



UNIVERSITY OF CALICUT

Abstract

General & Academic IV - Faculty of Science - Scheme and Syllabus of Allied core in Computer Science for Integrated MSc Statistics programme - For 2020 & 2021 admission only - Implemented - Subject to ratification by the Academic Council -Orders Issued.

G & A - IV - J

U.O.No. 11803/2022/Admn

Dated, Calicut University.P.O, 15.06.2022

*Read:-*1. U.O. No. 4852/2021/Admn dated 26.04.2021

2. Letter No. 83392/GA - IV - J3/2021/Admn, dated: 08.06.2022

3. Email from Chairperson, Board of Studies in Computer Science and Application UG dated 08.06.2022

4. Remarks of Dean, Faculty of Science dated 13.06.2022

5. Orders of the Vice Chancellor in the file of even no. dated 14.06.2022

ORDER

1. The regulations for the Integrated Programmes under Choice Based Credit Semester System (CBCSS) in affiliated colleges w.e.f. 2020 admission was implemented, vide paper read (1) above.

2. Steering Committee on Integrated programmes held on 23.05.2022, vide paper read (2) above, resolved the following;

- To direct the Board of Studies concerned to change the course code of the Complementary courses (CBCSS-UG) in accordance with the regulations for Integrated programmes , as some of the colleges have already taught the syllabus of complementary courses prepared in accordance with the CBCSS-UG - programmes instead of the allied core courses of their Integrated programmes and to request the Board of Studies to approve the same as an allied core course of the Integrated programmes.

3. Accordingly, Chairperson, Board of Studies in Computer Science and Application UG vide paper read (3) above, has forwarded the new syllabus of Computer Science - Allied Core course for Integrated MSc Statistics programme for 2020 & 2021 Admission after circulating among the board members as per Chapter 3(34) of Calicut University First Statute, 1976.

4. The Dean, Faculty of Science, vide paper read (4) above and the Vice Chancellor, subject to ratification by the Academic council, vide paper read (5) above , approved the Scheme and syllabus of Computer Science - Allied Core course to Integrated MSc Statistics programme in tune with the regulations for the Integrated Programmes under Choice Based Credit Semester System (CBCSS) in affiliated colleges for 2020 & 2021 Admissions only.

5. The Scheme and syllabus of Computer Science - Allied Core course to Integrated

MSc Statistics programme in tune with the regulations for the Integrated Programmes under Choice Based Credit Semester System (CBCSS) in affiliated colleges for 2020 & 2021 admission is therefore implemented, subject to ratification by the Academic Council.

6. Orders are issued accordingly. (Syllabus appended)

Abdussamad M

Assistant Registrar

To

The Principals of all Affiliated Colleges offering Integrated MSc Integrated Statistics programme

Copy to: PS to VC/PA to PVC/ PA to Registrar/PA to CE/CDC/DoA/Digital Wing/JCE I/JCE IV/JCE V/SF/DF/FC

Forwarded / By Order

Section Officer

UNIVERSITY OF CALICUT

THENHIPALAM, CALICUT UNIVERSITY P.O



SYLLABUS

FOR

COMPUTER SCIENCE-DATA SCIENCE (ALLIED CORE)

FOR INTEGRATED MSc STATISTICS PROGRAMME

UNDER THE

FACULTY OF SCIENCE

FOR 2020 & 2021 ADMISSIONS ONLY

BOARD OF STUDIES IN COMPUTER SCIENCE (UG)

THENHIPALAM, CALICUT
UNIVERSITY P.O. KERALA, 673 635,
INDIA
DECEMBER, 2020

© COPYRIGHT BY UNIVERSITY OF CALICUT, 2020

REGULATIONS

FOR

COMPUTER SCIENCE- DATA SCIENCE

(ALLIED CORE)

FOR 2020 & 2021 ADMISSIONS ONLY

COURSE EVALUATION:

Total marks for each complementary theory course is 75 marks and lab course shall be 100 marks. The evaluation scheme for each course shall contain two parts (1) Internal evaluation (2) external evaluation. 15 marks shall be given to the internal evaluation of theory course. The remaining 60 marks shall be for the external evaluation.

INTERNAL EVALUATION

The internal assessment shall be based on a predetermined transparent system involving writtentest, assignments, seminars and attendance in respect of theory courses and on test/record/viva/ attendance in respect of lab courses. 15 marks in each theory course, 20 for lab, are for internal examinations.

Components with percentage of marks of Internal Evaluation of Theory Courses are:

Test paper	–	6 Marks
Attendance	–	3 Marks
Assignment	–	3 Marks
Seminar	–	3 Marks

Components with percentage of marks of Internal Evaluation of Lab Courses are:

Test paper (50%)	–	10 Marks
Attendance (20%)	–	4 Marks
Assignment/Lab involvement (30%)	–	6 Marks

Split up of marks for Test paper:

	Theory	Practical
85% -100%	6	10
65% -85%	5	8
55% - 65%	4	6
45% - 55%	3	5
35%- 45%	2	4
Less than 35%	1	2

Attendance of each course will be evaluated as below

	Theory	Practical
85 % and above	3	4
75% \leq CRP < 85%	2	2
50% \leq CRP < 75%	1	1

(If a fraction appears in the final internal marks awarded, nearest whole number is to be taken)

To ensure transparency of the evaluation process, the internal assessment marks awarded to the students in each course in a semester shall be notified on the notice board at least one week before the commencement of external examination. There shall not be any chance for improvement for internal marks. The course teacher(s) shall maintain the academic record of each student registered for the course, which shall be forwarded to the University by the college Principal after obtaining the signature of both course teacher and HOD. The marks secured for internal examination only need be sent to university, by the colleges concerned.

EXTERNAL EVALUATION:

There shall be University examinations for each theory course at the end of each semester. Practical examinations shall be conducted by the University at the end of fourth semester. External evaluation carries 60 Marks for each theory course, 80 Marks for lab course. External evaluation of even (2 and 4) semesters will be conducted in centralized valuation camps immediately after the examination. Answer scripts of odd semester (1 and 3) examination will be evaluated by home

valuation. All question papers shall be set by the university. The model of the question paper for external examination (theory courses) of 2 Hrs. duration. The students can answer all the questions in Sections A and B. But there shall be Ceiling in each section.

Section A: Short answer type carries 2 marks each - 12 questions (**Ceiling - 20**)

Section B: Paragraph/ Problem type carries 5 marks each - 7 questions (**Ceiling - 30**)

Section C: Essay type carries 10 marks (1 out of 2) - (**Ceiling - 10**)

The external examination in theory courses is to be conducted with question papers set by external experts. The evaluation of the answer scripts shall be done by examiners based on a well-defined scheme of valuation and answer keys shall be provided by the University. The external examination in practical courses shall be conducted by two examiners, one internal and an external, appointed by the University.

The model of the question paper for external examination (lab courses) of 3 Hrs. duration shall be:

1. **Section A:** One marked question of 30 Marks from Programming Lab Part A (Python Programming) is to be attempted (Design - Algorithm/Flowchart/Interface: 10 Marks, Code: 10 Marks, Result: 10 Marks. **Total 30 Marks**)
2. **Section B:** One marked question of 30 Marks from Programming Lab Part B (MongoDB or Tableau) is to be attempted (Design - Algorithm/Flowchart/Interface: 10 Marks, Code: 10 Marks, Result: 10 Marks. **Total 30 Marks**)
3. **Section C:** Lab viva voce (**Total 10 Marks**)
4. **Section D:** Lab Record (**Total 10 Marks**)

Number of students in one batch for practical Examination must be limited to 14.

REVALUATION:

In the new system of grading, revaluation is permissible. The prevailing rules for revaluation are applicable. Students can apply for photocopies of answer scripts of external examinations. Applications for photocopies/scrutiny/revaluation should be submitted within 10 days of publication of results. The fee for this shall be as decided by the university.

IMPROVEMENT COURSE

A maximum of two courses (Common, Core, Complementary or Open) can be improved in each semester. Improvement of a particular semester can be done only once. The student shall avail the improvement chance in the succeeding year after the successful completion of the semester concerned. The internal marks already obtained will be carried forward to determine the

grades/marks in the improvement examination. If the candidate fails to appear for the improvement examination after registration, or if there is no change in the results of the improvement examination appeared, the marks/grades obtained in the first appearance will be retained. Improvement and supplementary examinations cannot be done simultaneously.



ALLIED CORE “ DATA SCIENCE” FOR INTEGRATED MSc STATISTICS PROGRAMME

(FOR 2020 & 2021 ADMISSIONS ONLY)

Total Courses: 5				Total Credits: 12						
Semester	Course No	Course Code	Course Title	Marks			Contact Hours			Credits
				Internal	External	Total	Theory	Lab	Total	
I	1	BCS1IC01	Python Programming	15	60	75	2	2	4	2
II	2	BCS2IC02	Introduction to Data Science	15	60	75	2	2	4	2
III	3	BCS3IC03	Database Management System	15	60	75	3	2	5	2
IV	4	BCS4IC04	Data Visualization	15	60	75	3	2	5	2
IV	5	BCS4IC05	Programming Lab: Python Programming and Database Management System Lab/ Data Visualization using Tableau	20	80	100	0	0	0	4
Total (5 Courses)						400				12

Item	Description
L	Lab Hours
T	Theory Hours

BCS1IC01 – Python Programming

Semester: 1

Course Number: 1

Contact Hours: 2Theory+2Lab

Number of Credits: 2

Number of Contact Hours: 64

Course Evaluation: Internal – 15 Marks + External – 60 Marks

Aim of the Course:

To impart the students with problem solving skills , dividing the problem into subtasks and solving developing programming skills, logical thinking.

Objectives of the Course:

To learn the basics of computer programming and logic

To acquire basic soft skills

Prerequisites:

Background of the basic knowledge of Computer science at +2 level

Course Outcome

At the end of the course, the student will be able to

Explain basic principles of Python programming language

Implement object oriented concepts,.

Implement GUI applications.

Course Outline

UNIT I [7T+6L]

Problem solving strategies – Problem analysis – formal definition of problem – Solution – Top- down design – breaking a problem into sub problems- overview of the solution to the sub problems by writing step by step procedure (algorithm) - representation of procedure by flowchart - Implementation of algorithms – use of procedures to achieve modularity. Examples for algorithms and flow charts - At least ten problems Starting with non-numerical examples, and numeric problems like factorial, largest among three numbers, largest among N, Fibonacci

UNIT II [6T+7L]

Introduction to Python – Variables, Data types, Expressions and Statements, Evaluation of Expressions, Operators and Operands, Order of precedence, String Operations, Control statements, Boolean. Expressions and logical operators, conditional and alternative executions, Nested Conditionals, Recursion. Iteration - Multiple Assignment, While Statement, Tables, Two Dimensional Tables. Encapsulation and generalization, Local Variables.

UNIT III [6T+7L]

Introduction to NumPy: The Basics of NumPy Arrays, Computation on NumPy Arrays: Universal Functions, Aggregations: Min, Max, and Everything in Between, Computation on Arrays: Broadcasting, Comparisons, Masks, and Boolean Logic, Fancy Indexing, Sorting Arrays, Structured Data: NumPy's Structured Arrays

UNIT IV [7T+6L]

Functions, Calling functions, Type conversion and coercion, composition of functions, Mathematical functions, User-defined Functions, Parameters and Arguments. Strings and Lists – string traversal and comparison with List operations with Examples, Tuples and dictionaries – Operations and Examples.

UNIT V [6T+6L]

Files and exceptions - Text files, Directories, Python – Modules, Python – JSON, Python- PIP, Lambda. Classes/Objects-Create class, Create Object, The `__init__()` Function, Python Object Oriented Principles, Object Methods, Self-Parameter, Python Exception Handling, Python MySQL, Python MongoDB

References

1. Downey, A. et al., How to think like a Computer Scientist: Learning with Python, John Wiley, 2015
2. Lambert K. A., Fundamentals of Python - First Programs, Cengage Learning India, 2015
3. Sprankle, M., Problem Solving & Programming Concepts, Pearson India

B C S 2 I C 0 2 – Introduction to Data Science

Semester: 2

Course Number: 2

Contact Hours: 2T+2L

Number of Credits: 2

Number of Contact Hours: 64

Course Evaluation: Internal – 15 Marks + External – 60 Marks

Aim of the Course:

To impart the students with the basic concepts of Data Science, Big Data, Statistical Modeling, and Machine Learning.

Objectives of the Course:

- To learn the basic concepts of Machine Learning
- To learn the basics of Data Science and Data Analysis
- To learn the basics of Statistical Modeling

Prerequisites:

Background of the basic Statistics and Programming skills

Course Outcome

Students will demonstrate proficiency with statistical **analysis of data**

Students will apply data science concepts and methods to **solve** problems in real-world contexts

Course Outline

UNIT I [7T+6L]

Introduction: What is Data Science, Big Data and Data Science hype and getting past the hype Why now. Datafication, Current landscape of perspectives, Skill sets needed, Statistical Inference, Populations and samples, Statistical modeling, probability distributions, Introduction to R.

UNIT II [6T+7L]

Exploratory Data Analysis and the Data Science Process, Basic tools (plots, graphs and summary statistics) of EDA, Philosophy of EDA, The Data Science Process Case Study: RealDirect (online real

estate_rm), Three Basic Machine Learning Algorithms, Linear Regression, k-Nearest Neighbors (k-NN), k-means

UNIT III [6T+7L]

Additional Machine Learning Algorithm and Usage in Applications, Motivating application: Filtering Spam, Why Linear Regression and k-NN are poor choices for Filtering Spam, Naive Bayes and why it works for Filtering Spam, Data Wrangling: APIs and other tools for scrapping the Web, Feature Generation and Feature Selection (Extracting Meaning From Data), Motivating application: user (customer) retention, Feature Generation (brainstorming, role of domain expertise, and place for imagination), Feature Selection algorithms, Filters; Wrappers; Decision Trees; Random Forests

UNIT IV [7T+6L]

Recommendation Systems: Building a User-Facing Data Product, Algorithmic ingredients of a Recommendation Engine, Dimensionality Reduction, Singular Value Decomposition, Principal Component Analysis, Exercise: build your own recommendation system, Mining Social-Network Graphs, Social networks as graphs, Clustering of graphs, Direct discovery of communities in graphs, Partitioning of graphs, Neighborhood properties in graphs.

UNIT V [6T+6L]

Data Visualization, Basic principles, ideas and tools for data visualization, Examples of inspiring (industry) projects. Exercise: create your own visualization of a complex dataset, Data Science and Ethical Issues, Discussions on privacy, security, ethics. A look back at Data Science, Next-generation data scientists.

References

1. Cathy O'Neil and Rachel Schutt. Doing Data Science, Straight Talk From The Frontline. O'Reilly. 2014.
2. Jure Leskovek, Anand Rajaraman and Jeffrey Ullman. Mining of Massive Datasets. v2.1, Cambridge University Press. 2014. (free online)
3. Kevin P. Murphy. Machine Learning: A Probabilistic Perspective. ISBN 0262018020. 2013.
4. Foster Provost and Tom Fawcett. Data Science for Business: What You Need to Know about Data Mining and Data-analytic Thinking. ISBN 1449361323. 2013.
5. Trevor Hastie, Robert Tibshirani and Jerome Friedman, Elements of Statistical Learning, Second Edition. ISBN 0387952845. 2009. (free online)
6. Avrim Blum, John Hopcroft and Ravindran Kannan. Foundations of Data Science.
7. Mohammed J. Zaki and Wagner Miera Jr. Data Mining and Analysis: Fundamental Concepts and Algorithms. Cambridge University Press. 2014.
8. Jiawei Han, Micheline Kamber and Jian Pei. Data Mining: Concepts and Techniques, Third Edition. ISBN 0123814790. 2011

BCS3IC03 – Database Management System

Semester: 3

Course Number: 3

Contact Hours: 3T+2L

Number of Credits: 2

Number of Contact Hours: 80 Hrs.

Course Evaluation: Internal – 15 Marks + External – 60 Marks

Aim of the Course:

1. To Gain knowledge of database systems and database management systems software.
2. Ability to model data in applications using conceptual modelling tools such as ER Diagrams and design data base schemas based on the model.
3. Formulate, using SQL, solutions to a broad range of query and data update problems.
4. Demonstrate an understanding of normalization theory and apply such knowledge to the normalization of a database..

Objectives of the Course:

- To learn the Database concepts
- To learn the MongoDB database

Prerequisites:

Background of the basic Programming

Course Outcome

Apply the database concepts and design database for given information system.

Create database and develop database programming skills in SQL and PL/SQL.

Analyse the concepts of Normalization and design database which possess no anomalies

4. Explain the issues of transaction like concurrency control, recovery and security.

Course Outline

UNIT I [10T+6L]

Database System concepts and applications, Introduction to databases, File Systems vs. DBMS, Advantages and Disadvantages of using DBMS Approach, Database administrators and user, Data Models, Schemas, and Instances, Types of Data Models, Three Schema Architecture and Data Independence, Database Languages and Interfaces.

UNIT II [10T+6L]

Entity-Relationship Model - Conceptual Data Models for Database Design, Entity Relationship Models, Concept of Entity, Entity Sets, Relationship Sets, Attributes, Domains, Constraints, Keys, Strong and Weak Entities, Concepts of EER. Relational Data Model Relations, Domains and Attributes, Tuples, Keys. Integrity Rules, Relational Algebra and Operations, Relational Calculus and Domain Calculus, Relational Database Design using ER to Relational Mapping.

UNIT III [10T+6L]

Relational Database Design - Relational database design Anomalies in a Database, Normalization Theory, Functional Dependencies, First, Second and Third Normal Forms, Relations with more than one Candidate Key, Good and Bad Decompositions, Boyce Codd Normal Form, Multivalued Dependencies and Fourth Normal Form, Join Dependencies and Fifth Normal Form.

UNIT IV [8T+8L]

SQL Concepts: Basics of SQL, DDL, DML, DCL, Tables – Create, Modify and Delete table structures, Rename and Drop tables, Defining constraints – Primary key, foreign key, unique, not null, check, IN operator Select Command, Logical Operators, Functions - aggregate functions, Built-in functions –numeric, date, string functions, set operations, sub-queries, correlated sub-queries, Use of group by, having, order by, join and its types, Exist, Any, All. View - Creation, Renaming the column of a view, destroys view.

UNIT V [8T+8L]

Transaction Management and Concurrency Control - Transaction Properties (ACID), states, Commit, Rollback; Concurrency Control Lost update problems, Locks, two phase locking. Introduction to MongoDB and its Features, Database, Collection and Documents, Various Data Types in MongoDB, Introduction to mongo shell.

References

1. Abraham Silberschatz, Henry F Korth, S.Sudharshan, Database System Concepts, 6th Edition
2. W. Gilmore, Beginning PHP and PostgreSQL 8: From Novice to Professional, Goels Computer Hut (2007), ISBN: 9788181286000
3. Karl Seguin, The little MongoDB Book
4. CJ Date, Introduction to Database Systems, Addison Wesley.

MongoDB Online References

i http://mdslab.unime.it/sites/default/files/mongodb_tutorial.pdf

li <https://pdfs.semanticscholar.org/presentation/066b/600837053e0172ea3e514efaaa3bd8485e46.pdf>

BCS4IC04– Data Visualization

Semester: 4
CourseNumber: 4

ContactHoursperWeek: 5(3T+2L)

Number of Credits:2

Number of Contact Hours: 80 Hrs

CourseEvaluation: Internal – 15 Marks+ External – 60 Marks

Aim of the Course:

To impart students about Computational Statistics and Data Visualization

To make the students aware of Data visualization tools

Objectives of the Course:

- To introduce the concept of statistical analysis of data.
- To learn Tableau

Prerequisites of the Course:

Knowledge of Data Science and DBMS

Course Outcome

- Students will execute statistical analyses with professional statistical software
- Students will be able to prepare reports using data visualization tools to make a cohesive narrative of the problem under scrutiny and offer guidance based on data insights

UNIT I [10T+6L]

Computational Statistics and Data Visualization, Data Visualization and Theory, Presentation and Exploratory Graphics, Graphics and Computing, Statistical Historiography Good Graphics –Introduction, Content, Context and Construction, Presentation Graphics and Exploratory Graphics, Presentation (What to Whom, How and Why), Choice of Graphical Form, Graphical Display Options, Higher-dimensional Displays and Special Structures, Scatterplot Matrices (Sploms), Parallel Coordinates, Mosaic Plots, Small Multiples and Trellis Displays, Time Series and Maps.

UNIT II [10T+6L]

Complete Plots, Sensible Defaults, Customization-Setting Parameters, Arranging Plots, Annotation, Extensibility-Building Blocks, Combining Graphical Elements, 3-D Plots, Speed, Output Formats, Data Handling Data and Graphs, Graph Layout Techniques- Force-directed Techniques, Multidimensional Scaling, The Pulling Under Constraints Model, Bipartite Graphs Graph Drawing, Hierarchical Trees, Spanning Trees, Networks, Directed Graphs, Treemaps.

UNIT III [10T+6L]

High-dimensional Data Visualization Introduction, Mosaic Plots, Associations in High-dimensional Data, Response Models, Models, Trellis Displays, Definition, Trellis Display vs. Mosaic Plots, Visualization of Models, Parallel Coordinate Plots, Geometrical Aspects vs. Data Analysis Aspects, Limits Multidimensional Scaling Proximity Data, Metric MDS, Non-metric MDS, Example: Shakespeare Keywords, Procrustes Analysis, Unidimensional Scaling, INDSCAL, Correspondence Analysis and Reciprocal Averaging, Large Data Sets and Other Numerical Approaches

UNIT IV [8T+8L]

Tableau. Introduction- Environmental setup, Design Flow, File Types, Data Types. Data Sources, Custom Data View, Extracting Data, Field operations, Metadata, Data Joining and Blending, Worksheets - Adding, renaming, reordering Worksheet Pages, Workbook Calculations- Operators, functions, Calculations, LOD Expressions. 2 9

UNIT V [8T+8L]

Sort and Filters- Sorting, Quick filtering, Context filtering, Condition filtering, Filter operations, Charts, Advanced tableau, Tableau — Bar Chart, Line Chart, Multiple Measure Line Chart, Pie Chart, Crosstab, Scatter Plot, Bubble Chart, Bullet Graph, Box Plot. Dashboard, Forecasting

Text Books

1. Handbook of Data Visualization by Chun-houh Chen, Wolfgang Härdle, Antony Unwin
2. Learning tableau by Joshua N. Milligan
3. The Visual Display of Quantitative Information by Edward R. Tufte

References

1. The Functional Art by Alberto Cairo
2. Tableau Dashboard Cookbook by Jen Stirrup



BCS4IC05 -Programming Lab: Python Programming and MongoDB/Tableau

Semester: 4

Course Number: 5

Number of Credits: 4

Course Evaluation: Internal – 20 Marks + External – 80 Marks

Aim of the Course:

- To provide practical skill in Python Programming, MongoDB, Tableau

Objectives of the Course:

- To develop Python Programming skills
- To make the students equipped to solve mathematical or scientific problems using Python
- To learn how to implement various data Visualization Implementation using Tableau

Prerequisites:

- Background of the basic computing knowledge

Course Outcome

- Students will apply data science concepts and methods to **solve** problems in real-world contexts and will **communicate** these solutions effectively

Course Outline

LAB PROGRAM CYCLE

Python Programming

1. Program to demonstrate basic data types in python
2. Program to demonstrate operators in python.
3. A cashier has currency notes of denominations 10, 50, and 100. If the amount to be withdrawn is input through the keyboard using input () function in hundreds, find the total number of currency notes of each denomination the cashier will have to give to the withdrawer.
4. Program to demonstrate list and tuple in python.
5. A library charges a fine for every book returned late. For first 5 days the fine is 50 paisa, for 6-10 days fine is one rupee and above 10 days fine is 5 rupees. If you return the book after 30 days your membership will be cancelled. Write a program to accept the number of days the member is late to return the book and display the fine or the appropriate message

6. Write a program to calculate overtime pay of 10 employees. Overtime is paid at the rate of Rs.12.00 per hour for every hour worked above 40 hours. Assume that employee do not work for fractional part of an hour.
7. Two numbers are entered through the keyboard. Write a program to find the value of one number raised to the power of another.
8. Write a function that receives marks received by a student in 3 subjects and returns the average and percentage of these marks. Call this function from main () and print the result in main.
9. Write a program to read a file and display its contents.
10. Write a program to demonstrate database connectivity in python.
11. Write a program to compute distance between two points taking input from the user (Pythagorean Theorem).
12. Write a program add.py that takes two numbers as command line arguments and prints its sum.
13. Write a Program for checking whether the given number is an even number or not.
14. Using a for loop, write a program that prints out the decimal equivalents of $1/2$, $1/3$, $1/4$.
15. Write a program using a while loop that asks the user for a number, and prints a countdown from that number to zero
16. Write a Python program to print Fibonacci series.
17. Write a program to count the numbers of characters in the string and store them in a dictionary data structure.
18. Write a program to use split and join methods in the string and trace a birthday with a dictionary data structure.
19. Write a program that combines the lists into a dictionary.
20. Write a program to count frequency of characters in a given file. Can you use character frequency to tell whether the given file is a Python program file, C program file or a text file.
21. Write a program to print each line of a file in reverse order.
22. Write a program to compute the number of characters, words and lines in a file.
23. Write a function that takes two balls as parameters and computes if they are colliding. Your function should return a Boolean representing whether or not the balls are colliding.
Hint: Represent a ball on a plane as a tuple of (x, y, r), r being the radius
If (distance between two balls centers) \leq (sum of their radii) then (they are colliding)
Find mean, median, mode for the given set of numbers in a list.
24. Write a function nearly equal to test whether two strings are nearly equal. Two strings a and b are nearly equal when a can be generated by a single mutation on b.
25. Write a function dups to find all duplicates in the list.
26. Write a function unique to find all the unique elements of a list.
27. Write a function cumulative product to compute cumulative product of a list of numbers.
28. Write function to compute gcd and lcm of two numbers.
29. Write a function reverse to reverse a list without using the reverse function.
30. Write a program that defines a matrix and prints.
31. Write a program to perform addition of two square matrices.
32. Write a program to perform multiplication of two square matrices.
33. Write a Python Program to connect to MySQL database.

34. Write a Python Program to insert and delete records.
35. Write a Python Program to search and update records

DBMS-MongoDB

1. Create the following tables with the specified constraints:

Department:

Department name - Not NULL unique

Department number - Primary Key

ManagerId - Refers to employee-id of employee table.

Manager

date of joining - Not NULL.

Employee:

First name - Not NULL

Middle initials

Last name - Not NULL

Employee id - Primary Key

Date of Birth - Not NULL

Address

Gender - M or F

Salary - Range of 5000 to 25000

Date of Joining

Department number - Refers to Department Number of Department table.

Department location:

Department number - Refers to Department number of department table.

Department location - Not NULL.

Department number & Department location are combined Primary Key

Project:

Project name-Not NULL.

Project number-Primary Key.

Project location-Not NULL.

Department number-Refers to department number of Department table.

Works-on:

Employee-id - Not NULL refers to employee-id of employee table.

Project number- Not NULL refers to Project number of Project table.

Hours - Not NULL.

Employee-id & Project number are combined primary key.

Dependent:

Employee-id - Refer to employee table employee id field

Dependent name - Gender - M or F

Date of Birth - Not NULL

Relationship - Not NULL

Now enter a few sets of meaningful data and answer the following queries.

1. List the department wise details of all the employees.

2. Find out all those departments that are located in more than one location.
3. Find the list of projects.
4. Find out the list of employees working on a project.
5. List the dependents of the employee whose employee id is 001

2. Create the following tables with specified fields.

Book Records:

Accession Number

ISBN Number

Books:

ISBN Number

Author

Publisher

Price

Members:

Member Id

Member Name

Maximum Number of books that can be issued

Maximum Number of days for which book can be issued

Book Issue:

Member Id

Accession Number

Issue Date

Return Date

You must create constraints, including referential integrity constraints, as appropriate. Please note accession number is unique for each book. A book, which has no return date, can be considered as issued book. Enter suitable data into the tables. Now answer the following:

1. Insert data in all the three tables (use insert).
2. Insert appropriate description associated with each table and the column (use comment).
3. Display the structure of the tables.
4. Display the comments that you have inserted.
5. Using SELECT statement, write the queries for performing the following function:
 - (a) Get the list of all books (No need to find number of copies).
 - (b) Get the list of all members.
 - (c) Get the Accession number of the books which are available in the library.
 - (d) On return of a book by a member calculate the fine on that book.
 - (e) List of books issued on 01-Jan-2021.
 - (f) Get the list of all books having price greater than Rs. 500/-
 - (g) Get the list of members who did not have any book issued at any time.
 - (h) Get the list of members who have not returned the book.
 - (i) Display member ID and the list of books that have been issued to him/her from time to time.
 - (j) Find the number of copies of each book (A book accession number would be different but ISBN number would be the same).
 - (k) Find the number of copies available of a book of given ISBN number.

(I) Get the member ID and name of the members to whom no more books can be issued, because they have already got as many books issued as the number for which they are entitled.

3. Create appropriate forms, reports, graphs, views and data filtering, etc. to answer the following queries.

1. Get the list of ISBN-Number, Book name, available copies of the books of which available copies are greater than zero.
2. Get the list of ISBN-Number, Book name, Total copies, available copies of the book of which available copies are greater than zero. List should be displayed in alphabetical order of book name.
3. Get the list of ISBN number, Book name, Author, total copies, cost (cost is price total copies). List should be displayed in descending order of cost.
4. Get the list of books issued to each member.
5. Write query to know the maximum and average price of the books.
6. Get the list of all existing members and the number of days for which a member is allowed to keep the book. Also find out the members who have got the maximum number of books issued.
7. Get the list of member codes of those members who have more than two books issued.
8. Find the details of the books presently issued to a member.
9. Create the history of issue of a book having a typical accession number.
10. To set the width of the book name as 35.

4 Create the CUSTOMER table and perform the following tasks

Column name	type	size
Customer id	Character	10
Name	Character	25
Area	Character	3
Phone	Numeric	7

Insert the appropriate data into table and do the following.

- Update Phone numbers of all customers to have a prefix as your city
- Print the entire customer table
- List the names of those customers who have *i* as second letter in their names.
- Find out the Customer belonging to area *abc*
- Delete record where area is NULL.
- Display all records in increasing order of name.
- Create a table temp from customer having customer-id, name, and area fields only
- Display area and number of records within each area (use GROUP by clause)
- Display all those records from customer table where name starts with *b* or area is *abc*.
- Display all records of those where name starts with *a* and telephone exchange is 55.

5. Create a view to know member name and name of the book issued to them. Use any inbuilt function and operators like IN, ANY, ALL, EXISTS.

- a. List the records of members who have not been issued any book using EXISTS operator.
- b. List the members who have got issued at least one book (use IN/ANY operator).
- c. List the books which have maximum Price using ALL operator.
- d. Display Book Name, Member Name, Issue date of Book. Create a view of this query of the currently issued books.

6. Create a table of Employee (emp-number, name, dept, salary) and Department (dept number, dept name). Insert some records in the tables through appropriate forms having integrity checks. Add some records in employee table where department value is not present in department table. Now answer the following query:
 - a. Display all records from employee table where department is not found in department table.
 - b. Display records from employee table in a report format with proper headings. This report must also contain those records where department number does not match with any value of department table.
 - c. Display those employee records who have salary less than the salary of person whose empcode= A100.
 - d. Create another table: SalesData (RegionCode, City, Salesperson-Code, SalesQty).
 - e. Display records where salesperson has achieved sales more than average sales of all sales persons of all the regions.

7. Create the following tables:

Order party: Order_number, Order_date, customer_code

Order: Order_number, Item_code, Quantity

The key to the second table is Order_number + Item_code

Create a form for data entry to both the tables.

8. Implement student information system

9. Implement hotel billing system

10. SQL scripts to display various reports like Result of an Examination, Salary Report, Sales Report, Sales reports grouped on Sales person or item, etc.

11. Create a STUDENT table with following fields:

Field Name	Type	Width
regno	character	10
name	character	20
paper1	numeric	3
paper2	numeric	3
paper3	numeric	3
paper4	numeric	3
paper5	numeric	3
total	numeric	3
result	character	6
grade	character	1



Enter the regno, name and marks in 5 papers of at least 10 students. Write a program to process the records to update the table with values for the fields total (paper1+paper2+paper3+paper4+paper5), result('passed' if total is greater than or equal to 50% of the total; 'failed' otherwise), and grade ('A' if mark obtained is greater than or equal to 90% of the total mark, 'B' if mark obtained is greater than or equal to 75% of the total mark, 'C' if mark obtained is greater than or equal to 60% of the total mark, 'D' if mark obtained is greater than or equal to 50% of the total mark, and 'F' if mark obtained is less than 50% of the total mark). Display a report in descending order of the total mark, showing the data entered into the table along with the total marks, result and grade.

12. Generate a Hospital information system that can generate the following reports:

- Patients who belongs to in-patient category

- Patients who belongs to out-patient category

For this purpose, create a table named HOSPITAL having the following structure.

Field Name	Type	Width
patientid	character	10
name	character	20
age	numeric	3
doctor	character	20
patienttype	character	15
consultcharge	numeric	6
bloodtestcharge	numeric	6
xraycharge	numeric	6
othercharges	numeric	6
totalamount	numeric	6

Enter the records of at least 10 patients. Write a SQL program to display the report in the ascending order of patient name.

13. Create the tables EMP and DEPT using the following Information.

a. DEPT:

COLUMN NAME	DATA TYPE(SIZE)
DEPTNO	NUMBER (2)
DNAME	VARCHAR2 (14)
LOC	VARCHAR2 (14)

b. EMP:

COLUMN NAME	DATATYPE(SIZE)
EMPNO	NUMBER (4)
ENAME	VARCHAR2 (10)
JOB	VARCHAR2 (9)
MGR	NUMBER (4)
HIREDATE	DATE
SAL	NUMBER (7, 2)
COMM	NUMBER (7, 2)
DEPTNO	NUMBER (2)



a) Check the Default Size of a Number, Char and Date Data types.

b) Describe the Structure of the Table and EMP Table.

14. Design a Hotel Bill calculating system that generates hotel bills for the customers.

15. Design a Hostel Accounting system that generates the Hostel Due Report.

16. Design an Electricity Bill Report generating system that generates electricity bills details of customers for a month.

17. Generate a Library Information System that generates report of the books available in the library.

18. Programs involving multiple tables.

19. Consider the schema for College Database:

STUDENT (USN, SName, Address, Phone, Gender)

SEMSEC (SSID, Sem, Sec)

CLASS (USN, SSID)

SUBJECT (Subcode, Title, Sem, Credits)

IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3, FinalIA)

Write SQL queries to

1. List all the student details studying in fourth semester 'C' section.
2. Compute the total number of male and female students in each semester and in each section.
3. Create a view of Test1 marks of student USN '1BI15CS101' in all subjects.
4. Calculate the FinalIA (average of best two test marks) and update the corresponding table for all students.
5. Categorize students based on the following criterion:

If FinalIA = 17 to 20 then CAT = 'Outstanding'

If FinalIA = 12 to 16 then CAT = 'Average'

If FinalIA < 12 then CAT = 'Weak'

Give these details only for 8th semester A, B, and C section students.

20. Consider the schema for Company Database:

EMPLOYEE (SSN, Name, Address, Sex, Salary, SuperSSN, DNo)

DEPARTMENT (DNo, DName, MgrSSN, MgrStartDate)

DLOCATION (DNo, DLoc)

PROJECT (PNo, PName, PLocation, DNo)

WORKS_ON (SSN, PNo, Hours)

Write SQL queries to

1. Make a list of all project numbers for projects that involve an employee whose last name is 'Scott', either as a worker or as a manager of the department that controls the project.
2. Show the resulting salaries if every employee working on the 'IoT' project is given a 10 percent raise.
3. Find the sum of the salaries of all employees of the 'Accounts' department, as well as the maximum salary, the minimum salary, and the average salary in this department
4. Retrieve the name of each employee who works on all the projects controlled by department number 5 (use NOT EXISTS operator). For each department that has more than five employees, retrieve the department number and the number of its employees who are making more than Rs. 6,00,000.

Data Visualization using Tableau

1. To load and display dataset on tableau.
2. To join tables from single and multiple databases.
3. Implement Data Preparation using Blending
4. Implement Data Preparation using Metadata Grid.
5. Implement Data Preparation using Pivot.
6. Implement Data Preparation using Union.
7. Implement Data Preparation using Data Interpreter.
8. To build groups, hierarchies and sets.
9. Implement Sorting using measure.
10. Implement Sorting using header and legends.
11. Implement sorting using pill(use-case).



12. Implement Interactive Filter,
13. Implement Quick Filter.
14. Implement context filtering.
15. To plot a graph to show the Data in histogram.
16. To plot a bullet graph.
17. To plot a dual axis graph.
18. To plot an area graph with dual axis.
19. To show Data in TreeMap.
20. To build a Box plots.
21. To build a Gantt Bar Charts.
22. To build a Paretos.
23. To build a Sparklines.
24. Implement Grouping using Data Window and Visual Grouping,
25. Implement using all Functions (Number, String, Date, Logical, Aggregate).
26. Create quick table calculations. 27. Implement LOD calculations; types of LOD calculations. 28. Implement Ad-hoc calculations.
29. Implement mapping using Pan & Zoom.
30. Implement mapping using Filtering.
31. Implement mapping using Map layering.
32. Implement mapping using Custom territories.
33. Implement mapping using Lasso & Radial selection.
34. To modify locations within Tableau.
35. To import and manage custom geocoding.
36. Create a custom territory, building a polygon map.
37. To use a background image map
38. To use Geographic search.
39. Implement analytics using Reference Lines and Reference Bands.
40. Implement analytics using Trend Lines and Trend Model.
41. Implement analytics using Forecasting using precise range.
42. Implement analytics using Drag & Drop Analytics and Box Plot.
43. Create a dashboard for publishing & sharing options.
44. Implement an interactive dashboard using actions (Filter, Highlight, UR



FIRST SEMESTER B.Sc. DEGREE EXAMINATION NOVEMBER 2020
(CBCSS-UG)
COMPLEMENTARY COURSE – COMPUTER SCIENCE DATA SCIENCE
BCS1IC01 – Python Programming

2 hrs

60 marks

BCS1IC01 – Python Programming

Section A – Short Answer type questions

(Answer all questions, each correct answer carries a maximum of 2 Marks. Ceiling 20 marks)

1. What you meant by operator precedence?
2. What is the difference between list and tuples in Python?
3. What is an identifier? Give examples
4. What is Python? What are the benefits of using Python?
5. Write a short note on global keyword in Python.
6. What is Random Number Generator in Python?
7. Write a Python program to calculate the length of a string?
8. What is a set in Python?
9. What is default parameter in Python?
10. How do you write comments in Python?
11. What is eval() in python? What is its syntax?
12. Write a Python program to check given number is Prime or not.

Section B – Short Essay type questions

(Answer all questions, each correct answer carries a maximum of 5 marks. Ceiling 30 marks)

13. What are the different data types available in Python?
14. Write a Python program to remove an item from a set if it is present in the set.
15. Explain the use of functions in Python in detail.
16. What is a tuple? Explain different tuple operations in Python.
17. Write a short note on Python IDLE.
18. Write a Python program to sum all the items in a list.
19. Explain recursion in Python with suitable examples?

Section C – Essay type questions

(Answer any one questions, correct answer carries a maximum of 10 marks)

20. What is a function? Explain different functions available in Python with suitable examples.
21. What is a loop? Explain different looping statement in Python.