Khandesh College Education Society's

Moolji Jaitha College, Jalgaon

An "Autonomous College" Affiliated to
KBC North Maharashtra University, Jalgaon

KNOWLEDGE IS POWER

ESTD. 1945

SYLLABUS

Computer Science

T.Y.B. Sc.

(Semester V & VI)

Under Choice Based Credit System (CBCS)

[w. e. f. Academic Year: 2021-22]

Department of Computer Science
M.J.College
Jalgaon

T.Y.B.Sc. Computer Science (CBCS pattern)

Program Specific Outcomes (PSO):

- T.Y. B.Sc. (Computer Science) graduates will have basic and applied knowledge in the following core knowledge areas
  - Data Structures
  - Programming Languages like R and Python
  - Databases
  - Software Engineering and Development
  - Data Science
  - Scripting Languages
  - Cloud computing
- They can further continue their education as PG and then Ph.D.
- After successful completion of the program, students will acquire laboratory and transferable skills which will help them to boost their career.
- Students can apply problem-solving skills and the knowledge of computer science to solve real world problems.
- Develop technical project reports and present them orally among the users

Learning Objectives:

- To acquaint the students with various disciplines of Computer Science.
- To articulate foundation and pillar level knowledge of Computer Science for the beneficiaries to apply them for advanced real life project development.
- To develop practical skills with a sound theoretical background.
- To apply the knowledge gained for higher education, research and profession of their choice.
- To analyse their interests among the various disciplines and implement them in their professional endeavours

Exam Pattern:

- Each theory and practical course will be of 50 marks comprising of 10 marks internal and 40 marks external examination.

External Theory Examination (40 marks):

- External examination will be of two hours duration for each theory course. There shall be 4 questions each carrying equal marks (10 marks each) while the tentative pattern of question papers shall be as follows;
  - Q.1) A) Attempt any Two — 6 Marks
    B) Attempt any Two — 4 Marks
  - Q.2 and Q.3 Same as Q.1
  - Q.4) Attempt any Two — 10 Marks

External Practical Examination (40 marks):

- Practical examination shall be conducted by the respective department at the end of the semester. Practical examination will be of minimum 3 hours duration and shall be conducted as per schedule. There shall be 05 marks for journal, 10 marks for *viva-voce*. Certified journal is compulsory to appear for practical examination.

Internal Theory/ Practical Examination (10 marks):

- Internal theory assessment of the student by respective teacher will be comprehensive and continuous, based on written test/ assignment. The written test may comprise of both objective and subjective type questions.
- Internal practical examination should be conducted by respective department as per schedule given. For internal practical examination student should perform at least one major and one minor experiment and should have completed journal.

Structure of T.Y.B.Sc. (Computer Science) Curriculum Semester V

Discipline	Course Type	Course Code	Course Title	Credits	Hours/ Week (Clock Hours)	Total Teaching hours	Marks	
							Int	Ext
DSC	Core I	CS-351	Systems Programming	3	3	45	10	40
	Core II	CS-352	Computer Graphics	3	3	45	10	40
	Core III	CS-353	Software Engineering	3	3	45	10	40
	Core IV	CS-354	Data Science -I	3	3	45	10	40
	Core V	CS-355	Programming in PYTHON-I	3	3	45	10	40
	Core VI	CS-356	'R' Programming	3	3	45	10	40
SEC	Skill Based	CS-350	Cloud Computing -I	2	2	30	10	40
DSC	Core (Practical)	CS-357	Practical Course based on CS-351 and CS-352	2	4 / batch	60	10	40
		CS-358	Practical Course based on CS-355	2	4 / batch	60	10	40
		CS-359	Practical Course based on CS-356	2	4 / batch	60	10	40

Structure of T.Y.B.Sc. (Computer Science) Curriculum Semester VI

Discipline	Course Type	Course Code	Course Title	Credits	Hours/ Week (Clock Hours)	Total Teaching hours	Marks	
							Int	Ext
DSC	Core I	CS-361	Operating System	3	3	45	10	40
	Core II	CS-362	Computer Network	3	3	45	10	40
	Core III	CS-363	Theoretical Computer Science	3	3	45	10	40
	Core IV	CS-364	Data Science -II	3	3	45	10	40
	Core V	CS-365	Programming in PYTHON-II	3	3	45	10	40
	Core VI	CS-366	PHP Programming	3	3	45	10	40
SEC	Skill Based	CS-360	Cloud Computing -II	2	2	30	10	40
DSC	Core (Practical)	CS-367	Practical Course based on CS-361 and CS-366	2	4 / batch	60	10	40
		CS-368	Practical Course based on CS-365	2	4 / batch	60	10	40
		CS-369	Project work	2	4 / batch	60	10	40

DSC: Discipline Specific Core Courses/Core Practical; **SEC:** Skill Enhancement Course;
Int : Internal examination; **Ext :** External examination

T.Y. B.Sc. (Computer Science): Semester-V
Discipline Specific Core (DSC) Course
CS-351: Systems Programming

Total Hours: 45 **Credits: 3**

Course Objectives:

- To make students understand use and development of software tools.
- To make students understand the design structure of Assembler and macro preprocessor
- To make students understand the design structure of compiler
- To make students understand the functions of linkers and loaders

Course Outcomes:

Students will be able to:

- understand details about system software
- develop basic system program like editors, lexical analyzers etc.
- understand language processing activities- functions of translators, loader and linkers

Unit-I: Introduction **(10 h)**

- Types of program – System program and Application program
- Difference between system programming and application programming.
- Goal of system software 1.4components of system software
- View of system software
- What is a Software Tools?
- Software Tools for Program Developments
- Editors
- Debug Monitors
- Programming Environments

Unit-II: Overview of Language Processors **(5 h)**

- Programming Languages and Language Processors
- Language Processing Activities
- Fundamentals of Language Processing

Unit-III: Assembler **(10 h)**

- Definition.
- Features of assembly language, advantages
- Statement format, types of statements
- Constants and Literals.
- Advanced assembler directives
- Design of assembler – Analysis Phase and Synthesis Phase.
- Overview of assembly process
- Pass Structure of Assembler – One pass, two pass assembler.
- Problems of One-pass assembler
- Design of Two-pass Assembler

Unit-IV: Macro and Macro Preprocessor **(5 h)**

- Macro Definition and Call
- Macro Expansion
- Nested Macro Calls
- Tables used in Macro
- Advanced Macro Facilities
- Design of Macro Preprocessor

Unit-V: Compiler **(10 h)**

- What is Compiler?
- Scanning and Parsing
- Programming Language Grammars
- Scanning
- Parsing
- Language Processors Development Tools

Unit-VI: Linkers and Loaders **(5 h)**

- Introduction
- Relocation and Linking Concepts
- Self Relocating Programs
- Linking for Overlays
- Dynamic Linking
- Loaders

References:

- Dhamdhere D. M., (2011), Systems Programming, 1st edition, Tata McGraw-Hill Education, New Delhi
- Dhamdhere D. M., (1999), Systems Programming and Operating System, 2nd edition, Tata McGraw-Hill Education, New Delhi
- Donovan J. (2017), "System programming.", 1st edition, Tata McGraw-Hill Education, New Delhi

Teaching methods:

- Classroom teaching, Lecture method, ICT enabled teaching

T.Y. B.Sc. (Computer Science): Semester-V
Discipline Specific Core (DSC) Course
CS-352: Computer Graphics

Total Hours: 45 **Credits: 3**

Course Objectives:

- To study how graphics objects are represented in Computer.
- To study how graphics system in a computer supports presentation of graphics information
- To study how to manipulate graphics object by applying different transformations
- To provide the programmer's perspective of working of computer graphics
- To study how interaction is handled in a graphics system

Course Outcome:

Students will be able to:

- differentiate between interactive and non-interactive graphics.
- study line Drawing and Circle Drawing techniques and algorithms.
- perform 2D and 3D transformation on different images.
- understand working of 2D and 3D clipping and windowing.
- understand raster graphics and hidden surface elimination.

Unit-I: Introduction to Graphics **(8 h)**

- The origin of computer graphics
- Application of Computer Graphics
- Definitions: Pixel, Resolution, Aspect Ratio, Interactive, Non interactive graphics, Active graphics, Passive graphics
- How the interactive graphics display works
- Display types: Random Scan and Raster Scan

Unit-II: Line Drawing Technique **(7 h)**

- Co-ordinate Systems
- The Simple DDA
- The Symmetrical DDA
- Bresenham's line drawing Algorithm
- Bresenham's circle drawing Algorithm

Unit-III: Two Dimensional and Three Dimensional Transformations **(8 h)**

- Transformation principles
- Concatenations
- 2D Transformations, 2D Matrix Representation
- 3D Transformations, 3D Matrix Representation
- Transformation in Viewing
- The Perspective Transformation

Unit-IV: Clipping and Windowing **(9 h)**

- Definitions: Window, View port, Clipping
- Cohen-Sutherland line clipping algorithm
- Mid-point Subdivision line clipping algorithm
- Polygon Clipping
- The Windowing Transformation
- 3-D Clipping

Unit-V: Raster Graphics and Solid Area Scan-Conversion **(7 h)**

- Introduction
- Scan Converting Line and Polygon drawing
- Coherence
- (YX) Algorithm
- Priority: Painter's Algorithm

Unit-VI: Hidden Surface Elimination **(6 h)**

- Object Space and Image Space Algorithms
- The Depth Buffer Algorithm
- Warnock's Algorithm

Reference:

- Newman W.M., Robert F. S., Principles of Interactive Computer Graphics, 2nd Edition, Tata-McGraw Hill, New Delhi
- Hughes J. F., Dam A.V., Sklar D. F., Feiner F. K., Computer Graphics Principals and Practice, 3rd edition, Addison Wesley
- Baker H., (1997), Computer Graphics C version, 2nd edition, Pearson, India

Teaching methods:

- Classroom teaching, Lecture method, ICT enabled teaching

T.Y. B.Sc. (Computer Science): Semester-V
Discipline Specific Core (DSC) Course
CS-353: Software Engineering

Total Hours: 45 **Credits: 3**

Course objectives:

- To understand software development process
- To learn various software development models
- To elaborate the concepts of designing, testing & quality about software.

Course Outcomes:

Students will be able to

- perform the E-R Diagram, DFD, Data dictionary, Decision tree about software
- design the software in learned language using the course content
- get the knowledge of types of testing & procedure to conduct testing of software in industry

Unit-I: Introduction to Software Engineering **(8 h)**

- Software and Software Engineering
- Evolution of Software
- Software Characteristics
- Software Applications
- Software Myths
- Software Process
- Software Development Life Cycle (SDLC)

Unit-II: Software Development Model **(8 h)**

- Waterfall Model
- Prototyping Model
- Incremental Development Model
- RAD model
- Spiral Model

Unit-III: Requirement Analysis and Specification **(8 h)**

- Requirements Engineering
- Fact finding Techniques
- Introduction to Types of Requirement Modeling
- Data Modeling Concepts- Data Objects, Data Attributes & Relationship.

Unit-IV: Design Engineering **(7 h)**

- Characteristics of good Software Design
- Design Concepts- Architecture, Modularity, Information Hiding
- Cohesion & Coupling
- Decision Table & Decision Tree
- Data flow Diagram

- Data Dictionary

Unit-V: Software Coding & Testing **(7 h)**

- Coding standards & Guidelines
- What is testing?
- Testing Activities
- Black box testing
- White box testing
- Introduction to Debugging Approaches – Brute force Method, Backtracking, Case Elimination Method, Programming Slicing

Unit-VI: Software Quality **(7 h)**

- What is Quality?
- Software Quality - Garvin's quality dimensions, Mc Calls quality factors, ISO 9125 quality factors
- Elements of Software Quality Assurance
- ISO 9000 & Certification

References:

- Pressman R. S., (2009), Software Engineering a Practitioners Approach, 7th edition, McGraw Hill International Edition .
- Mall R., Fundamental of Software Engineering, ISBN 4th edition, PHI Learning Private Limited.
- Aggrawal K.K., Singh Y., (2008), Software Engineering, 3rd edition, New Age International Publisher, New Delhi

Teaching methods:

- Classroom teaching, Lecture method, ICT enabled teaching

T.Y. B.Sc. (Computer Science): Semester-V
Discipline Specific Core (DSC) Course
CS-354: Data Science-I

Total Hours: 45 **Credits: 3**

Course objectives:

- To understand role of Statistics in Data Science.
- To understand role of Mathematics in Data Science.
- To impart knowledge for enabling students to develop data analytics skills and meaningful interpretation to the data sets.

Course outcome:

Students will be able to-

- understand about the collection of the data, condensation and summarization into a compact form.
- understand about the representation of data in a neat, compact and clear form.
- understand the concepts of Sample space and events, theory of Permutation and Combinations.
- understand the concept of Probability, Conditional probability of an event, Independence of events.

Unit-I: Fundamentals of Data Science **(10 h)**

- Introduction to Data Science,
- Need of Data Science,
- Big Data and Data Science',
- Data Science and machine learning,
- Data Science Life Cycle, Data Science Platform,
- Data Science Use Cases Skill Required for Data Science.

Unit-II: Mathematics for Data Science **(10 h)**

- Linear Algebra: Vectors, Matrices, etc.
- Optimization: Theory of optimization, Gradients Descent, etc.

Unit-III: Introduction to Statistics **(13 h)**

- Descriptive vs. Inferential Statistics,
- Types of data,
- Measures of central tendency and dispersion Hypothesis & inferences,
- Hypothesis Testing, Confidence Interval, Central Limit Theorem

Unit-IV: Probability and Probability Distributions **(12 h)**

- Probability Theory,
- Conditional Probability,
- Data Distribution,
- Distribution Functions: Normal Distribution, Binomial Distribution

References:

- Gupta S.C., Kapoor V. K. (2017), Fundamentals of Mathematical Statistics, S. Chand and Sons, New Delhi.
- Rohatgi V.K., (2015), An Introduction to Probability theory and Mathematical Statistics, 3rd edition John Wiley and Sons, New York.
- Rachel S.., Cathy N., (2013), Doing Data Science: Straight Talk from the Frontiline, 1st edition Schroff/O'Reilly,.
- Foster P., Tom F., (2013), Data Science for Business What You Need to Know About Data Mining and Data-Analytic Thinking, O'Reilly

Teaching methods:

- Classroom teaching, Lecture method, ICT enabled teaching

Department of Computer Science M. J. College Jalgaon

T. Y. B. Sc (Computer Science): Semester V
Discipline Specific Core (DSC) Course
CS-355: Programming in PYTHON-I

Total Hours: 45 **Credits: 3**

Course Objectives:

- To introduce Python Environment to students.
- To develop problem solving skills and their implementation through Python
- To understand and implement concepts of List, Tuple and Dictionary in Python

Course Outcome:

Student will be able to

- explain basic principles of Python programming language
- construct and apply various filters for a specific task.
- apply the best features of mathematics, engineering and natural sciences to program real life problems.

Unit-I: Introduction to Python **(5 h)**

- Getting Started: Introduction to Python- an interpreted high level language, interactive mode and script mode.
- Variables, Expressions and Statements
- Variables and Types-mutable and Immutable variable and Keywords.
- Operators and Operands in Python.
- (Arithmetic, relational and logical operators),
- Operator precedence, Expressions and Statements (Assignment statement);
- Taking input (using raw_input() and input()) and displaying output - print statement
- Comments in Python.

Unit-II: Conditional and Looping Construct **(4 h)**

- if - else statement and nested if – else
- while loop, for loop,
- use of range function in for,
- Nested loops
- break, continue, pass statement
- Use of compound expression in conditional constructs

Unit-III: Functions **(6 h)**

- Built-In Function, invoking built in functions
- Module (Importing entire module or selected objects using from statement)
- Functions from math, random, time & date module.
- Composition
- User Define Function: Defining, invoking functions, passing parameters (default parameter values, keyword arguments)
- Scope of variables, void functions and functions returning values

Unit-IV: Strings **(10 h)**

- Creating, initializing and accessing the elements;
- String operators: +, *, in, not in, range, slice [n:m]
- String built in functions & methods: len, capitalize, find, isalnum, isalpha, isdigit, lower, islower, isupper, upper, lstrip, rstrip, isspace, istitle, partition, replace, join, split, count, decode, encode, swapcase
- Strings constants defined in string module Regular Expression and Pattern Matching

Unit-V: Lists **(10 h)**

- Concept of mutable lists, creating, initializing and accessing the elements of list
- List operations (Concatenation, Repetation, Membership, list slices), List comprehensions
- List functions & methods: len, insert, append, extend, sort, remove, reverse, pop Tuples
- Immutable concept, creating, initializing and accessing the elements in a tuple;
- Tuple functions: cmp(), len(), max(), min(), tuple() Sets
- Concept of Sets, creating, initializing and accessing the elements of set
- Sets operation (Membership, union, intersection, difference, and symmetric difference Dictionaries Concept of key-value pair, creating, initializing and accessing the elements in a dictionary,
- Traversing, appending, updating and deleting elements
- Dictionary functions & Methods: cmp, len, clear (), get(), has_key(), items(), keys(), update(), values()

Unit-VI: Modules **(10 h)**

- More on Modules: Executing modules as scripts, The Module Search Path, "Compiled" Python files Standard Modules
- The dir( ) Function
- Packages Importing * From a Package, Intra-package References, Packages in Multiple Directories

Reference Books

- Lutz M.Z., (2009), Learning Python, 4th edition, O'Reilly USA
- Dawson M., (2008), Programming with python, A users Book, Cengage Learning
- Beazley D., (2010), Python Essential Reference, 4th edition, Addision Welsely, USA
- McGrath M., (2018), Python in easy steps, 2nd edition, In easy steps Limited,
- Peter C. N., Samuel A., (2005), Beginning Pythonl, Wrox Publication.
- Brown C.M.., (2018), Python, the Complete Reference, McGraw Hill Education, USA

Web References:

- https://docs.python.org

Teaching methods:

- Classroom teaching, Lecture method, ICT enabled teaching

T. Y. B. Sc (Computer Science): Semester V
Discipline Specific Core (DSC) Course
CS-356: 'R' Programming

Total Hours: 45 **Credits: 3**

Course objectives:

- To study R software for statistical analysis
- To study Basics of R Programming.
- To introduce R graphics

Course outcomes:

Students will be able to --

- understand how to download and install R software.
- know various R packages with their utility.
- understand data structures in R.
- use R software for statistical computations.
- use R software for exploratory data analysis.

Unit-I: Introduction to R **(5 h)**

- Downloading and installation of R.
- Features of R, Meaning of package, base package,
- Installation and loading of a package, to delete a package, getting help.
- Meaning of workspace, saving a workspace, loading a workspace, deleting a workspace.

Unit-II: Basics of R **(8 h)**

- Data Types in R: numeric, character, logical.
- Variables in R: Assignment, Finding, Deleting variable
- Operators in R: Arithmetic, Relational, Logical, Assignment, Miscellaneous Operators
- Conditional and looping statements in R.
- Functions in R: Definition, Built in function, User Defined Function, Calling function (Without parameter, with parameter, default parameter)
- String functions

Unit-III: Working with data objects and using functions **(12 h)**

- Data objects, Types of data object: vector, list, matrix, array, factor, and data frame
- Functions for working with objects: mode( ), length( ), cbind( ), rbind( ), names( ), ls( ), rm( )
- Vectors: creating a vector, modifying a vector, deleting a vector.
- Working with vectors: the functions to be discussed- c( ), rep( ), rev( ), sort( ), diff( ), max( ), min( ), colon operator(:), seq( ), scan( ), cut( ), cat( ), table( ), which( ), unique( )
- Mathematical functions: abs( ), sqrt( ), log( ), log10( ), exp( ), sin( ), cos( ), tan( ), atan( ),
- round( ).
- Meaning of data frame, creation of data frame, modifying a data frame, deleting data frame, extracting elements from a data frame

- Functions to be discussed: subset( ), transform( ), attach( ), detach( ), with( ), data.entry( ),
- edit( ), is.data.frame( ), as.data.frame( ).

Unit-IV: Graphics **(10 h)**

- Low level and high level functions.
- Functions to be discussed: plot( ), lines( ), points( ), smooth.spline( ), curve( ),barplot( ), pie( ), hist( ), mtext( ), legend( ).

Unit-V: Statistical applications **(10 h)**

- Diagrams and Graphs: Bar Chart (Subdivided, multiple), Pie diagram, Boxplot, Histogram, Line Graph, Scatterplots.
- Measures of Central Tendency: Mean, Mode, Median,
- Measures of Dispersion: Range, M. D. about Mean, Mode, Median, S.D., Variance and C.V.

References:

- Purohit S.G., Gore S.D., Deshmukh S.R. (2008). Statistics Using R. 2nd edition, Narosa Pub.
- Dalgaard P.,. (2002), Statistics and computing: Introductory Statistics with R. 2nd edition, Springer
- Maindonald, J., Braum, J. (2007), Data Analysis and Graphics Using R: An example-based Approach , 3rd edition, Cambridge University Press, England
- Hey-Jahans, C. (2012). An R Companion to Linear Statistical Models. 1st edition, CRC Press.
- Gardener, M. (2012). Beginning R: The Statistical Programming Language. Wiley & Sons.
- Acharya, S. (2018). Data Analytics using R. McGraw Hill Education.
- Wickham, H. and Grolemund, G. (2017). R for Data Science. O'Reilly Media.
- Lander, J.P. (2017). R for Everyone: Advanced Analytics and Graphics. Addison-Wesley

Web Reference:

- www.tutorialspoint.com/r

Teaching methods:

- Classroom teaching, Lecture method, ICT enabled teaching

T. Y. B. Sc (Computer Science): Semester V
Skill Enhancement Course (SEC)
CS-350: Cloud Computing-I

Total Hours: 30 **Credits: 2**

Course Objectives:

- To provide learners with the comprehensive and in-depth knowledge of Cloud Computing concepts
- To learn how to use Cloud Services.
- To implement Virtualization
- To understand security issues in cloud computing

Course Outcome:

Students will be able to-

- learn cloud computing types and models
- get knowledge of cloud os and architectures.
- understand virtualization and security concept

Unit-I: Introduction to Cloud Computing. **(8 h)**

- Cloud Computing definition,
- Types of cloud,
- Benefits and challenges of cloud computing,
- Cloud Characteristics,
- Evolution of Cloud Computing

Unit-II: Principles of Parallel and Distributed Computing **(8 h)**

- Eras of Computing,
- Parallel v/s distributed computing,
- Elements of Parallel Computing,
- Elements of distributed computing,
- Technologies for distributed computing

Unit-III: Virtualization **(8 h)**

- Introduction,
- Characteristics of virtualized environments,
- Taxonomy of virtualization techniques,
- Virtualization and cloud computing,
- Pros and cons of virtualization,
- Technology examples.

Unit-IV: Cloud Security **(6 h)**

- Cloud Security Fundamental,
- Privacy and Security in Cloud,
- Cloud Security Architecture,
- Identity Management and Access control,

- Cloud Computing security challenges

References:

- Saurabh K.,(2012), Cloud Computing, 2nd edition, Wiley Publication.
- Gautam S., (2010), Enterprise Cloud Computing, 1st edition, Cambridge publication.
- Buyya R., Christian V., Selvi S.T., (2013), Mastering Cloud Computing Foundations and Applications Programming, Morgan Kaufmann
- Ronald K., Russell D. V.,(2014), Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Wiley-India.
- Judith H., Bloor R., Kanfman R., Halper F., (2010), Cloud Computing for Dummies, Wiley, India

Teaching methods:

- Classroom teaching, Lecture method, ICT enabled teaching

Department of Computer Science M. J. College Jalgaon

T. Y. B. Sc (Computer Science): Semester V
Discipline Specific Core (DSC) Course
CS-357: Practical Course based on CS-351 and CS-352

Total Hours: 60 **Credits: 2**

Course Objectives:

- To understand how development of software tools is done
- To understand the design and execution of small computer
- To understand the design and implementation of Interrupt Handler
- To understand the basic concepts of Computer Graphics
- To understand the different types of geometric transformation.

Course Outcomes:

Students will be able to:

- design and code the different system software like line editor, small computer simulators and many more.
- understand Graphics Concept Practically
- get hands on using standard graphics library
- implement DDA, Bresenham's Line,Circle Drawing Algorithm
- implement 2D Transformation: Translation, Scaling and Rotation

Lab on Systems Programming

Sr. No.	Topic Particular	Hours
1	To create line editor with features like create a new file, open existing file, Append in the file, Save and print file as well as to insert, delete, copy & move Lines in the file.	08
2	Simulate CPU for SMAC0 (Small Computer) SMAC0 Programming:- 1. Addition of two numbers 2. Subtraction of two numbers, 3. Multiplication of two numbers 4. Division of two numbers 5. Find MOD 6. GCD of two numbers 7. LCM of two numbers 8. Factorial of given number 9. Square & Cube of given number. 10. Fibonacci series (Do not use op-codes for MULT, MOD and DIV operation)	08
3	Interrupt handler in C (Keyboard interrupt should be disabled and alt-C should be used to toggle CAPS Lock and alt-N should be used to toggle NUM lock)	04

Lab on Computer Graphics

Sr. No	Topic Particular	Hours
1	Implement Bresenham's Line Drawing Algorithm	04
2	Implement Bresenham's Circle Drawing Algorithm	04
3	Implement DDA line Drawing Algorithm	04
4	Implementing Translation transformation on polygon	04
5	Implementing Scaling transformation on polygons	04
6	Implementing Rotation transformation on polygons	04

References

- Dhamdhere D. M., (2011), Systems Programming, 1st edition, Tata McGraw-Hill Education, New Delhi
- Baker H., (1997), Computer Graphics C version, 2nd edition, Pearson, India

Teaching methods:

- Laboratory method, Lecture cum demonstration methods

Department of Computer Science M.J. College Jalgaon

T. Y. B. Sc (Computer Science): Semester V
Discipline Specific Core (DSC) Course
CS-358: Practical Course based on CS-355

Total Hours: 60 **Credits: 2**

Course Objectives:

- To provide students hands on practice on Python programs
- To apply List, Tuple and Dictionary for problem solving in Python

Course Outcome:

Student will be able to

- develop the programs using Python.
- develop their own module according to requirement and reuse it wherever necessary.
- solve real life problems using Python.

Instruction:

At the time of Practical you can use any Python IDEs and Code Editors (PyCharm, Spyder, Thonny, etc.).

Sr. No	Topic Particular	Hours
1	Write a program to check whether the number is even or odd, print out an appropriate message to the user.	04
2	Write a program which will find all such numbers which are divisible by 7.	04
3	Write a program that prints out all the elements of the list that are less than 10.	04
4	Write a program to determine whether the number is prime or not.	04
5	Write a program to check whether a number is palindrome or not.	04
6	Write a program which can compute the factorial of a given numbers. (using recursion and without recursion).	04
7	Write a program that asks the user how many Fibonnaci numbers to generate and then generates them.	04
8	Write a program to demonstrate all string functions in Python.	04
9	Write a program that returns a list that contains only the elements that are common between the lists (without duplicates). Make sure your program works on two lists of different sizes.	04
10	Write a program which accepts a sequence of comma-separated numbers from console and generate a list and a tuple which contains every number.Given the input: 34,67,55,33,12,98 Then, the output should be: ['34', '67', '55', '33', '12', '98'] ('34', '67', '55', '33', '12', '98')	04

References:

- Lutz M.z., (2009), Learning Python By Mark Lutz, 4th edition, O'Reilly USA
- Dawson M., (2008), Programming with python, A users Book, Cengage Learning
- Beazley D., (2010), Python Essential Reference, 4th edition, Addision Welsely, USA
- McGrath M., (2018), Python in easy steps, 2nd edition, In easy steps Limited,
- www.w3school.org

Teaching methods:

- Laboratory method, Lecture cum demonstration methods

T. Y. B. Sc (Computer Science): Semester V
Discipline Specific Core (DSC) Course
CS-359: Practical Course based on CS-356

Total Hours: 60

Credits: 2

Course objectives:

- To study R software for statistical analysis
- To study Basics of R Programming.
- To introduce R graphics

Course outcomes:

Students will be able to --

- understand how to download and install R software.
- know practically R packages with their utility.
- use R software for statistical computations.
- use R software for exploratory data analysis

Sr. No	Topic Particular	Hours
1	Demonstrate the basics operations in R.	04
2	Demonstrate the use of various data types and use of class ().	04
3	Demonstrate the use of various string functions in R.	04
4	Demonstrate the use of mathematical functions in R.	04
5	Demonstrate various data objects in R.	04
6	Write a R script to print the Fibonacci Series.	04
7	Write a R script to print Factorial of a given number.	04
8	Demonstrate the use of vector and perform various operations of it.	04
9	Demonstrate creation of data frame, modifying a data frame, deleting data frame, extracting elements from a data frame	04
10	Demonstrate all the graphics functions.	04
11	Considering a dataset plot various graphs of it.	04
12	Considering a dataset find various Measures of Central Tendency.	04

Mandatory to perform any 8 practical from above.

References

- Purohit S.G., Gore S.D., Deshmukh S.R. (2008). Statistics Using R. 2nd edition, Narosa Pub.
- Dalgaard P.,. (2002), Statistics and computing: Introductory Statistics with R. 2nd edition, Springer
- Maindonald, J., Braum, J. (2007), Data Analysis and Graphics Using R: An example-based Approach , 3rd edition, Cambridge University Press, England
- www.tutorialpoint.com/r

Teaching methods:

- Laboratory method, Lecture cum demonstration methods

Department of Computer Science M. J. College Jalgaon

T.Y. B.Sc. (Computer Science): Semester-VI
Discipline Specific Core (DSC) Course
CS-361: Operating System

Total Hours: 45 **Credits: 3**

Course objectives:

- To understand basic concepts and functions of modern operating systems.
- To understand the concept of process, and thread with scheduling, process synchronization and deadlock.
- To know the concept of I/O and File management, various Memory management techniques and related algorithms. Title and Contents No. of Lectures

Course Outcomes:

Students will be able to-

- familiar with Operating System Services.
- understand CPU scheduling algorithms, memory Management Techniques, Disk Drum Scheduling algorithms, Deadlock preventions and avoidance.
- aware about android operating systems – its architecture, applications and uses.

Unit-I: Introduction **(4 h)**

- What is an operating system?
- Types of Operating System
- Services of Operating System
- Functions of operating system.

Unit-II: CPU scheduling **(10 h)**

- Multiprogramming Concepts
- **Basic Concept of CPU scheduling:** CPU-I/O burst cycle, CPU scheduler,
- Preemptive scheduling, Dispatcher , Performance criteria's
- **Scheduling Algorithms**: FCFS, SJF, Priority scheduling, Round-robin scheduling
- Multilevel queues, multilevel feedback queue

Unit-III: Memory Management **(10 h)**

- Logical versus Physical Address space, Swapping
- Multiple partition allocation MFT, MVT
- Paging, Segmentation
- Virtual Memory Management – Background, Demand paging

Unit-IV: Disk and Drum Scheduling **(6 h)**

- First Come first serve scheduling,
- Shortest Seek Time First Scheduling,
- SCAN Scheduling,
- C-SCAN Scheduling

Unit-V: Deadlocks **(10 h)**

- Concept of Deadlock, Deadlock Characterization, Deadlock Prevention,
- Deadlock, Avoidance, Deadlock Detection, Recovery from Deadlock

Unit-VI: Overview of Android Operating system **(5 h)**

- What is android operating system? Android Architecture, Features of Android operating system, Applications of android operating system, What is Google play store?

References:

- Silberschatz P., Galvin P. B., Gangne G., (2012), Operating system concepts, 9th edition, Addison Wesley, USA
- Tanenbaum A. S., (2007), Modem Operating Systems, 3rd edition, P .H.I. New Delhi.
- Godbole A.S., Kahate A., (2010), Operating Systems, 3rd edition, McGraw Hill Education.
- Garaenta M., (2014), Learning Android, 2nd edition, O Reilly
- Wolfson M., (2013), Android developers tools Essential, O Reilly

Teaching methods:

- Classroom teaching, Lecture method, ICT enabled teaching

T.Y.B.Sc. (Computer Science): Semester-VI
Discipline Specific Core (DSC) Course
CS-362: Computer Network

Total Hours: 45 **Credits: 3**

Course Objectives:

- To understand different types of networks, various topologies and application of networks.
- To understand the various Internetworking models.
- To understand various working of each layer in above models.
- To aware about Computer security using crypotography.

Course Outcomes:

Students will be able to-

- understand the information exchange done across the network with the help of OSI & TCP/IP models.
- learn how errors are captured & handled in network.
- familiar with logical addressing and routing algorithms.

Unit-I: Introduction to Computer Networks and Network Model **(8 h)**

- **Computer Networks**: Goals and applications – Business Application , Home Application, Mobile User, Social Issues
- **Network Types**: LAN, MAN, WAN, Wireless Networks, Home Networks, Internet
- Work, **Topologies**: star, bus, mesh, ring etc.
  OSI Reference Model: Functionality of each layer, TCP/IP Reference Model

Unit-II: The Physical Layer **(8 h)**

- Transmission Media
- Guided Media: Fiber Optics, Satellite Communication, Microwave Communication, Submarine Cables.
- Unguided Media: Radio Transmission, Microwave Transmission, Infrared &
- Millimeter Waves, Light wave Transmission
- Switching - Circuit Switching, Message Switching and Packet Switching, comparison of circuit & packet switching

Unit-III: The Data link Layer **(8 h)**

- Services Provided to Network Layer
- Framing, Error Control , Flow Control
- Error Detection – Redundancy, Parity Check, Checksum & CRC
- Error Correction – Hamming Code.

Unit-IV: The Network Layer **(7 h)**

- Logical Addressing IPV4 Addresses:
  - Address Space
  - Classfull Addressing.

- Classless Addressing, Routing Algorithm:
    - Shortest Path,
    - Multicast Routing,
    - Congestion Control,
    - Introduction to Congestion Control,
    - Deadlocks

Unit-V: Transport Layer **(7 h)**

- Process to Process Delivery, Client-Server Paradigm, Multiplexing and
- Demultiplexing Connectionless v/s Connection Oriented Services, Reliable v/s
- Unreliable Transmission, UDP and TCP

Unit-VI: Cryptography and Public key Infrastructure **(7 h)**

- Introduction:
    - Cryptography,
    - Cryptanalysis,
    - Cryptology,
    - Substitution
- Techniques:
    - Caesar's cipher,
    - Monoalphabetic and Polyalphabetic
- Transposition techniques:
    - Rail fence technique,
    - Simple Columnar
- Public key infrastructures:
    - basics, digital certificates,
    - certificate authorities,
- registration authorities, Digital Signature

References:

- Tanenbaum A. S., (2003), Computer Networks, 4th edition, Prentice Hall, Netharland
- Forouzan B. A., (2013), Data Communication & Networking, 5th edition, McGraw Hill Higher Education Delhi
- Black U.D.,(1987), Data Communication & Distributed Networks, 2nd edition Prentice-Hall, Englewood Cliffs
- Kahate A., (2017), Cryptography and Network Security, 3rd edition, McGraw Hill.

Teaching methods:

- Classroom teaching, Lecture method, ICT enabled teaching

T.Y.B.Sc. (Computer Science): Semester-VI
Discipline Specific Core (DSC) Course
CS-363: Theoretical Computer Science

Total Hours: 45 **Credits: 3**

Course Objectives:

- To understand finite state and pushdown automata.
- To have a knowledge of regular languages and context free languages.
- To know the relation between regular language, context free language and corresponding recognizers.
- To study the Turing machine and classes of problems.

Course Outcomes:

Students will be able to-

- understand the use of Sets, Relations and Graphs.
- understand Various Languages in TCS.
- learn Regular Languages and Expressions.
- familiar with Pumping Lemma and its applications.
- explore the knowledge of Pushdown Automata.
- Solve examples based on Normal forms used in TCS.
- Realize Turing Machine.

Unit-I: Mathematical Preliminaries **(4 h)**

- Symbol, Alphabet, String, Formal Language, Operation on languages
- Sets, Relations: Sets and Subsets, Relations, Closure of Relations
- Graphs & Trees: Graphs, Trees
- Principal of Induction: Method of Proof by Induction

Unit-II: Finite Automata **(14 h)**

- Definition of Automata, Why study Automata Theory?
- Introduction to finite Automata, Structural representations, Automata and Complexity
- Descriptions of Finite Automata, Transition Systems, Transition Functions,
- Deterministic Finite Automata (DFA), Nondeterministic Finite Automata (NFA) ,
- The Equivalence of DFA and NFA, Minimization of DFA, Finite Automata with ε-
- Moves, Melay and Moore Machines: Definition and Examples, Applications of Finite
- Automata

Unit-III: Regular Expressions & Regular Sets **(8 h)**

- Regular Expressions, FA & Regular Expressions,
- Convert Regular Expression to FA
- Construct FA from Regular Expression
- Pumping Lemma for Regular Sets and applications.

Unit-IV: Context Free Grammars **(10 h)**

- Introduction to Context Free Grammars

- Derivation Trees, Ambiguity in CFG, Simplification of Context Free Grammars
- Useless Symbols, Null Production, Unit Production, Normal forms for CFG
- Chomsky Normal Form (CNF), Greibach Normal Form (GNF)

Unit-V: Pushdown Automata **(4 h)**

- Basic Definitions, Types of PDA, Acceptance by Pushdown Automata,
- PDA and Context Free Language

Unit-VI: Turing Machine **(5 h)**

- Introduction, Turing Machine Model, Representation of Turing Machine
- Design of Turing Machine

References:

- Hopcraft J. E., Motwani R., Ullman J. D., (2008), Introduction to Automata Theory, Languages & Computations, 3rd edition, Pearson publication
- Mishra K. L.P., Chandrasekaran N., (2006), Theory of Computer Science, 3rd edition, Prentice-Hall of India Pvt. Ltd.
- Cohen D.A., (1996), Introduction to Computer Theory, 2nd Revised edition John Wiley & Sons.
- Rajpal S., (2004), Theory of Automata and Formal Languages, 1st edition, Galgotia Publications, New Delhi.
- http://nptel.ac.in/

Teaching methods:

- Classroom teaching, Lecture method, ICT enabled teaching

T.Y. B.Sc. (Computer Science): Semester-VI
Discipline Specific Core (DSC) Course
CS-364: Data Science-II

Total Hours: 45 **Credits: 3**

Course objectives:

- To understand basic data science concepts.
- To Learn to detect and diagnose common data issues, such as missing values, special values, outliers, inconsistencies, and localization.
- To address advanced statistical situations, Modeling and Machine Learning.

Course outcome:

Students will be able to-

- understand & comprehend the problem;
- define suitable statistical method to be adopted

Unit-I: Data Scientist's Tool Box **(10 h)**

- Turning data into actionable knowledge
- Introduction to the tools that will be used in building data analysis software:version control, markdown, git, GitHub, R, and RStudio.

Unit-II: Exploratory Data Analysis **(12 h)**

- Essential exploratory techniques for summarizing data applied before formal modeling commences,
- eliminating or sharpening potential hypotheses about the world that can be addressed by the data,
- common multivariate statistical techniques used to visualize high-dimensional data.

Unit-III: Data Pre-processing **(10 h)**

- Need for pre-processing of the data,
- Descriptive data summarization, Data cleaning,
- Data Integration and transformation, Data reduction,
- Data discretization and concept hierarchy generation.
- **Data transformations:**
    - Dimension reduction, Feature extraction,
    - Smoothing and aggregating

Unit-IV: Machine Learning **(13 h)**

- What is machine learning and why should you care about it?
- Applications for machine learning in data science
- Where machine learning is Used in the data science process,
- Python tools used in machine learning
- The modeling process:
    - Engineering features and selecting a model,
    - Training your model, Validating a model,

- Predicting new observations
- Types of machine learning:
  - Supervised learning,
  - Unsupervised learning

References:

- Gupta S.C., Kapoor V. K. (2017), Fundamentals of Mathematical Statistics, S. Chand and Sons, New Delhi.
- Rohatgi V.K., (2015), An Introduction to Probability theory and Mathematical Statistics, 3rd edition John Wiley and Sons, New York.
- Rachel S., Cathy N., (2013), Doing Data Science: Straight Talk from the Frontiline, 1st edition, Schroff/O'Reilly.
- Foster P., Tom F., (2013), Data Science for Business What You Need to Know About Data Mining and Data-Analytic Thinking, O'Reilly

Teaching methods:

- Classroom teaching, Lecture method, ICT enabled teaching

T.Y.B.Sc (Computer Science): Semester VI
Discipline Specific Core (DSC) Course
CS-365: Programming in PYTHON-II

Total Hours: 45 **Credits: 3**

Course Objectives:

- To learn how to design and code Python applications.
- To develop problem solving skills and their implementation through Python.
- To develop the ability to write database applications in Python

Course Outcomes:

Student will be able to

- Implement object oriented concepts, database applications.
- Implement GUI applications using Python.
- Apply the best features of mathematics, engineering and natural sciences to program real life problems.

Unit-I: Errors and Exceptions **(5 h)**

- Syntax Errors, Exceptions, Handling Exceptions, Raising Exceptions, User-defined Exceptions, Defining Clean-up Actions(try - finally), Predefined Clean-up Actions

Unit-II: I/O and File Handling **(10 h)**

- Output Formatting, Reading and Writing Files, File Exceptions, Paths and
- Directories, Directory Contents, Obtaining Information about Files, Renaming,
- Moving, Copying, and Removing Files, Creating and Removing Directories, Globbing

Unit-III: Introduction to Object Oriented concepts in Python **(5 h)**

- Object Oriented concepts, Objects
- Python Scopes and Namespaces, Classes, Class Objects,
- Instance Objects, Method Objects,
- Class and Instance Variables, Inheritance

Unit-IV: GUI with Python **(15 h)**

- GUI Programming Toolkits for Python,
    - Tkinter Introduction,
    - Creating GUI Widgets with Tkinter,
    - Resizing the Widget, Configuring Widget Options,
- Putting the Widgets to Work,
- Creating Layouts, Packing Order,
- Controlling Widget Appearances,
- Radio Buttons and Checkboxes, Dialog Boxes, Other Widget Types

Unit-V: Python with MySQL **(10 h)**

- Introduction to MySQL,
    - Installing MySQL Driver - MySQL Connector or MySQL

- Db MySQL Database connection with Python,
    - Creating Database in MySQL using Python
    - Create a Table in MySQL with Python,
    - Insert, Select, Update and Delete
- Operation in MySQL with Python, COMMIT and ROLLBACK Operations

References:

- Lutz M.Z., (2009), Learning Python, 4th edition, O'Reilly USA
- Dawson M., (2008), Programming with python, A users Book, Cengage Learning
- Beazley D., (2010), Python Essential Reference, 4th edition, Addision Welsely, USA
- McGrath M., (2018), Python in easy steps, 2nd edition, In easy steps Limited,
- Peter C. N., Samuel A., (2005), Beginning Pythonl, Wrox Publication.
- Brown C.M.., (2018), Python, the Complete Reference, McGraw Hill Education, USA

Teaching methods:

- Classroom teaching, Lecture method, ICT enabled teaching

Department of Computer Science M.J. College Jalgaon

T.Y.B.Sc (Computer Science): Semester VI
Discipline Specific Core (DSC) Course
CS-366: PHP Programming

Total Hours: 45 **Credits: 3**

Course Objectives:

- To understand Core-PHP concepts, Server Side Scripting Language
- To acquaint knowledge of Object oriented in PHP.

Course Outcomes:

Student will be able to

- To Design dynamic and interactive Web pages.
- To use PHP framework for effective design of web applications.

Unit-I: Introduction to PHP **(5 h)**

- Web architecture, Web Server (xamp Server, apache server), Web Browser,
- Introduction to Web Development, Introduction to PHP, Features & Drawbacks of PHP, How PHP Works? Version of PHP.

Unit-II: PHP Language Basics **(10 h)**

- Lexical Structure of PHP, Structure & Syntax of PHP, PHP with HTML, Comments,
- Data Types, Variables, Operator, Flow Control Statements, Conditional Statements,
- Looping Statements, Exit, Return, Die, Include and Require Statements

Unit-III: PHP Array **(5 h)**

- Indexed Vs Associative arrays, Multidimensional arrays, Storing data in arrays,
- Extracting multiple values, Converting between arrays and variables, Traversing
- arrays, Sorting arrays, Different array function in PHP

Unit-IV: PHP Function and String **(8 h)**

- Introduction to Function: Defining and Calling a function
- Scope of variables in function, Function Parameters, Returning Values from a function,
- Recursive Functions, Introduction to String: Types of strings in PHP, Printing functions
- Comparing strings, Manipulating and Searching strings, Regular Expressions

Unit-V: Web Techniques **(7 h)**

- Introduction, HTTP Basics, Processing Forms • Methods (Get and Post Method),
- Parameters ($_GET and $_POST), Self-Processing Pages, File Uploads, Form Validation, Emailing in PHP, Sending Free SMS to Mobile, Loading PHP application on web server
- By FTP.

Unit-VI: Object-Oriented PHP **(10 h)**

- Introduction and Benefits of OOPs, Creating a Class, Creating an Object, Adding a Method

- Adding a Properties, Visibility (Public, Private and Protected) ,Constructor and Destructors
- Inheritance (Extending a class), Abstract classes, Final classes, Interfaces, Exception handling.

References:

- Gilmore W. J., (2010), Beginning PHP and MySQL, 3rd edition., Apress Publication.
- Bayross I., Shah. S., (2010), PHP 5.1 for Beginners, ShroffX-team
- Mercer D., Kent A., Steven N., Davin M., (2004), Beginning PHP 5, Wiley India Pvt Ltd
- Holzner S., (2000), HTML Black Book, Coriolis Group, USA.
- Kogent Learning Solutions Inc, (2009), Web Technologies, Black Book, Dreamtech Press

Websites:

- http://www.php.net.in
- http://www.w3schools.com
- http://www.tutorialpoints.com

Teaching methods:

- Classroom teaching, Lecture method, ICT enabled teaching

T. Y. B. Sc (Computer Science): Semester VI
Skill Enhancement Course (SEC)
CS-360: Cloud Computing–II

Total Hours: 30 **Credits: 2**

Course objectives:

- To give technical overview of Cloud Programming and Services.
- To understand different Cloud services

Course outcome:

Students will be able to-

- articulate the main concepts, key technologies, strengths, and limitations of cloud computing.
- identify the architecture and infrastructure of cloud computing, including SaaS, PaaS, IaaS.
- explain the core issues of cloud computing.
- understand Security Mechanisms and issues in various Cloud Applications

Unit-I: Software as a Service (SaaS) **(4 h)**

- Introduction to SaaS, Web services ,Case Study on SaaS

Unit-II: Infrastructure as a Service (IaaS) **(8 h)**

- Introduction to IaaS
- Introduction to virtualization
    - Different approaches to virtualization,
- Data storage in cloud computing, Examples of clouds

Unit-III: Platform as a Service (PaaS) **(6 h)**

- Evolution of computing paradigms and related components,
- Introduction to PaaS-
    - What is PaaS, Service Oriented Architecture (SOA)

Unit-IV: Industrial Platforms and New Developments **(8 h)**

- Amazon Web Services
- Google App Engine
- Microsoft Azure
- Cloud Applications

Unit-V: Issues in Cloud Computing **(4 h)**

- Issues in Inter cloud computing , Quality of services in cloud Computing , Data
- Migration in Cloud , Streaming in Cloud

References:

- Saurabh K.,(2012), Cloud Computing, 2nd edition, Wiley Publication.
- Gautam S., (2010), Enterprise Cloud Computing, 1st edition, Cambridge publication.
- Buyya R., Christian V., Selvi S.T., (2013), Mastering Cloud Computing Foundations and Applications Programming, Morgan Kaufmann
- Ronald K., Russell D. V.,(2014), Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Wiley-India.

Teaching methods:

- Classroom teaching, Lecture method, ICT enabled teaching

Department of Computer Science
M. J. College
Jalgaon

T.Y. B.Sc. (Computer Science): Semester-VI
Discipline Specific Core (DSC) Course
CS-367: Practical Course Based on CS-361 and CS-366

Total Hours: 60 **Credits: 2**

Course objectives:

- To familiarize students with the architecture of OS.
- To provide necessary skills for developing and debugging programs.
- To study the process management and scheduling.
- To Acquire knowledge and Skills for creation of Web Site considering both client- and server-side Programming.
- To create Web application using tools and techniques used in industry.

Course outcome:

Students will be able to-

- describe the important computer system resources and the role of operating system in their management policies and algorithms.
- aware concepts of multiprogramming, multithreading and multitasking.
- demonstrate memory management algorithms
- illustrate file-handling concepts by implementing suitable algorithms
- develop simple web application using server side PHP programing and Database Connectivity using MySQL.

Part I: Lab on Operating System

Sr.No	Topic Particular	Hours
1	CPU Scheduling – FCFS	04
2	CPU Scheduling - SJF Preemptive	04
3	CPU Scheduling - SJF Non-Preemptive	04
4	Demand Paging – FIFO	04
5	Demand Paging – LRU	04
6	Demand Paging –Optimal Page Replacement	04

Part II: Lab on PHP Programming

Sr.No	Topic Particular	Hours
1	Design web pages using HTML that will contain online admission forms.	04
2	Write PHP scripts that demonstrate fundamentals PHP.	04
3	Write PHP script that will display grade based on criteria given below using the marks obtained in T.Y.Bsc. Examination. a. Distinction (70 and above) b. First Class (60 - 69) c. Pass (40 - 59) d. Fail (below 40)	04
4	Write a PHP script to demonstrate different String functions.	04
5	Write a PHP script to use Functions (Call by Value, Call by reference).	04

6	Write a PHP script to Demonstrate OOPS Concept in PHP.	04
7	Write a PHP script to demonstrate Exception Handling.	04
8	Write a PHP script to demonstrate Form Data Handling using Get and Post methods.	04
9	Write a PHP Script to Sending Email to your friend.	04
10	Write a PHP Script to Sending SMS to your friend	04

Mandatory to perform any 08 practical from above.

References:

- Godbole A.S., Kahate A., (2010), Operating Systems, 3rd edition, McGraw Hill Education.
- Garaenta M., (2014), Learning Android, 2nd edition, O Reilly
- Gilmore W. J., (2010), Beginning PHP and MySQL, 3rd edition., Apress Publication.
- Bayross I., Shah. S., (2010), PHP 5.1 for Beginners, Shroff/X-team
- Mercer D., Kent A., Steven N., Davin M., (2004), Beginning PHP 5, Wiley India Pvt Ltd

Teaching methods:

- Laboratory method, Lecture cum demonstration methods

T. Y. B. Sc (Computer Science): Semester VI
Discipline Specific Core (DSC) Course
CS-368: Practical Course based on CS-365

Total Hours: 60 **Credits: 2**

Course Objectives

- To learn how to design and code Python applications.
- To develop different application using Python.
- To develop the ability to write database applications in Python

Course Outcome:

Student will be able to

- develop Python Programs using object oriented concepts.
- connect Database with Python applications.
- implement GUI applications using Python.

Instruction:

At the time of Practical you can used any Python IDEs and Code Editors (PyCharm, Spyder, Thonny, etc.).

Sr. No	Topic Particular	Hours
1	Write a program to demonstrate Exception Handling in Python	04
2	Write a program to demonstrate the working of classes and objects.	04
3	Write a program to demonstrate Inheritance in Python	04
4	Write a program to demonstrate Overloading Methods and Operator in Python.	04
5	Write a program to demonstrate read & write file, renaming, Moving, Copying, and Removing Files.	04
6	Write a Python GUI program to create text-box to accept a value from the user using tkinter module	04
7	Write a Python GUI program to add a button in your application using tkinter module.	04
8	Write a Python GUI program to create a Combobox and Checkbutton widget using tkinter module	04
9	Write a program to create a database application for insert, update and delete in a table using MySQL.	04

Mandatory to perform any 08 practical from above.

References:

- Lutz M.z., (2009), Learning Python By Mark Lutz, 4th edition, O'Reilly USA
- Dawson M., (2008), Programming with python, A users Book, Cengage Learning
- Beazley D., (2010), Python Essential Reference, 4th edition, Addision Welsely, USA
- McGrath M., (2018), Python in easy steps, 2nd edition, In easy steps Limited,
- www.w3school.org

Teaching methods:

- Laboratory method, Lecture cum demonstration methods

Department of Computer Science M. J. College Jalgaon

T. Y. B. Sc (Computer Science): Semester VI
Discipline Specific Core (DSC) Course
CS-369: Project Work

Total Hours: 60 **Credits: 2**

Course objectives:

- To provide comprehensive learning platform to students where they can enhance their employability skills.
- To enhance students' knowledge in one technology.
- To increase self-confidence of students and helps in finding their own proficiency.
- To provide learners hands on practice within a real job situation.

Course outcome:

Students will be able to-

- improve communication skills.
- update the latest changes in technological world.
- develop multi-skilled Computer Science professional with good technical knowledge, management, leadership and entrepreneurship skills.
- identify, formulate and model problems .
- aware about handling real time problems and finding their solution.

Project Guidelines:

1. Any open problem statement can be taken for implementation.
2. Each student shall have to carry out the project work based on System Development which may include Application Program, Database Management System, Web Based Application, Smart phone Application, System Tools, Network System Application, etc.
3. The project work should be carried out individually. No group work is allowed in the Project work. The project title should not be repeated.
4. The topic of the project should be decided with the consultation & guidance of an internal guide-teacher of the institute/college. The project should be necessarily innovative and problem solving.
5. The student should clearly mention the need of project , database(s), files required for the project, DFD , Normalization, ERD, software used for the project, reasons for selection of that software, inputs required, outputs produced etc.
6. Duration of project completion will be full semester.
7. Student should fill the status of the project work on the progress report and get the Signature of project guide regularly.
8. Student needs to spend minimum 90 hours for the project implementation.
9. No student will be permitted to appear for Viva-Voce examinations, unless and until the project report is submitted within the stipulated time.
10. Project report should be submitted with two hard copies.
11. Student will have to submit the spiral bound project report in college prescribed format at the end of the semester. For project report the specifications are – Font size 12, Name – Times New Roman, Spacing 1.5 with header and footer.

Teaching methods:

- Laboratory method, Guidance about project development

Skills acquired and Job prospects for the Computer Science students

After successful completion of three years degree course in Computer Science, student will be acquire abilities such as:

- Design, implement, and evaluate a computational system to meet desired needs within realistic constraints.
- Identify, formulate, and develop solutions to computational challenges.
- To analyze impacts of computing on individuals, organizations, and society
- Able to model and design computational systems
- Able to Administer the systems.

Laboratory Skills:

- Hands on programming languages like C, C++, Java, etc.
- Hands on Scripting languages like Html-5, PHP, etc
- Skills related to handling data analysis tools such as Python, R programming etc.
- In-depth knowledge about core computing concepts in computer science
- Design and implementation of a computational problem.

Transferable Skills:

During the course student will develop skills other than laboratory skills that are transferable across the number of career areas which include;

- Planning and Time management
- Creative thinking, Problem solving
- Report writing skill, Presentation skill
- Project Development
- Team Management
- Analytical skill

Job Opportunities:

After successful completion of B.Sc. in Computer Science, student may continue further studies like M.Sc. in Computer Science, Master of Computer Application (MCA), M. Sc. in Information Technology and then Ph.D. in Computer Science and make career in research field. Students have opportunities in private as well as public (Government) sectors with different roles such as:

- Software Developer
- Application Analyst
- Database Administrator
- Games Developer
- Information Systems Manager
- IT Consultant
- Systems Analyst
- Web Designer
- Data Scientist
- Programmer

Opportunities in higher studies

After successful completion of B.Sc. in Computer Science, student may continue further studies like M.Sc. in Computer Science / Information Technology, Master of Computer Application (MCA) and pursue higher studies. Even students can pursue other courses where graduation is essential.

--x--