to Data Mining**	**10 Hrs**
	1.1 Definition Data mining 1.2 Data Mining issues 1.3 Stages of the Data Mining Process (KDD) 1.4 Data Mining Techniques/Tasks 1.5 Knowledge Representation Methods 1.6 Applications of Data mining 1.7 Data Pre-processing 1.7.1 Data Cleaning 1.7.2 Data Transformation 1.7.3 Data Reduction 1.7.4 Data Discretization	
Unit II	**Data Warehousing**	**08 Hrs**
	2.1 Introduction to Data Warehouse 2.2 Data Warehouse Architecture and its components 2.3 Data Modeling with OLAP 2.3.1 Introduction 2.3.2 Difference between OLTP and OLAP	

	2.3.3 Data Mart 2.3.4 Fact Table, Dimension Table, OLAP cube 2.3.5 Different OLAP Operations 2.4 Schema Design 2.4.1 Introduction 2.4.2 Star and snow-Flake Schema 2.5 Introduction to Machine Learning 2.6 Introduction to Pattern Matching 2.7 Case study based on Schema Design	
Unit III	**Classification**	**12 Hrs**
	3.1 Introduction, Definition 3.2 Decision Tree 3.2.1 Introduction 3.2.2 Construction Principle 3.2.3 Attribute Selection Measures 3.2.4 Tree Pruning 3.3 Rule-Based Classification 3.3.1 Using IF-THEN Rules for Classification 3.3.2 Rule Extraction from a Decision Tree 3.4 Bayes Classification Methods 3.4.1 Bayes‟ Theorem 3.4.2 Naive Bayesian Classification 3.5 Bayesian Networks 3.6 Parameter and structure learning 3.7 Linear classifier 3.8 Perceptron 3.9 k-Nearest-Neighbor Classifiers 3.10 SVM classifiers 3.10.1 Introduction 3.10.2 Types of SVM 3.10.3 Working of SVM 3.11 Regression 3.11.1 Linear Regression 3.11.2 Non linear Regression 3.12 Introduction to Prediction	
Unit IV	**Clustering and Association Rule Mining**	**10 Hrs**
	4.1 Cluster Analysis 4.1.1 Introduction 4.1.2 Requirements for Cluster Analysis 4.2 Hierarchical Methods 4.2.1 Agglomerative Hierarchical Clustering 4.2.2. Divisive Hierarchical Clustering 4.3 Partitioning Methods 4.3.1 k-Means: A Centroid-Based Technique 4.3.2 k-Medoids: A Representative Object-Based Technique	

	4.4 Introduction to Association Rule Mining Market Basket Analysis, Items, Itemsets and Large Itemsets 4.5 Apriori Algorithm 4.6 Kinds of Association Rules Mining Multilevel association rules Constraint Based Association rules mining	
Unit V	**Introduction to Data Science**	**10 Hrs**
	5.1. Basics of Data 5.2 What is Data Science? 5.3. Data science process 5.4. Stages in a Data Science project 5.5 Applications of Data Science in various fields 5.6. Basics of Data Analytics 5.7 Types of Analytics – Descriptive, Predictive, Prescriptive 5.8 Statistical Inference - Populations and samples - Statistical modeling - probability distributions	
Unit VI	**EDA and Data Visualization**	**10 Hrs**
	6.1 What is Exploratory Data Analysis ? 6.2 Steps in EDA 6.3 Basic tools (plots, graphs and summary statistics) of EDA 6.4 Types of exploratory data analysis 6.5 Basic principles of data visualization 6.6 Benefits of Data Visualization 6.7 Data visualization techniques 6.8 Tools for data visualization	

Reference Books:

1) Data Mining : Introductory and Advanced Topics by Margaret Dunham, S. Sridhar, Pearson Publication

2) Data Mining concepts and Techniques by Jiawei Han and Micheline Kamber, ELSEVIER, Third Edition,

3) R and Data Mining, By Yanchang Zhao, Elsevier Inc., ISBN-10: 0123969638

4) Data Science from Scratch: First Principles with Python By O‟Reilly Media, 20153.

5) Making sense of Data: A practical Guide to Exploratory Data Analysis and Data Mining by Glenn J. Myatt John Wiley Publishers, 2007

E-Books:

1. http://myweb.sabanciuniv.edu/rdehkharghani/files/2016/02/The-Morgan-Kaufmann-Series-in-Data-Management-Systems-Jiawei-Han-Micheline-Kamber-Jian-Pei-Data-Mining.-Concepts-and-Techniques-3rd-Edition-Morgan-Kaufmann-2011.pdf
2. https://data-flair.training/blogs/data-mining-tutorial/
3. http://infolab.stanford.edu/~ullman/mmds/book.pdf
4. https://archive.org/details/IntroductionToDataMining
5. https://www.academia.edu/6489220/Data_Mining_ebook
6. https://en.wikibooks.org/wiki/Data_Mining_Algorithms_In_R
7. https://www.cs.umd.edu/~samir/498/10Algorithms-08.pdf
8. http://www.elfhs.ssru.ac.th/morakot_wo/file.php/1/9781788290678-

STATISTICS_FOR_DATA_SCIENCE.pdf

9. https://www.programmer-books.com/introducing-data-science-pdf/
10. https://www.cs.uky.edu/~keen/115/Haltermanpythonbook.pdf
    http://math.ecnu.edu.cn/~lfzhou/seminar/[Joel_Grus]_Data_Science_from_Scratch_First_Princ.pdf
11. https://www.pdfdrive.com/doing-data-science-d58735039.html

WebSites:

https://www.datacamp.com/community/open-courses/statistical-inference-and-data-analysis

https://www.coursera.org/learn/python-plotting?specialization=data-science-python

https://epgp.inflibnet.ac.in/

Online Courses:

1) Data Mining : https://onlinecourses.swayam2.ac.in/cec19_cs01/preview
2) https://onlinecourses.nptel.ac.in/noc20_cs12/preview
3) https://www.classcentral.com/course/swayam-data-mining-13982
4) Data Science For Engineers : https://onlinecourses.nptel.ac.in/noc19cs60
5) Introduction to Data Science in Python :https://www.coursera.org/learn/python-data-analysis
6) Python for Data Science : https://onlinecourses.nptel.ac.in/noc21cs33

SEMESTER V

BCA 353: DSE III (Principles of Operating Systems)

Teaching Scheme:	**Credits**	**Examination Scheme:**
Theory: 05 Hrs./Week	**04**	**Continuous Evaluation: 30 Marks** **End-Semester : 70 Marks**

Pre-requisite Course:

- BCA123 Operating System Concepts

Companion Course:-

- BCA 358 DSE III Laboratory (Operating Systems & AI)

Course Objectives:

- To study algorithms for CPU-scheduling, process creation and termination.
- To understand the notion of a Multithreading and Inter-process communication.
- To learn critical-section problems and classical process-synchronization problems.
- To know the concept of deadlock, different methods for preventing or avoiding deadlocks and techniques for memory management.
- To learn and understand file system, directory structure, file allocation methods and disk scheduling algorithms.

Course Outcomes:

After successful completion of this course, learner will be able to:

- Describe algorithms for process, memory and disk scheduling
- Apply technique for inter-process communication and Multithreading.
- Implement concept of critical-section
- Compare and contrast deadlock avoidance and prevention.
- Use functions for file system management

	Course Contents	
Unit I	**Process Scheduling and Multithreaded Programming**	**14 Hrs**
	1.1 Process Scheduling – Scheduling queues, Schedulers, context switch 1.2 Operations on Process – Process creation with program using fork(), Process termination 1.3 Interprocess Communication – Shared memory system, Message passing systems 1.4 Multithreaded Programming – Overview, Multithreading Models 1.5 Basic Concept – CPU-I/O burst cycle, CPU Scheduler, Pre-emptive Scheduling, Dispatcher 1.6 Scheduling Criteria 1.7 Scheduling Algorithms – FCFS, SJF, Priority scheduling, Round-robin scheduling, Multiple queue scheduling, Multilevel feedback queue scheduling 1.8 Thread Scheduling	
Unit II	**Process Synchronization**	**8 Hrs**

	2.1 Background 2.2 Critical Section Problem 2.3 Semaphores: Usage, Implementation 2.4 Classic Problems of Synchronization – The bounded buffer problem, The reader writer problem, The dining philosopher problem	
Unit III	**Deadlocks**	**10 Hrs**
	3.1 System Model 3.2 Deadlock Characterization – Necessary Conditions, Resource Allocation Graph 3.3 Deadlock Prevention 3.4 Deadlock Avoidance - Safe state, Resource-Allocation-Graph Algorithm, Banker‟s Algorithm 3.5 Deadlock Detection 3.6 Recovery from Deadlock – Process Termination, Resource Pre-emption	
Unit IV	**Memory Management**	**12 Hrs**
	4.1 Background – Basic Hardware, Address Binding, Logical Versus Physical Address Space, Dynamic Loading, Dynamic Linking and Shared Libraries, Overlays 4.2 Swapping 4.3 Contiguous Memory Allocation – Memory mapping and protection, Memory allocation, Fragmentation 4.4 Paging – Basic Method, Hardware support, Protection, Shared Pages 4.5 Segmentation – Basic concept, Hardware 4.6 Virtual Memory Management – Demand paging, Performance of demand paging, Page replacement – FIFO, Optimal, LRU, Second Chance Algorithm 4.7 Thrashing – Cause of thrashing, Working-Set Model	
Unit V	**File System**	**10 Hrs**
	5.1 File Concept, File Attribute, File Operations, File Types, File Structure 5.2 Access Methods - Sequential Access Method, Direct Access Method, Other Access Methods 5.3 Directory overview, Single level directory, Two level directory, Tree structure directory, Acyclic graph directory, General graph directory 5.4 File System Structure and Implementation - Partitions and Mounting, Virtual File Systems 5.5 Allocation Methods - Contiguous allocation, Linked allocation, Indexed allocation 5.6 Free Space Management – Bit vector, Linked list, Grouping, Counting, Space maps	
Unit VI	**Disk Scheduling**	**6 Hrs**

<table>
<tr><td></td><td>6.1 Overview
6.2 Disk Structure
6.3 Disk Scheduling, FCFS Scheduling, SSTF Scheduling, SCAN Scheduling, C-SCAN Scheduling, LOOK Scheduling
6.4 Disk Management</td><td></td></tr>
<tr><td colspan="3"><b>Reference Books:</b>
1) Operating System Concepts, Avi Silberschatz, Peter Galvin, Greg Gagne, 8th Edition, Wiley Asia
2) Operating Systems: Internals and Design Principles, William Stallings, Prentice Hall of India.
3) The Design of the UNIX Operating System By Maurice J. Bach., PHI publication</td></tr>
<tr><td colspan="3"><b>E-Books:</b>
1) http://www.uobabylon.edu.iq/download/M.S%202013-2014/Operating_System_Concepts,_8th_Edition%5BA4%5D.pdf</td></tr>
</table>

SEMESTER V

BCA354: Artificial Intelligence

Teaching Scheme:	Credits	Examination Scheme:
Theory: 3 Hrs./Week	**02**	**Continuous Evaluation: 15 Marks** **End-Semester :35 Marks**

Pre-requisite Course:

- BCA112 Problem Solving and C Programming
- BCA231 Data Structures.

Companion Course:

- BCA358 Laboratory (Operating Systems and AI)

Course Objectives:

- To learn various types of algorithms useful in Artificial Intelligence (AI).
- To convey the ideas in AI research related to emerging technology.
- To introduce ideas and techniques underlying the design of intelligent computer systems

Course Outcomes:

After successful completion of this course, learner will be able to:

- Apply the suitable algorithms to solve AI problems
- Identify and apply suitable Intelligent agents for various AI applications
- Build smart system using different informed search / uninformed search or heuristic approaches
- Represent complex problems with expressive language of representation

	Course Contents	
Unit I	**Introduction to Artificial Intelligence**	**4 Hrs**
	1.1 Introduction to AI 1.2 Comparison of AI, Machine Learning, Deep Learning 1.3 Applications of AI 1.4 AI Techniques 1.5 Intelligent Agents, Agents and Environments, Structure of Agents	
Unit II	**Problems, Problem Spaces and search**	**5 Hrs**
	2.1 Defining problem as a State Space Search 2.2 Production System 2.3 Problem Characteristics 2.4 Search & Control Strategies 2.5 Problems – Water Jug problem, Missionary Cannibal Problem, Block words Problem, Monkey & Banana problem	
Unit III	**Searching Algorithms**	**6 Hrs**
	3.1 Uninformed Search Algorithms/Blind Search Techniques 3.1.1 Breadth-first Search 3.1.2 Depth-first Search 3.2 Informed (Heuristic) search Techniques 3.2.1 Generate-and-test 3.2.2 Simple Hill Climbing	

	3.2.3 Best First Search 3.2.4 Constraint Satisfaction 3.2.5 Means End Analysis 3.2.6 A* and AO*	
Unit IV	**Knowledge Representation**	**7 Hrs**
	4.1 Definition of Knowledge 4.2 Types of knowledge (Procedural and Declarative knowledge) 4.3 Approaches to Knowledge Representation 4.4 Knowledge representation using Propositional and Predicate logic 4.5 Conversion to clause form 4.6 Resolution in Propositional logic 4.7 Resolution in Predicate logic	
Unit V	**Slot and Filler Structures**	**4 Hrs**
	5.1 Weak structures (Semantic networks and Frame) 5.2 Strong structures (Conceptual dependencies and Script)	
Unit VI	**Recent Trends in AI and Applications**	**4 Hrs**
	6.1 Introduction to Machine Learning, 6.2 Types of Learning, (Supervised, Unsupervised and Reinforcement Learning), 6.3 Predictive Analytics (Weather Forecasting) 6.4 AI-Powered Chatbots (SBI card chatbot (ILA))	

Reference Books:

1) Artificial Intelligence, Tata McGraw Hill, Elaine Rich and Kevin Knight
2) Computational Intelligence, Eberhart, Elsevier, ISBN 9788131217832
3) Artificial Intelligence: A New Synthesis, Nilsson, Elsevier, ISBN 9788181471901
4) Introduction to Artificial Intelligence and Expert System, Dan Patterson, Prentice Hall of India Pvt. Ltd., New Delhi, 1997
5) Artificial Intelligence: A Modern Approach, Russel &Norvig, Pearson Education
6) Introduction to Machine Learning, Ethem Alpaydin, PHI

E-References:

1) https://www.oracle.com/in/chatbots/what-is-a-chatbot/
2)https://www.dataversity.net/case-study-predictive-analytics-and-data-science-keep-an-eye-on-the-weather/
3) https://www.senseforth.ai/conversational-ai-case-studies/SBI-Cards/

<table>
<tr><th colspan="3">SEMESTER V</th></tr>
<tr><th colspan="3">BCA 355: SEC II (Cloud Computing)</th></tr>
<tr><td><b>Teaching Scheme:</b>
<b>Theory: 3 Hrs./Week</b></td><td><b>Credits</b>
<b>02</b></td><td><b>Examination Scheme:</b>
<b>Continuous Evaluation: 15 Marks</b>
<b>End-Semester : 35 Marks</b></td></tr>
</table>

Pre-requisite Course:

- BCA233 Computer Network

Companion Course: NIL

Course Objectives:

- To study cloud computing concepts, technologies, architecture and applications.
- To understand issues in application deployment and implementations in cloud environment.
- To learn recent trends in cloud computing.

Course Outcomes:

After successful completion of this course, learner will be able to:

- Explain the core issues in cloud computing such as security, privacy, and interoperability.
- Choose the appropriate technologies, algorithms, and approaches for the given application.
- Compare and contrast various cloud services

	Course Contents	
Unit I	**Introduction to Cloud Computing**	**3 Hrs**
	1.1 Overview, Layers and Types of Cloud 1.2 Desired Features of a Cloud 1.3 Benefits and Disadvantages of Cloud Computing 1.4 Cloud Infrastructure Management 1.5 Infrastructure as a Service	
Unit II	**Abstraction and Virtualization**	**6 Hrs**
	2.1 Using Virtualization Technology 2.2 Load Balancing and Virtualization – The Google Cloud 2.3 Understating Hypervisors – Virtual Machine types 2.4 Exploring SaaS – salesforce.com 2.5 Exploring PaaS- force.com, Exploring IaaS – Amazon EC2	
Unit III	**Programming Environment**	**6 Hrs**
	3.1 Features of Cloud and Grid Platforms, 3.2 Programming Support of Google App Engine 3.3 Programming on Amazon AWS 3.4 Microsoft Azure 3.5 Emerging Cloud Software Environments.	

Unit IV	**Deploying Applications and cloud services**	**5 Hrs**
	4.1 Moving application to cloud 4.2 Microsoft Cloud Services 4.3 Google Cloud Applications 4.4 Amazon Cloud Services 4.5 Cloud Applications.	
Unit V	**Emerging trends in cloud computing**	**5 Hrs**
	5.1 Multi-Cloud Vs Omni-Cloud 5.2 Integrated Blockchain technology 5.3 Kubernetes 5.4 Cloud AI 5.5 Intelligent SaaS 5.6 Kubernetes Supremacy 5.7 Containerization by Industry Giants	
Unit VI	**Security In The Cloud**	**5 Hrs**
	6.1 Security Overview 6.2 Cloud Security Challenges and Risks 6.3 Software-as-a-Service Security 6.4 Security Governance 6.5 Risk Management – Security Monitoring 6.6 Security Architecture Design.	

Reference Books:

1) Cloud Computing: Principles and Paradigms, Editors, RajkumarBuyya, James Broberg, Andrzej M. Goscinski, Wiley,2011.

2) Enterprise Cloud Computing - Technology, Architecture, Applications, Gautam Shroff, Cambridge University Press, 2010.

3) Cloud Computing Bible, Barrie Sosinsky, Wiley-India, 2010.

4) Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Ronald L. Krutz, Russell Dean Vines, Wiley- India,2010.

5) Cloud Computing: Technologies and Strategies of the Ubiquitous Data Center, Brian J. S. Chee and Curtis Franklin.

5) AWS, The ultimate guide from beginners to advanced, Maveric Koston.

6) Microsoft Azure: Planning, Deploying, and Managing Your Data Center in the Cloud, Anthony Puca, Mike Manning, Marshal Copeland, Julian Soh, David Gollob.

E-Books:

1) https://openlibrary.org/

2) https://nlist.inflibnet.ac.in/

3) https://archive.org/

4) https://books.google.co.in/

5)https://en.wikibooks.org/wiki/Main_Page

SEMESTER V

BCA356: DSE I Laboratory (Programming in JAVA)

Teaching Scheme: **PR: 4* Hours/Week**	**Credits** **02**	**Examination Scheme:** **Continuous Evaluation: 15 Marks** **End-Semester Exam : 35 Marks**

Prerequisite Courses:

- BCA 241 Object Oriented Programming and C++

Companion Courses:

- BCA 351 DSE I (Programming in Java)

Course Objectives:

- To learn implementation of object-oriented concepts with Java.
- To understand collection classes and interfaces.
- To know the process of application development using Graphical User Interface (GUI).
- To acquire knowledge about handling databases using Java.
- To study web components for developing web applications.

Course Outcomes:

After Completion of this course, students will able to

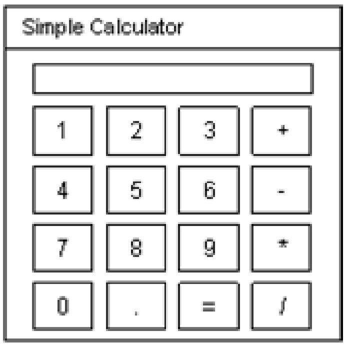
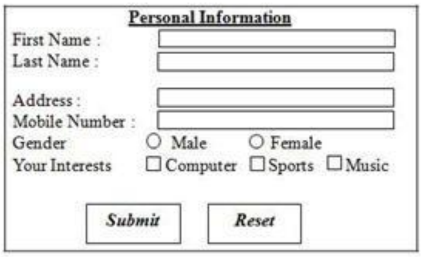
- Identify classes, objects, class members and relationships for a given problem.
- Design end to end applications using object oriented constructs.
- Apply collection classes for storing java objects.
- Use Java APIs for program development.
- Handle abnormal termination of a program using exception handling.

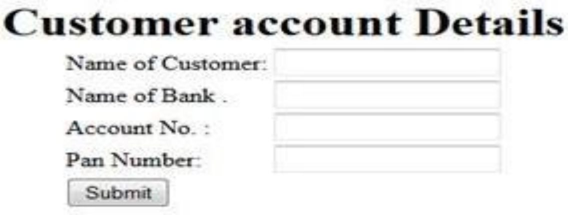
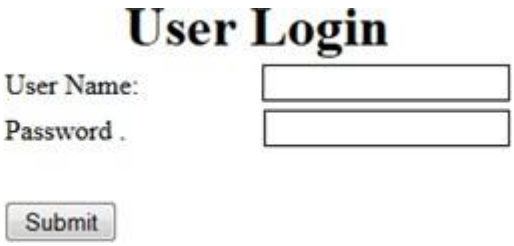
Lab Course Contents

Assignment No	Assignment Name	No.Of Sessions
1	**Basics of Java, Classes and Objects** Classes and objects, Array of objects, Static keyword, Constructor 1. Write a Java program to accept a number from command prompt and generate multiplication table of a number. 2. Write a Java program to print the factors of a number. 3. Write a Java program to display Fibonacci series using recursion. 4. Write a Java program to accept a number from user and print all prime numbers upto that number(Use Buffered Reader class) 5. Write a Java program to print the sum of elements of the array. 6. Write a Java Program which define class Product with data member as id, name and price Store the information of 5	04

	products and Display the name of product having minimum price(Use array of object). 7. Write a Java Program which define class Employee with data member as id, name and salary Store the information of n employees and Display the name of employee having maximum salary (Use array of object). 8. Define a class student having rollno, name and percentage. Define Default and parameterized constructor. Overload the constructor. Accept the 5 student details and display it.(use this keyword). 9. Define a class MyNumber having one private integer data member. Write a default constructor initialize it to 0 and another constructor to initialize it to a value. Write methods isNegative, isPositive, isOdd, iseven. Use command line argument to pass a value to the object and perform the above tests. 10. Define a class CricketPlayer(name, no_of_innings, no_of_times_notout, total_runs, bat_avg). Create an array of "n" player objects. Calculate the batting average for each player using a static method avg (). Handle appropriate exception while calculating average. Define static method "sortPlayer" which sort the array on the basis of average. Display the player details in sorted order.		
2	**Inheritance and Interface** Types of Inheritance, Method Overriding, Super keyword, Abstract classes, Interfaces, Package 1. Write a Java program to create a super class Vehicle having members Company and Price. Derive two different classes LightMotorVehicle (mileage) and HeavyMotorVehicle (capacity_in_tons). Accept the information for "n" vehicles and display the information in appropriate form. While taking data, ask user about the type of vehicle first. 2. Define a class Employee having members – id, name, department, salary. Define default and parameterized constructors. Create a subclass called Worker with private member bonus. Define methods accept and display in both the classes. Create "n" objects of the Worker class and display the details of the worker having the maximum total salary (salary + bonus). 3. Define a class Student with attributes rollno and name. Define default and parameterized constructor. Override the toString () method. Keep the count of Objects created. Create objects using parameterized constructor and Display the object count after each object is created. 4.Create an interface "CreditCardInterface" with methods: viewCreditAmount(), useCard(), payCard(), and increaseLimit(). Create a class "SolverCardCustomer" (name,	04	

	cardnumber (16digit), creditamount-initialized to 0, creditLimit-set to 50,000) which implements above interface. Inherit class GoldCardCustomer from SilverCardCustomer having same methods but creditLimit of 1,00,000. Create an object of each class and perform operations. Display appropriate messages for success or failure of transaction. (Use method overloading) a) useCard() method increase the creditAmount by a specific amount uptocreaditLimit. b) payCreadit() reduces the creditAmount by a specific amount. c) increaseLimit() increases the creaditLimit for GoldCardCustomers (only 3 times, not more than 5000 rupees each time.) 5. Create an abstract class Shape with methods area & volume. Derive two classes Sphere (radius), Cylinder(radius, height) from it. Calculate area and volume of both. (Use Method Overriding) 6. Define an Interface Shape with abstract method area(). Write a java program to calculate an area of Circle and Sphere.(use final keyword) 7. Create a package named Series having three different classes to print series: a. Fibonacci series b. Cube of numbers c. Square of numbers Write a java program to generate „n‟ terms of the above series. 8. Write a package game which will have 2 classes Indoor & Outdoor. Use a function display() to generate the list of player for the specific game. Use default & parameterized constructor.		
3	**Collection, Exception handling and I/O** List, Set, Map interface related classes, Exception handling-try and catch, User defined Exception, File class 1.Accept N integers from the user and store them in a collection. Display them in the sortedorder. The collection should not accept duplicate elements. (Use a suitable collection). 2. Construct a Linked List having names of Fruits: Apple, Banana, Guava and Orange and Display i. the contents of the List using an Iterator; ii. the contents of the List in reverse order using a ListIterator; 3.Create a Hash table containing Student Name and Percentage. Display the contents of the hash table. Also	04	

	search for a specific Student and display his percentage. 4.Write a java program to accept a number from the user, if number is zero then throw user defined exception —Number is 0, otherwise check whether no is prime or not 5. Write a java program to accept Doctor Name from the user and check whether it is valid or not. (It should not contain digits and special symbol) If it is not valid then throw user defined Exception —Name is Invalid -- otherwise display it. 6. Write a class Driver with attributeslicense_no, name, address and age. Initialize values through the parameterized constructor. If age of Driver is less than 18 then user-defined exception should be generated —Age is below 18 years -. 7. Write a java program that displays the number of characters, lines and words of a file. 8. Write a java program to accept details of n customers (c_id, cname, address, mobile_no) from user and store it in a file (Use DataOutputStream class). Display the details of customers by reading it from file.(use DataInputStream class)		
4	**Swing** GUI designing, Event handling on GUI 1.Write a java program to design the following GUI using Swing components.  Simple Calculator 1 2 3 + 4 5 6 - 7 8 9 * 0 . = / 2. Write a java program to design a following GUI using Swing. After submission, display the accepted details on the next page. (Use Event Handling)  Personal Information First Name : Last Name : Address : Mobile Number : Gender ○ Male ○ Female Your Interests □ Computer □ Sports □ Music Submit Reset 3. Write a java program to design the following GUI using Swing components.	04	

	 Customer account Details Name of Customer: Name of Bank . Account No. : Pan Number: Submit 4.Write a program to create following GUI using Swing and check whether username and password is correct or not. (Use Event Handling)  		
5	**Database Programming :** Use DriverManager class, Statement, PreparedStatement interface, ResultSet interface, ResultsetMetaData Interface to Execute DML commands 1. Create a table Country (Name, continent, Capital, Region).Insert values in the table. Write a program to display all the countries located in West Region. 2. Write a program to display information about the ResultSet like number of columns available in ResultSet and SQL type of the column. (Consider the table mentioned in 1.) 3. Write a menu driven program to perform the following operations on the table District(ID, Name, Population, State, Area). a. Insert b. Modify c. Delete d. Search e. Display f. Exit 4. Create a table Course (Code,name, department, number_of_students). Insert values in the table. Write a program to update number_of_studentsof “BCA Science” to 1000. 5. Write a program to insert the records into the table Employee (ID,name,salary) using PreparedStatement interface	04	

6	**Servlets and JSP :** Servlet lifecycle: Init(), Service(), doGet(), doPost(), destroy(), Form data parsing: getParameter(), getParameterValues(), JSP implicit objects, JSP tags: Directives, Declarations, Scriptlets, Expressions 1. Write a program to display Servlet Life Cycle. 2. Write a SERVLET program to display number of times user has visited the page. (use cookies) 3. Write a JSP program to check whether given number is Prime or not. (Use Include directive). 4. Write a JSP program to display the details of Player (ID, Name, TeamName) from a database on a browser using JSP. 5. Write a JSP Program to validate username and password 6. Write a JSP program to calculate factorial value for an integer number. Take the input in a HTML form.	04	
	Total Number of Sessions	24	

SEMESTER V

BCA357: DSE II Laboratory (Data mining)

Teaching Scheme: **PR:4* Hours/Week**	**Credits** **02**	**Examination Scheme:** **Continuous Evaluation : 15 Marks** **End-Semester Exam : 35 Marks**

Prerequisite Courses:

- **BCA 246 Python Programming Laboratory**

Companion Courses:

- **BCA 352 DSE II (Data Mining and Data Science)**

Course Objectives:

- To understand the basics of R programming
- To study facilities for performing data mining with R packages
- To learn python functionalities and features used for data mining
- To explore Data analysis and Data Visualization using Python

Course Outcomes: After completion of this course, students will able to

- Implement data mining tasks using R
- Use the python packages to carry out data mining tasks.
- Perform data analysis and data visualization using python packages.

Lab Course Contents

Assignment No	**Assignment Name**	**No. Of Sessions**
1	**R Programming Basics, Programs using List, Matrix, String and Factors, Program using data frame and visualization** 1. Write a R program to add, multiply and divide two vectors of integer type. (vector length should be minimum 4) 2. Write a R program to calculate the multiplication table using a function. 3. Write a script in R to create a list of employees and perform the following: Display names of employees in the list. Add an employee at the end of the list. Remove the third element of the list. 4. Write a R program to sort a list of strings in ascending and descending order. 5. Write a R program to reverse a number and also calculate the sum of digits of that number. 6. Write a R program to calculate the sum of two matrices of given size. 7. Write a R program to concatenate two given factors. 8. Write a R program to create a data frame using two given vectors and display the duplicate elements. 9. Write a R program to perform the following:	10

	Display all rows of the data set having height greater than 120. Display all rows of data set in ascending order of weight. a. (Use inbuilt data set woman) 10. Write a R program to perform the following: Display all the cars having mpg more than 20. Subset the data set by mpg column for values greater than 15.0 Display all cars having four gears. (Use inbuilt data set mtcar) 11. Write a R Program to perform the following: Create a Scattered plot to compare wind speed and temperature. Create a Scattered plot to show the relationship between ozone and wind values by giving appropriate values to colour argument. Create a Bar plot to show the ozone level for all the days having temperature > 70. (Use inbuilt datasetair quality)	
2	**Classification – Decision tree** Conversion of Categorical values in numeric format for a given dataset. Perform Classification using Decision Tree algorithm	03
3	**Association Rules and Clustering (Using inbuilt Data set)** Perform ARM using Apriori Algorithm Perform Clustering using k-means clustering algorithm	04
4	**Regression Analysis and Outlier detection** Perform Regression Analysis. Perform Linear Regression	03
5	**Python programs for Clustering** Write a python program to implement k-nearest Neighbors ML algorithm to build prediction model (Use Forge Dataset) Write a python program to implement k-means algorithms on a synthetic dataset.	04
	Total Number of Sessions	24

SEMESTER V

BCA 358: DSE III Laboratory (Operating Systems and AI)

Teaching Scheme: PR: 4* Hours/Week	Credits 02	Examination Scheme: Continuous Evaluation : 15 Marks End-Semester Exam : 35 Marks

Prerequisite Courses:

- BCA123 Operating System Concepts

Companion Courses:

- BCA 353 : DSE III (Principles of Operating Systems)
- BCA 354: SEC I (Artificial Intelligence)

Course Objectives:

- To study the process management and scheduling.
- To Study Memory Management
- To study and understand searching techniques

Course Outcomes:

After successful completion of this course, learner will be able to:

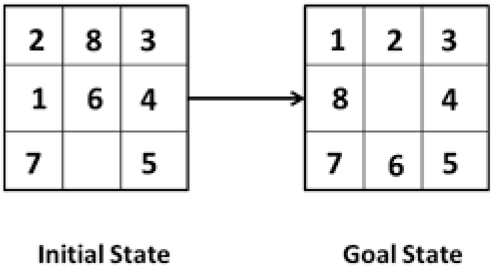
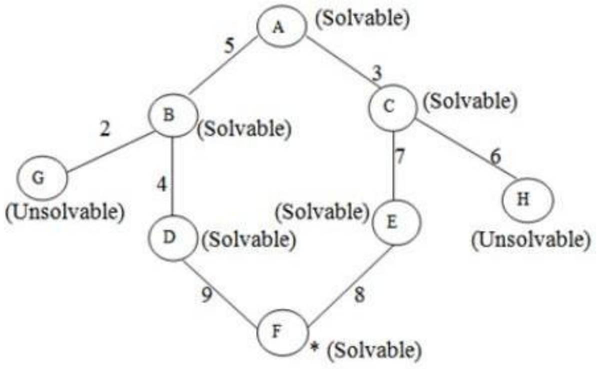
- Implement algorithms for Process scheduling and Memory management
- Describe process synchronization and multithreading
- Compare and contrast the algorithms for memory management and its allocation policies.
- Use searching algorithms
- Design a simple Expert system

Lab Course Contents

Assignment No	Assignment Name	No. Of Session
1	**Assignment 1: Operations on processes** 1. Create a child process using fork(), display parent and child process id. Child process will display the message "Hello World" and the parent process should display "Hi". 2. Creating a child process using the command exec(). Note down process ids of the parent and the child processes, check whether the control is given back to the parent after the child process terminates. Write a similar program using execv() and execvp() and observe the differences in behaviors of the commands 3. Creating a child process without terminating the parent process Write a program to create a child process using fork().The parent should goto sleep state and child process should begin its	2

	execution. In the child process, use execl() to execute the "ls" command. 4.Write a program to illustrate the concept of orphan process (Using fork() and sleep()) 5.Write a program that demonstrates the use of nice() system call. After a child process is started using fork(), assign higher priority to the child using nice() system call. 6.Write a program to find the execution time taken for execution of a given set of instructions (use clock() function)		
2	**Assignment 2: CPU Scheduling** FCFS, SJF, Priority Scheduling, Round-robin scheduling a. Write a program to simulate FCFS CPU-scheduling algorithm. Accept number of Processes as input. Also accept arrival time and CPU burst time for each process as input. The output should generate a Gantt chart, turnaround time and waiting time for each process. Also display the average turnaround time and average waiting time. b. Write a program to simulate Non-Pre-emptive Shortest Job First (SJF) CPU scheduling algorithm. Accept the number of Processes and arrival time and CPU burst time for each process as input. The output should generate a Gantt chart, turnaround time and waiting time for each process. Also display the average turnaround time and average waiting time. c. Write a program to simulate Non-Pre-emptive Priority and Round robin CPU scheduling algorithm. Accept the number of Processes and arrival time, CPU burst time and priority for each process as input. Priorities should be in High to Low order (Example 1 is High and 5 is Low). For Round robin, consider time slice is 2 units.	4	
3	**Assignment 3: Deadlock detection and avoidance** Write a program for deadlock detection considering a single instance of each resource using wait for graph. Write a program which will traverse the graph to check for cycles. If a cycle is detected, print the list of processes that are involved in the deadlock. Deadlock avoidance using Banker‟s Algorithm Write a program to implement a Banker″s Algorithm. Accept the total number of processes (n) and resource types (m) as input. Also accept the number of instances for each resource type, Allocation and Max of	4	

<table>
<tr>
<td></td>
<td>
size “n x m” as input and perform the following operations:
a) Show the contents of Available array of size “m” b) Calculate and display the contents of Need matrix of size “n x m”
Using Safety and Resource-Request algorithm perform the following operations: a) Check whether the system is in safe state or not
b) If a request of size “m” arrives from process P_i , can it be granted immediately by keeping the system in safe state? Where 0 <= i<=m-1.
2. Consider the following snapshot of the system
<table>
<tr><td>Process</td><td colspan="4">Allocation</td><td colspan="4">MAX</td><td colspan="4">Available</td></tr>
<tr><td></td><td>A</td><td>B</td><td>C</td><td>D</td><td>A</td><td>B</td><td>C</td><td>D</td><td>A</td><td>B</td><td>C</td><td>D</td></tr>
<tr><td>P0</td><td>0</td><td>0</td><td>1</td><td>2</td><td>0</td><td>0</td><td>1</td><td>2</td><td>1</td><td>5</td><td>2</td><td>0</td></tr>
<tr><td>P1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>7</td><td>5</td><td>0</td><td></td><td></td><td></td><td></td></tr>
<tr><td>P2</td><td>1</td><td>3</td><td>5</td><td>4</td><td>2</td><td>3</td><td>5</td><td>6</td><td></td><td></td><td></td><td></td></tr>
<tr><td>P3</td><td>0</td><td>6</td><td>3</td><td>2</td><td>0</td><td>6</td><td>5</td><td>2</td><td></td><td></td><td></td><td></td></tr>
<tr><td>P4</td><td>0</td><td>0</td><td>1</td><td>4</td><td>0</td><td>6</td><td>5</td><td>6</td><td></td><td></td><td></td><td></td></tr>
</table>
Calculate and display the contents of matrix Need
Using Safety and Resource-Request algorithm decide:
a) Whether the system is in safe state or not.
b)If the request from process P_4 arrives as (0, 0, 4, 1), can it be granted immediately by keeping system in safe state?
</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>
<b>Assignment 4: Page Replacement Algorithms:</b> FIFO, Optimal, LRU
a. Consider the following page reference string:
1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 2, 3, 7, 6, 3, 2, 1, 2, 3
How many page faults would occur for the following page replacement algorithms assuming four frames? All frames are initially empty.
1. LRU 2. Optimal 3. FIFO
</td>
<td>4</td>
<td></td>
</tr>
</table>

	b. Write a program to simulate demand paging using FIFO, LRU, optimal page replacement algorithms. Assume the memory of “n” frames. Show the contents of the page after every page replacement in a frame and at the end show the total number of page faults accordingly. Reference String : 3,4,5,4,3,4,7,2,4,5,6,7,2,4,6		
5	**Assignment 5: Implement A* Algorithm** **Sample Assignment** Given an initial state of a 8-puzzle problem and final state to be reached-  Initial State: 2 8 3 / 1 6 4 / 7 5 Goal State: 1 2 3 / 8 4 / 7 6 5 Find the most cost-effective path to reach the final state from initial state using A* Algorithm. Consider g(n) = Depth of node and h(n) = Number of misplaced tiles.	4	
6	**Assignment 6: Implement AO* Algorithm** **Sample Assignment** Implement AO* algorithm on the following graph.  A (Solvable) 5 3 B (Solvable) C (Solvable) 2 4 7 6 G (Unsolvable) D (Solvable) (Solvable) E H (Unsolvable) 9 8 F * (Solvable) Figure	4	
7	**Design of an Expert System for popular domains such as Medical Diagnosis**	2	
	Total Number of Sessions	**24**	

Semester VI

SEMESTER VI		
BCA 361: Android Programming		
Teaching Scheme: **Theory: 05 Hrs./Week**	**Credits** **04**	**Examination Scheme:** **Continuous Evaluation: 30 Marks** **End-Semester: 70 Marks**

Pre-requisite Course:

- BCA 351: DSE I (Programming in Java)
- BCA 356: DSE I Lab (Programming in Java)

Companion Course:

- BCA 366: DSE IV (Android Programming)

Course Objectives:

- To understand the Android Operating System
- To study Android Apps Development Cycle
- To learn to create Android Applications.

Course Outcomes:

After successful completion of this course, learner will be able to:

- Describe the process of developing mobile applications.
- Create mobile applications on the Android Platform.
- Design and implement mobile applications involving data storage in SQLite database
- Use location-based services while developing applications

	Course Contents	
Unit I	**Introduction to Android**	**06 Hrs**
	1.1. Overview 1.2. History 1.3. Features of Android 1.4. Architecture of Android • Overview of Stack • Linux Kernel • Native Libraries • Android Runtime • Application Framework • Applications 1.5. SDK Overview • Platforms • Tools – (JDK, SDK, Eclipse/Android Studio, ADT, AVD, Android Emulator), Versions 1.6. Creating your first Android Application	
Unit II	**Activities, Fragments and Intents**	**10 Hrs**
	2.1. Introduction to Activities 2.2. Activity Lifecycle 2.3. Introduction to Intents 2.4. Linking Activities using Intents 2.5. Calling built-in applications using Intents 2.6. Introduction to Fragments 2.7. Adding Fragments Dynamically 2.8. Lifecycle of Fragment 2.9. Toast	

Unit III	**Android User Interface**	**06 Hrs**
	3.1. Understanding the components of a screen • Views and View Groups • Linear Layout • Absolute Layout • Table Layout • Relative Layout • Frame Layout • Scroll Layout • Scroll View • Constraint Layout 3.2. Adapting to Display Orientation • Anchoring Views • Resizing and Repositioning 3.3. Split Screen / Multi-Screen Activities	
Unit IV	**Designing Your User Interface with Views**	**13 Hrs**
	4.1. Using Basic Views • TextView • Button, ImageButton, EditText, CheckBox • Switch, ToggleButton, RadioButton, and RadioGroup Views • ProgressBar View • AutoCompleteTextView View 4.2. Using Picker Views • TimePicker View • DatePicker View 4.3. Using List Views to Display Long Lists • ListView View • Using the Spinner View 4.4. Understanding Specialized Fragments • Using a ListFragment • Using a DialogFragment 4.5. Displaying Pictures and Menus 4.5.1. Using Image Views to Display Pictures • Gallery and ImageView views • Image Switcher • Grid View 4.5.2. Using Menus with Views • Creating the helper methods • Options Menu • Context Menu 4.6. VideoView ● Play video from URL with using VideoView ● VideoView Create ● Optimized VideoView ● Optimized VideoView in ListView	
Unit V	**Databases – SQLite, Messaging and E-mail**	**14 Hrs**
	6.1. Introduction to SQLite 6.2. SQLite Open Helper and SQLite Database 6.3. Creating , opening and closing database 6.4. Working with cursors, Insert, Update, Delete 6.5. Building and executing queries	

	6.6. SMS Messaging • Sending SMS Messages Programmatically • Getting Feedback after Sending a Message • Sending SMS Messages Using Intent • Receiving SMS Messages • Caveats and Warnings 6.7. Sending E-mail	
Unit VI	**Location-Based Services and Google Map**	**11 Hrs**
	6.1. Display Google Maps • Creating the project • Obtaining the Maps API Key • Displaying the Map • Displaying the Zoom Control • Changing Views • Navigating to a specific location • Adding Markers • Getting the location that was touched • Geocoding and Reverse Geocoding 6.2. Getting Location Data 6.3. Monitoring a Location	

Reference Books:

1) Beginning Android4 Application Development, By Wei-Meng Lee WILEY India Edition WROX Publication

2) Professional Android 4 Application Development, By Reto Meier WROX Publication

3)Head First Android Development: A Brain-Friendly Guide, By David Griffiths and Dawn Griffiths

E-Books:

1) https://enos.itcollege.ee/~jpoial/allalaadimised/reading/Android-Programming-Cookbook.pdf
2) https://www.programming-book.com/download/?file=10988
3) http://projanco.com/Library/Android%20App%20Development%20in%20Android%20Studio%20-%20Java%20plus%20Android%20edition%20for%20beginners.pdf
4) https://www.programmer-books.com/professional-android-4th-edition-pdf/

Websites:

1) The official site for Android developers - https://developer.android.com
2) https://www.tutorialspoint.com/android/index.htm
3) https://www.javatpoint.com/android-tutorial

SEMESTER VI

BCA 362: DSE-V Programming in Go

Teaching Scheme: Theory: 05 Hrs./Week	Credits 04	Examination Scheme: Continuous Evaluation: 30 Marks End-Semester : 70 Marks

Pre-requisite Course:

- BCA122: Advanced C Programming
- BCA241: Object Oriented Programming and C++

Companion Course:

- BCA367: DSE V Lab (Programming in GO and IoT)

Course Objectives:

- To study various programming constructs in GO
- To understand salient features in GO
- To know advance features in GO

Course Outcomes:

After successful completion of this course, learner will be able to:

- Describe the core features and concepts in Go
- Write simple Go programs using functions
- Apply defining methods and Go Interfaces
- Use Go routines and Channels
- Explore Go Packages

	Course Contents	
Unit I	**Introduction**	**10 Hrs**
	1.1 Go Runtime and Compilations 1.2 Keywords and Identifiers 1.3 Constants and Variables 1.4 Operators and Expressions 1.5 Local Assignments 1.6 Booleans, Numeric , Characters 1.7 Pointers and Addresses 1.8Strings 1.9 if-else, switch, for loop 1.10 Iterations 1.11Using break and continue	
Unit II	**Functions**	**10 Hrs**
	2.1 Parameters and Return Values 2.2 Call by Value and Reference 2.3 Named Return Variables 2.4 Blank Identifiers 2.5 Variable Argument Parameters 2.6 Using defer statements 2.7 Recursive Functions 2.8 Functions as Parameters	
Unit III	**Working with Data**	**8 Hrs**
	3.1 Array Literals 3.2 Multidimensional Arrays 3.3 Array Parameters 3.4 Slices and Slice Parameters	

	3.5 Multidimensional Slices 3.6 Structures and Structure Parameters	
Unit IV	**Methods and Interfaces**	**12 Hrs**
	4.1 Method Declarations 4.2 Functions vs. Methods 4.3 Pointer and Value Receivers 4.4 Method Values and Expressions 4.5 Interface Types and Values 4.6 Type Assertions and Type Switches 4.7 Method Sets with Interfaces 4.8 Embedded Interfaces 4.9 Empty Interfaces	
Unit V	**Goroutines and Channels**	**10 Hrs**
	5.1 Concurrency vs. Parallelism 5.2 Goroutine Functions and Lambdas 5.3 Wait Groups 5.4 Channels 5.5 Sending and Receiving 5.6 Unbuffered and Buffered Channels 5.7 Directional Channels 5.8 Multiplexing with select 5.9 Timers and Tickers	
Unit VI	**Packages and Files**	**10 Hrs**
	6.1 Packages and Workspaces 6.2 Exporting Package Names 6.3 Import Paths and Named Imports 6.4 Package Initializations 6.5 Blank Imports 6.6 Unit Testing with Test Functions 6.7 Table Tests and Random Tests 6.8 Benchmarking 6.9 Working with Files	
Reference Books: 1) Introducing Go, Caleb Doxey, Oreilly publication 2) Learning Go Programming: Build Scalable Next-Gen Web Application using Golang (English Edition), Shubhangi Agarwal, BPB publication		
E-Books: 1) Introducing Go By Caleb Doxey, Released January 2016Publisher(s): O'Reilly Media, Inc. ISBN: 9781491941959https://www.oreilly.com/library/view/introducing-go/9781491941997/ 2) Go Bootcamp by Matt Aimonettihttp://www.golangbootcamp.com/book		
Online Courses: Golang free online course with MOOC		

SEMESTER VI

BCA 363: DSE VI Software Project Management

Teaching Scheme:	Credits	Examination Scheme:
Theory: 05 Hrs./Week	**04**	**Continuous Evaluation: 30 Marks** **End-Semester: 70 Marks**

Pre-requisite Course:

- BCA243 Software Engineering

Companion Course:

- BCA 368 DSE VI Project Laboratory

Course Objectives:

- To understand the fundamentals of Software Project Management
- To introduce Software project planning and management tools
- To study software project scheduling and tracking
- To know the agile project management
- To learn managing people in software project

Course Outcomes:

On Successful completion of the course, learners should be able to

- Comprehend Software Project Management Concepts
- Use various tools for Software Project Management Schedule various activities in software projects
- Track a project and manage changes
- Apply Agile Project Management concepts
- Analyze staffing process for team building and decision making

	Course Contents	
Unit I	**Introduction to Software Project Management**	**10 Hrs**
	1.1 Project Definition 1.2 Project versus Flow type work 1.3 Project Lifecycle 1.4 Processes and Knowledge Areas in Project Management (PM) 1.5 Build or Buy decision, 1.6 Work Breakdown Structure (WBS) and its types 1.7 Introduction to PMBOK, 1.8 Program and Portfolio Management	
Unit II	**Project Planning and Project Management Tools**	**10 Hrs**
	2.1 Project Planning 2.2 Steps for Project Planning, 2.3 PERT and Gantt Charts 2.4 Introduction to Project Management using Gantt Project 2.5 Objectives of Activity planning 2.6 Project Schedules, Activities, Sequencing and Scheduling 2.7 Network Planning Models, 2.8 Formulating Network Model.	
Unit III	**Activity based Scheduling**	**10 Hrs**
	3.1 Introduction 3.2 Objectives of Activity Planning 3.3 Activity relationships (FS, SF, SS, FF)	

	3.4 Forward Pass technique 3.5 Backward Pass techniques 3.6 Critical Path concept and remedies	
Unit IV	**Project Tracking and Control**	**10 Hrs**
	4.1 Introduction 4.2 Collection of Project data 4.3 Visualizing progress 4.4 Cost monitoring 4.5 Earned Value Analysis 4.6 Project tracking 4.7 Change Control 4.8 Software Configuration Management 4.9 Managing contracts and Contract Management.	
Unit V	**Agile Project Management**	**10 Hrs**
	5.1 Predictive versus Empirical Management 5.2 Comparison between Non-Agile and Agile Project 5.3 Three stages of Agile Project 5.4 Estimation 5.5 Scope Management 5.6 Roles and Responsibilities 5.7 Scheduling and Tracking	
Unit VI	**Staffing in Software Projects**	**10 Hrs**
	6.1 Managing People Organizational behavior 6.2 Best methods of Staff Selection 6.3 Motivation 6.4 The Oldham Hackman job characteristic Model 6.5 Stress, Health and Safety 6.6 Ethical and Professional Concerns 6.7 Working in Teams, Decision Making 6.8 Organizational structures, Dispersed and Virtual Teams 6.9 Communications Genres and Communication Plans.	
Text Book: **1.** Bob Hughes, Mike Cotterell and Rajib Mall, “Software Project Management”, Sixth Edition, Tata McGraw Hill, New Delhi, 2012. **2.** Robert K. Wysocki, “Effective Software Project Management”, Wiley Publication,2011		
Reference Books: 1. Ken Schwaber, “Agile Project Management”, MicrosoftPress,2004 2. Walker Royce, “Software Project Management”, Addison-Wesley,1998. 3. Jalote Pankaj, “Software Project Management in Practice”, Addison-WesleyProfessional,2002 4. PMBOK Guide		
e-books: ● https://www.kornev-online.net/ITIL/Mcgraw.Hill.Software_Project_Management_2nd_Edition.pdf		

SEMESTER VI		
BCA364: SEC III Management Information System		
Teaching Scheme: **Theory: 3 Hrs./Week**	**Credits** **02**	**Examination Scheme:** **Continuous Evaluation: 15 Marks** **End-Semester : 35 Marks**

Pre-requisite Course:

- BCA243 Software Engineering

Companion Course: None

Course Objectives:

- To know the role of information technology and decision support systems in business model.
- To learn the fundamental principles of information systems
- To understand the principles and techniques used for management and decision making
- To explore various applications of MIS

Course Outcomes:

After successful completion of this course, learners will able to

- Describe MIS, BPR, EMS
- Compare MIS with BPR, DSS and EMS
- Identify various ERP modules for a given application
- List the applications of MIS in Manufacturing and service sectors

	Course Contents	
Unit I	**Introduction to MIS**	**3Hrs**
	1.1 Introduction to MIS 1.2 Definition 1.3 Role of MIS 1.4 Impact of MIS 1.5 MIS as a control system. 1.6 MIS: a support to the management.	
Unit II	**Decision Making and Information**	**5 Hrs**
	2.1 Decision Making Process and models 2.2 Decision Analysis by Analytical Modeling 2.3 Behavioral Concepts in Decision-Making 2.4 Organizational Decision-Making 2.5 Information Management: Classification of Information 2.6 Methods of Data and information collection	
Unit III	**Business Process Re-engineering (BPR)**	**5 Hrs**
	3.1 Introduction to BPR and Business process 3.2 Process Model of the organization 3.3 Value stream Model of Organization 3.4 Relevance of Information Technology 3.5 MIS and BPR	
Unit IV	**Enterprise Management Systems (EMS)**	**7Hrs**
	4.1 Introduction to EMS and ERP 4.2 ERP Model, Modules, Benefits of ERP	

	4.3 ERP Product Evaluation and Implementation 4.4 Introduction to Supply Chain management (SCM) and Customer Relationship Management) 4.5 EMS and MIS	
Unit V	**Decision Support Systems & Knowledge Management**	**6 Hrs**
	5.1 Introduction to Decision Support Systems (DSS) 5.2 Group Decision Support Systems (GDSS) 5.3 Business Intelligence and Analytics 5.4 Executive Information Systems & Executive Support Systems 5.5 Introduction to Knowledge Management 5.6 Knowledge Management Systems 5.7 Knowledge Based Expert Systems (KBES)	
Unit VI	**Applications of MIS in Manufacturing and Service Sectors**	**4Hrs**
	6.1 MIS for Financial and marketing Management 6.2 Introduction to service and service sector 6.3 Service Process Cycle 6.4 Customer service Design and Service Management System	
	Reference Books • Jawadekar, W.S., “Management Information Systems”, Tata McGraw Hill Private Limited, New Delhi, 2009. • Kenneth C. Laudon and Jane P. Laudon: “Management Information Systems” 9/e, Pearson Education, New Delhi. • Goyal, D.P.: “Management Information System”, MACMILLAN India Limited, New Delhi, 2008. • Mahadeo Jaiswal, Monika Mital: “Management Information System”, Oxford University Press, New Delhi, 2008. • Murthy C.S.V.: “Management Information System”, Himalaya Publications, New Delhi, 2008.	

SEMESTER VI

BCA365: SEC IV Internet of Things (IoT)

Teaching Scheme: Theory: 3 Hrs./Week	Credits 02	Examination Scheme: Continuous Evaluation: 15 Marks End-Semester : 35 Marks

Pre-requisite Course:

- BCA121: Computer Organization
- BCA125: Computer Organization Laboratory
- BCA233: Computer Networks
- BCA355: SEC II (Cloud Computing)

Companion Course:

- BCA367 DSE V Laboratory (Programming in GO and IoT)

Course Objectives:

- To understand fundamentals of Internet of Things (IoT) and Embedded Systems
- To know methodologies for IoT application development
- To study the IoT protocols, cloud platforms and security issues in IoT
- To learn real world application scenarios of IoT along with its societal and economic impact

Course Outcomes:

On successful completion of the course, learners should be able to

- Define Embedded Systems and the Internet of Things
- Apply enabling technologies for developing IoT systems
- Design simple IoT applications
- Analyze protocols for communication among IoT devices
- Describe cloud-based IoT systems
- Comprehend security issues in IoT applications

	Course Contents	
Unit I	**Introduction to Embedded Systems**	**5 Hrs**
	1.1 Definition, Characteristics of Embedded System, 1.2 Real time systems, Real time tasks 1.3 Processor basics: General Processors in Computer Vs Embedded Processors, Micro controllers and Properties, Components of Microcontrollers, System-On- Chip and its examples 1.4 Components of Embedded Systems, 1.5 Introduction to embedded processor.	
Unit II	**Internet of Things : Concepts**	**5 Hrs**
	2.1 Definition, Characteristics of IoT, Trends in Adoption of IoT, 2.2 IoT Devices, IoT Devices Vs Computers, 2.3 Basic Building Blocks. 2.4 Physical Design of IoT: - Things in IoT, Interoperability of IoT Devices, Sensors and Actuators, Need of Analog/Digital Conversion. 2.5 Logical Design of IoT:- IoT functional blocks, IoT Enabling technologies, IoT levels and deployment templates, Applications of	

	IoT	
Unit III	**Introduction to IoT Design Methodology**	**5 Hrs**
	3.1 Design Steps 3.2 Basics of IoT Networking, Networking Components, Internet Structure 3.3 IoT Communication Models and IoT Communication APIs, 3.4 Sensor Networks 3.5 Four pillars of IoT: M2M, SCADA,WSN, RFID	
Unit IV	**Introduction to IoT Protocols**	**5 Hrs**
	4.1 Protocol Standardization for IoT 4.2 M2M and WSN Protocols 4.3 RFID Protocol, 4.4 Modbus Protocol, Zigbee Architecture. 4.5 IP based Protocols: MQTT (Secure), 6LoWPAN, LoRa.	
Unit V	**Cloud Platforms for IoT**	**5 Hrs**
	5.1 Introduction to Cloud Storage Models, Communication API 5.2 Cloud for IoT 5.3 Introduction to Amazon Web Services for IoT and SkyNet IoT 5.4 Messaging Platform 5.5 Introduction to RESTful Web Services -GRPC,SOAP.	
Unit VI	**Security in IoT**	**5 Hrs**
	6.1 Introduction, Vulnerabilities of IoT, Security Requirements 6.2 Challengesfor Secure IoT, Threat Modeling 6.3 Key elements of IoT Security: Identity establishment, Access control, Data and message security, Non-repudiation and availability 6.4 Security model for IoT 6.5 Challenges in designing IOT applications, Introduction to Lightweight cryptography	

Reference Books:

1. Arshdeep Bahga,Vijay Madisetti, “Internet of Things– A hands- on approach”, Universities Press, ISBN: 0: 0996025510, 13:978-996025515
2. Olivier Hersent, David Boswarthick, Omar Elloumi, “The Internet of Things: Key Applications and Protocols”, 2nd Edition, Wiley Publication,ISBN:978-1-119-99435-0
3. Dawoud Shenouda, Peter Dawoud, “Microcontroller and Smart Home Networks”, ISBN: 9788770221566, e-ISBN: 9788770221559
4. Charles Crowell, “Internet of Things for Beginners: An Easy-to-Understand Introduction to IoT”, ISBN-13 : 979-8613100194
5. David Hanes, Gonzalo Salgueiro, Robert Barton, Jerome Henry, “IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things”, Cisco Press, ISBN-13: 978-1-58714-456-1 ISBN-10: 1-58714-456-5

SEMESTER VI

BCA366: DSE IV Laboratory (Android Programming)

Teaching Scheme:	**Credit**	**Examination Scheme:**
PR: 4* Hours/Week	**02**	**Continuous Evaluation : 15 Marks** **End-Semester Exam : 35 Marks**

Prerequisite Courses:

- BCA 351: DSE I (Programming in Java)
- BCA 356: DSE I Lab (Programming in Java)

Companion Courses:

- BCA – 361 Android Programming

Course Objectives:

- To understand the Android Operating System and
- To study Android Apps Development Cycle
- To learn to create Android Applications.

Course Outcomes:

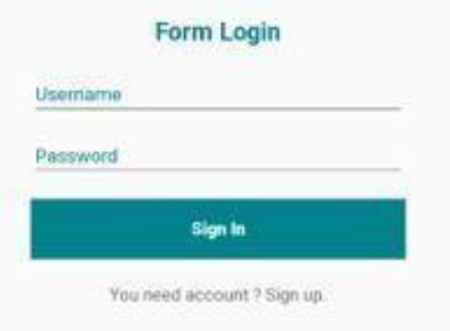
After completion of this course, students will able to

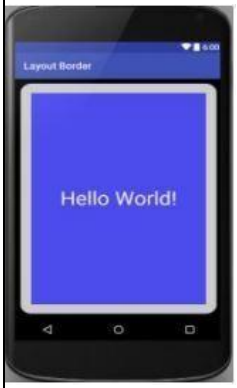
- Describe the process of developing mobile applications.
- Create mobile applications on the Android Platform.
- Design and implement mobile applications involving data storage in SQLite database
- Use location-based services while developing applications

Lab Course Contents

Assignment No	**Assignment Name**	**No. Of Sessions**
1	Introduction to Android	02
2	Activities, Fragments and Intents	04
3	Android User Interface	04
4	Designing User Interface with Views	05
5	Databases-SQLite, Messaging and E-mail	05
6	Location-Based Services and Google Map	04
	Total Number of Sessions	24

Sample Assignments Programs

Assignment 1	Introduction to Android Install Android Studio and build simple Hello World application.
Assignment 2	Activities, Fragments and Intents Create Simple application to display details of selected list item on Second Activity (use Fragmentation). Create simple application with Login Screen. On successful login, gives message go to next Activity (Without Using Database).  Form Login Username Password Sign In You need account ? Sign up

	Create a Simple Application Which Send —Hello‖ message from one activity to another with help of Button (Use Intent).
Assignment 3	Android User Interface Design following-add a border to an Android Layout 
Assignment 4	Designing Your User Interface with Views Construct an app that toggles a light bulb on and off when the user clicks on toggle button. Create gallery application to display all images date wise (Use Grid View).
Assignment 5	Databases-SQLite, Messaging and E-mail Construct a simple notes list that lets the user add new notes but not edit them. Demonstrates the basics of ListActivity. Use a SQLite database to store the notes. Create application to send and receive messages. Create application to send email with validation.
Assignment 6	Location-Based Services and Google Map Write a program to track android device using Google Maps. Write a program to draw path along a route in Google map.

Suggested Online Courses:

Learners may study following online courses

- Advanced App Development in Android Capstone (Coursera)
- Android Basics: Multi screen Apps (Udacity)
- Android Basics: Networking (Udacity)
- Firebase in a Weekend: Android (Udacity)
- Android App Development for Beginners (edX)
- Introduction to Mobile Application Development using Android (edX)

SEMESTER VI

BCA367: DSE V Laboratory (Programming in GO and IoT)

Teaching Scheme:	Credits	Examination Scheme:
PR: 4* Hours/Week	02	Continuous Evaluation : 15 Marks End-Semester Exam : 35 Marks

Prerequisite Courses:

- Knowledge of Programming Languages C, C++, Java, Python

Companion Courses:

- BCA362 Programming in GO
- BCA365 Internet of Things (IoT)

Course Objectives:

- To introduce essential programming features in GO
- To become familiar with programming techniques in GO
- To understand the technique of building Packages and File handling
- To learn developing simple IoT applications

Course Outcomes:

On completion of this course, students will be able to:

- Write programs using features supported in GO
- Handle errors and utilize Goroutines and Channels
- Write programs on File handling
- Compare and contrast features of GO with other object oriented languages
- Design Simple IoT application

Lab Course Contents

Assignment No	Assignment Name	No. Of Sessions
Programming in GO		
1	Introduction to Go Programming	02
2	Functions	03
3	Working with data	03
4	Methods and Interfaces	04
5	Go routines and channels	03
6	Packages and files	03
Internet of Things (IoT)		
7	Interfacing Raspberry-Pi/ Adriano with IR sensor	02
8	Interfacing Raspberry-Pi/ Beagle board with temperature sensor	02
9	Interfacing Raspberry-Pi/ Beagleboard with camera	02
		24

Sample Assignments Programs

Assignment 1

SET A

1. WAP in go language to print Student name, rollno, division and college name
2. WAP in go language to print whether number is even or odd.
3. WAP in go language to swap the number without temporary variable.

SET B

1. WAP in go to print table of given number.
2. WAP in go language to print PASCALS triangle.

	3. WAP in go language to print Fibonacci series of n terms. **SET C** 1. WAP in go language to concatenate two strings using pointers. 2. WAP in go language to accept two strings and compare them. 3. WAP in go language to accept user choice and print answer of using arithmetical operators.
Assignment 2	**SET A** 1. WAP in go language to print addition of two number using function. 2. WAP in go language to print recursive sum of digits of given number. 3. WAP in go language using function to check whether accepts number is palindrome or not. **SET B** 1. WAP in go language to swap two numbers using call by reference concept. 2. WAP in go language to demonstrate use of names returns variables. 3. WAP in go language to show the compiler throws an error if a variable is declared but not used. **SET C** 1. WAP in go language to illustrate the concept of call by value. 2. WAP in go language to create a file and write hello world in it and close the file by using defer statement. 3. WAP in go language to illustrate the concept of returning multiple values from a function
Assignment 3	**SET A** 1. WAP in go language to find the largest and smallest number in an array. 2. WAP in go language to accept the book details such as BookID, Title, Author, Price. Read and display the details of n number of books. 3. WAP in go language to Initialize a Slice using Multi-Line Syntax and display **SET B** 1. WAP in go language to create and print multidimensional Slice. 2. WAP in go language to sort array elements in ascending order. **3.** WAP in go language to accept n student details like roll_no, stud_name, mark1,mark2, mark3. Calculate the total and average of marks using structure. **SET C** 1. WAP in go language to accept two matrices and display it's multiplication. 2. WAP in go language to accept n records of employee information (eno,ename,salary) and display record of employees having maximum salary. 3. WAP in go language to demonstrate working of slices (like append, remove, copy etc.)
Assignment 4	**SET A** 1. WAP in go language to create an interface shape that includes area and perimeter. Implements these methods in circle and rectangle type. 2. WAP in go language to print multiplication of two numbers using method. 3. WAP in go language to create structure author. Write a method show() whose receiver is struct author. **SET B** 1. WAP in go language to create structure student. Write a method show() whose receiver is a pointer of struct student. 2. WAP in go language to demonstrate working type switch in interface. 3. WAP in go language to copy all elements of one array into another using method. **SET C**

	1. WAP in go language to create an interface and display it‟s values with the help of type assertion. 2. WAP in go language to store n student information(rollno, name, percentage) and write a method to display student information in descending order of percentage. 3. WAP in go language to demonstrate working embedded interfaces.
Assignment 5	**SET A** 1.WAP in Go to illustrate how to create an anonymous Goroutine. 2.WAP in Go how to create channel and illustrate how to close a channel using for range loop and close function. 3. WAP in Go maingoroutine computes the 10th fibonacci number using an inefficient recursive algorithm. **SET B** 1. WAP in GO prints out the numbers from 0 to 10, waiting between 0 and 250 ms after each one using delay function . 2.WAP in GO prints “from 1” every 2 seconds and “from 2” every 3 seconds. select picks the first channel that is ready and receives from it. If more than one of the channels are ready then it randomly picks which one to receive from. If none of the channels are ready, the statement blocks until one becomes available. 3.WAP in Go to illustrate channels buffering. **SET C** 1 WAP in GO . Lambda function handler using structured types you can also pass in structured events to your function handler: 2. WAP in GO implement multiple goroutine function and schedule is determined by the scheduler. 3. WAP in Go program such that the squares are calculated in a separate Goroutine, cubes in another Goroutine and the final summation happens in the main Goroutine
Assignment 6	**SET A** 1. WAP to create student struct with student name and marks and sort it based on student marks using sort package 2. WAP in Go language using user defined package calculator that performs one calculator operation as per the user's choice. 3. WAP in Go language to create an user defined package to find out the area of a rectangle. **SET B** 1. WAP in Go language to add two integers and write code for unit test to test this code. 2. WAP in Go language to subtract two integers and write code for table test to test this code. 3. Write a function in Go language to find square of a number and write a benchmark for it. **SET C** 1. WAP in Go language to read a XML file into structure and display structure 2. WAP in Go language to print file information. 3. WAP in Go language to add or append content at the end of text file.
Assignment 7	Understanding the connectivity of Raspberry-Pi/ Adriano with IR sensor. Write a program to detect obstacle and notify a user using LEDs

Assignment 8	Understanding the connectivity of Raspberry-Pi/ Beagle board Circuit with temperature sensor. Write a program to sense temperature. If temperature crosses a threshold value, generate alerts using LEDs.
Assignment 9	Understanding and connectivity of Raspberry-Pi/Beagle board with camera. Write a program to capture and store the image.

Semester VI

BCA368: DSE VI Project Laboratory

Teaching Scheme:	Credits	Examination Scheme:
PR: 4* Hours/Week	**02**	**Continuous Evaluation : 15 Marks** **End-Semester Exam : 35 Marks**

Prerequisite Courses:

- BCA243 Software Engineering

Companion Course: BCA363 DSE VI (Software Project Management)

Course Objectives:

1. To understand concepts of Project Management
2. To know how various tools for development and management of software projects are used to carry out various tasks involved
3. To learn the importance of project documentation

Course Outcomes:

After completion of this course, learners will able to

1. Demonstrate a sound technical knowledge of selected project topic.
2. Apply techniques for project management
3. Create various documents used during the development of the project and a project report

Project Implementation Guidelines:

Sr. No.	Guidelines
1	Students shall choose any topic for project work in consultation with project guide, Project In-charge and head of the department.
2	The students shall work on a Project in a group of not more than three students.
3	Students are expected to work on the chosen project during the entire semester.
4	Students shall undertake application oriented/web-based/database-oriented/research based work.
5	Students shall successfully implement the chosen work. Only a hypothetical / theoretical study shall not be accepted.
6	Students shall choose any appropriate programming language/ platform, computational techniques and tools in consultation with the guide, In-charge and the head of the department.
7	The faculty members from affiliated college shall act as a project guide for each project group with equal distribution of groups amongst each eligible faculty.
8	The guide shall track and monitor the project progress on a weekly basis by considering the workload of 4 laboratory hours per week.
9	The project work shall be evaluated based on the novelty of the topic, scope of the work, relevance to computer science, adoption of emerging techniques/technologies and its real-world application etc.
10	Students shall prepare a project report with the following contents: a) Title Page b) Certificate c) Index Page detailing description of the following with their sub sections:- - Title: A suitable title giving the idea about what work is proposed. - Introduction: An introduction to the topic giving proper background of the topic. - Requirement Specification: Specify Software/hardware/data requirements. - System Design details :Methodology/Architecture/UML/DFD/Algorithms/protocols used(whichever is applicable) - System Implementation: Code implementation

	- Results: Test Cases/Tables/Figures/Graphs/Screen shots/Reports etc. - Conclusion and Future Scope: Specify the Final conclusion and future scope - References: Books, web links, research articles etc.
11	The Project report should be prepared in a spiral bound form with adequate number of copies. Copy shall be submitted to the guide and college for the records.
12	The Project work and report shall be certified by the concerned Project guide and Head of the department.
13	Students shall make a presentation of working project and will be evaluated as per the Project evaluation scheme as detailed below
Assignments using Gantt Project tools • Students are advised to carry out the following assignments w.r.t. their chosen project topics	
1	**Create Project Plan** • Specify project name and start and finish dates. • Identify and define project tasks. • Define duration for each project task. • Define milestones in the plan • Define dependency between tasks • Define project calendar. • Define project resources and specify resource type • Assign resources against each task and baseline the project plan
2	**Execute and Monitor Project Plan** • Update %Complete with current task status. • Review the status of each task. • Compare Planned vs Actual Status • Review the status of Critical Path • Review resources assignation status
3	**Generate Dashboard and Reports** • **Dashboard** ○ Project Overview ○ Cost Overview ○ Upcoming Tasks • **Resource Reports** ○ Over-allocated Resources ○ Resource Overview • Cost Reports ○ Earned Value Report ○ Resource Cost Overview ○ Task-Cost Overview • Progress Reports ○ Critical Tasks ○ Milestone Report ○ Slipping Tasks
Evaluation Scheme	

I. Continuous Evaluation, Progress Report: 15 marks

II. End Semester Examination in the form of presentation/demonstration and viva: 35 marks

Description	**Marks**
Presentation & Project Report	15
Demonstration of the Project	15
Viva	05
Total	**35**

Note: Submission of Certified Project Report is mandatory for appearing the Practical Examination (Project).

Equivalence for TYBCA syllabus (2016) with TYBCA syllabus (2019)			
Pattern 2016		**Pattern 2019**	
CourseCode	**Course Name**	**Course Code**	**Course Name**
Semester V			
BCA501	Java Programming	BCA351	DSE I Programming in Java
BCA502	Advanced Web Technology	BCA242	Web Technology
BCA503	Software Quality Assurance	**To be continued**	
BCA504	Operating System	BCA353	DSE III Principles of Operating Systems
BCA505	Lab I (Core Java)	BCA 356	DSE I Lab (Programming in Java)
BCA506	Lab II (Adv. Web Technology)	BCA245	Web Technology Laboratory
BCA507	Soft Computing (2 credits)	**To be continued**	
Semester VI			
BCA601	Android Programming	BCA361	DSE IV Android Programming
BCA602	Python Programming	**To be continued**	
BCA603	Recent Trends in -IT (Internet of Things)	**To be continued**	
BCA604	Data Analytics	**To be continued**	
BCA605	Android Programming Lab	BCA366	DSE IV Laboratory (Android Programming)
BCA606	Python Lab	BCA246	Python Programming Lab
BCA607	Introduction to Green Computing	**To be continued**	

Details of Subcommittee for revision of syllabus

Advisors – All Members of BOS in Computer Applications

Prof. Dr. R M Sonar, Dr. M N Shelar, Dr. Pallavi Bulakh, Dr. Razak Sayyad

Special Invitees

Dr. Abhijat Vichare, Dr. Manisha Bharambe, Dr. Anjali Sardesai, Prof. Arun Gangarde, Prof. Rahul Patil,

TYBCA Semester-V

Name of the Coordinator: **Dr. Pallawi U Bulakh**

Name of the Institute: PES Modern College of Arts, Science and Commerce, Ganeshkhind, Pune

Course wise Team Details

Subject Code	Name of the Course	Name of the Team Leader	Names of Team Members
BCA 351	DSE I (Programming in Java)	Aparna Gohad	1. SonaliShivarkar 2.Surekha Jadhav
BCA 352	DSE II (Data Mining and Data Science)	Dr.DipaliMeher	1. Sanjay Wani 2. VidyaBankar
BCA 353	DSE III (Principles of Operating Systems)	Swati Jadhav	1. Suvarna Patil 2. NiketTajne
BCA 356	DSE I Laboratory (Programming in Java)	VandanaNemane.	2.Sheetal Patil 3. DeepashreeMehendale
BCA 357	DSE II Laboratory (Data Mining)	Dr.DipaliMeher	1. KavitaKhoje 2.SonaliNemade
BCA 358	DSE III Laboratory (Operating Systems and AI)	Dr. Manisha Bharambe	1.Mayuri Dasri 2. Sampada Vaishampayan
BCA 354	SEC I (Artificial Intelligence)	CharushilaPatil	1. MeenalJabde 2.Rajesh Dhumal 3. MadhuriKhadtare
BCA 355	SEC II (Cloud Computing)	SuvarnaPardeshi	1. Sachin Mhaske 2. Ambike Satish

TYBCA Semester-VI

Name of the Coordinator: **Dr. Madhukar Shelar**
Name of the Institute: KRT Arts, BH Commerce and AM Science (KTHM) College, Nashik

Course wise Team Details

Subject Code	Name of the Course	Name of the Team Leader	Names of Team Members
BCA 361	DSE IV – Android Programming	Kamil Khan	1. Satish Mulgi 2.Vrushali Shinde
BCA 362	DSE V-Programming in Go	Dr. Kalyani Salla	1. Dr.DipaliMeher. 2.Thorat Surekha
BCA 363	DSE VI-(Software Project Management)	KishorDhane	1. Rupali Jadhav 2.Manisha Jagdale 3.Nagesh Dhyatonde
BCA 366	DSE IV Laboratory (Android Programming)	MohsinTamboli	1. RajashreeNehe 2. Deepak Kumbhar
BCA 367	DSE V Laboratory (Programming in GO and IoT)	Dr. Kalyani Salla	1. Neeta Nadgude 2. SonaliPowar
BCA 368	DSE VI Project Laboratory	Dr.ArunGangarde	1. BhushanNikam 2. DeepaliJagdale 3.Priyamvada Patil
BCA 364	SEC III Management Information System	Rahul Patil	1.Deepak Derle 2. SudarshanLakhdive 3. Abhishek Awate
BCA 365	SEC IV Internet of Things (IoT)	Dr. A.B. Nimbalkar	1.Veena Gandhi 2.Rasika Rahalkar 3. Veena Gandhi

Acknowledgment

On behalf of Board of Studies in Computer Applications, it is a great pleasure and honor for me to present the syllabi for TYBCA (Third Year Computer Applications - 2019 Pattern).

At the outset I am thankful to Hon. Vice Chancellor Prof. Dr. Nitin Karmalkar, Pro-Vice Chancellor Prof. Dr. N S Umrani and Hon. Dean, Science and Technology, Prof. Dr. M G Chaskar, SPPU, Pune for their guidance and motivation.

I am thankful each and every faculty members from all affiliated colleges, each member of the subcommittee, all Heads of the departments and members of the BoS Prof. Dr. R M Sonar, Dr. M N Shelar, Dr. Pallavi Bulakh and Dr. Razak Sayyad for their help and support in preparation of this draft. Special thanks to all ex-BoS members, Dr. Anjali Sardesai, Prof. Arun Gangarde, Prof. R A Patil and Mr. Abhijeet Vichare for their inputs. I thank Prof. Dr. Manisha Bharambe for her suggestions.

Regards,

Dr. Shirish S Sane
Chairman, Board of Studies (Computer Applications), SPPU, Pune