SEMESTER -3

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
20MCA201	DATA SCIENCE & MACHINE LEARNING	CORE	3	1	0	4

Preamble: This is an introductory course on data science and basic concepts behind various machine learning techniques. Machine learning is the study of adaptive computational systems that improve their performance with experience. At the end of the course the students should be able to design and implement machine learning solutions to classification, regression, and clustering problems and to evaluate and interpret the results of the algorithms.

Prerequisite: Probability and Statistics, Linear Algebra, Programming in Python/R.

Course Outcomes: After the completion of the course the student will be able to:

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Discuss the fundamental concepts of data science and data visualization techniques.	Level 2: Understand
CO 2	Explain the basics of machine learning and use lazy learning and probabilistic learning algorithms to solve data science problems.	Level 3: Apply
CO 3	Describe decision trees, classification rules & regression methods and how these algorithms can be applied to solve data science problems.	Level 3: Apply
CO 4	Solve data science problems using neural networks and support vector machines.	Level 3: Apply
CO 5	Discuss clustering using k-means algorithm and evaluate & improve the performance of machine learning classification models.	Level 3: Apply

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	1	1	-	-	-	-	3	-	-	-	-	-
CO 2	3	3	3	2	-	-	3	-	-	-	-	-
CO 3	3	3	3	2	-	-	3	-	-	-	-	-
CO 4	3	3	3	2	-	-	3	-	-	-	-	-
CO 5	3	3	3	2	-	-	3	-	-	-	-	-

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember (K1)	15	10	10
Understand (K2)	25	20	30
Apply (K3)	10	20	20
Analyse (K4)			
Evaluate (K5)			
Create (K6)			

Mark Distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks

Continuous Assessment Test (2 numbers) : 20 marks

Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be two parts: Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module, of which the student should answer any one. Each question can have a maximum of 2 subdivisions and carry 6 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):

1. What is data science and why do we need data science?
2. Explain the data science classification and illustrate data science tasks.
3. Describe the various methods to understand data.
4. Explain the typical methods to visualize data.

Course Outcome 2 (CO2)

1. Explain the differences between supervised and unsupervised machine learning algorithms.
2. Describe the key concepts that define nearest neighbour classifiers, and why they are considered "lazy" learners.
3. Explain how to apply *k*-NN classifier in a data science problem.
4. State Bayes' theorem in statistics. Outline the Naive Bayes algorithm to build classification models.
5. Use Naive Bayes algorithm to determine whether a red domestic SUV car is a stolen car or not using the following data:

Example	Colour	Type	Origin	Stolen?
1	red	sports	domestic	yes
2	red	sports	domestic	no
3	red	sports	domestic	yes
4	yellow	sports	domestic	no
5	yellow	sports	imported	yes
6	yellow	SUV	imported	no
7	yellow	SUV	imported	yes
8	yellow	SUV	domestic	no
9	red	SUV	imported	no
10	red	sports	imported	yes

Course Outcome 3 (CO3):

1. Classify data science tasks using decision trees and classification rule learners.
2. Discuss the various feature selection measures.
3. How to simplify a decision tree by pruning.
4. Describe how to construct classification rules from decision trees.
5. Explain the concepts of regression and correlation.
6. How to estimate a linear regression model.
7. Consider the following set of training examples:

Instance	Classification	a_1	a_2
1	+	T	T
2	+	T	T

3	-	T	F
4	+	F	F
5	-	F	T
6	-	F	T

a) Find the entropy of this collection of training examples with respect to the target function "classification"?
b) Calculate the information gain of $\mathbf{a_2}$ relative to these training examples?

Course Outcome 4 (CO4):

1. Explain how artificial neural networks mimic human brain to model arbitrary functions and how these can be applied to real-world problems.
2. Describe different activation functions and network topology.
3. Discuss basic idea behind the backpropagation algorithm.
4. Explain how a support vector machine can be used for classification of linearly separable data.
5. How to compute the distance of a point from a hyperplane.
6. How the kernel trick is used to construct classifiers in nonlinearly separated data.

Course Outcome 5 (CO5):

1. Explain how the clustering tasks differ from the classification tasks.
2. How clustering defines a group, and how such groups are identified by *k*-means clustering algorithm.
3. Find the three clusters after one epoch for the following eight examples using the *k*-means algorithm and Euclidean distance: A1 = (2,10), A2 = (2,5), A3 = (8,4), A4 = (5,8), A5 = (7,5), A6 = (6,4), A7 = (1,2), A8 = (4,9). Suppose that the initial seeds (centres of each cluster) are A1, A4 and A7.
4. Explain the various matrices used to measure the performance of classification algorithms
5. Explain the concepts of bagging and boosting.
6. Suppose 10000 patients get tested for flu; out of them, 9000 are actually healthy and 1000 are actually sick. For the sick people, a test was positive for 620 and negative for 380. For the healthy people, the same test was positive for 180 and negative for 8820. Construct a confusion matrix for the data and compute the precision and recall for the data.

Model Question Paper
Course Code: 20MCA201

Course Name: DATA SCIENCE AND MACHINE LEARNING

Max. Marks :60 **Duration: 3 Hrs**

Part A

Answer all questions. Each question carries 3 marks (10 * 3 = 30 Marks)

1. What is data science?
2. Explain the different types of data.
3. Differentiate between supervised and unsupervised learning algorithms.
4. Explain how to choose the value of *k* in *k*-NN algorithm.
5. Explain entropy and information gain.
6. Explain the Ordinary Least Square method in regression.
7. Define activation function. Give two examples.
8. What is maximum margin hyperplane.
9. Define precision, recall and F-measure.
10. Explain bootstrap sampling

Part B

Answer one full question from each module, each carries 6 marks.

11. Explain the various methods for visualising multivariate data. (6 marks)

OR

12. Explain the various processes for preparing a dataset to perform a data science task. (6 marks)

13. Based on a survey conducted in an institution, students are classified based on the two attributes of academic excellence and other activities. Given the following data, identify the classification of a student with X = 5 and Y = 7 using *k*-NN algorithm (choose *k* as 3).

X (Academic Excellence)	Y (Other Activities)	Z (Classification)
8	6	Outstanding
5	6	Good
7	3	Good
6	9	Outstanding

(6 marks)

OR

14. Given the following data on a certain set of patients seen by a doctor. Can the doctor conclude that a person having chills, fever, mild headache and without running nose has flu? (Use Naive Bayes classification).

Chills	Running nose	Headache	Fever	Has flu
Y	N	mild	Y	N
Y	Y	no	N	Y
Y	N	strong	Y	Y
N	Y	mild	Y	Y
N	N	no	N	N
N	Y	strong	Y	Y
N	Y	strong	N	N
Y	Y	mild	Y	Y

(6 marks)

15. Obtain a linear regression for the data given in the table below assuming that *y* is the independent variable.

x	55	60	65	70	80
y	52	54	56	58	62

(6 marks)

OR

16. Given the following data, draw a decision tree to predict whether a person cheats. Give the corresponding set of classification rules also.

Sl. No.	Refund	Marital status	Income	Cheats?
1	Yes	Single	High	No
2	No	Married	High	No
3	No	Single	Low	No
4	Yes	Married	High	No
5	No	Divorced	High	Yes
6	No	Married	Low	No

7	Yes	Divorced	High	No
8	No	Single	High	Yes
9	No	Married	Low	No
10	No	Single	High	Yes

(6 marks)

17. Define an artificial neuron. What are the characteristics of an artificial neural network (ANN)?

(6 marks)

OR

18. a) Define linearly separable dataset. Give an example each of a dataset that is linearly separable and of a dataset that is not linearly separable. (3 marks)

    b) Define kernel function. Explain the kernel trick to construct a classifier for a dataset that is not linearly separable. (3 marks)

19. Suppose 10000 patients get tested for flu; out of them, 9000 are actually healthy and 1000 are actually sick. For the sick people, a test was positive for 620 and negative for 380. For the healthy people, the same test was positive for 180 and negative for 8820. Construct a confusion matrix for the data and compute the precision and recall for the data. (6 marks)

OR

20. Assume the following: A database contains 80 records on a particular topic of which 55 are relevant to a certain investigation. A search was conducted on that topic and 50 records were retrieved. Of the 50 records retrieved, 40 were relevant. Construct the confusion matrix for the search and calculate the precision and recall scores for the search. (6 marks)

Syllabus

Module 1 (9 Hours)

Introduction to data science, Data science classification, Data science process - Prior knowledge, Data preparation, Modelling, Application, Data exploration - Data sets, Descriptive statistics for univariate and multivariate data

Data visualisation – Histogram, Quartile plot, Distribution chart, Scatter plot, Bubble chart, Density chart

Module 2 (9 Hours)

Introduction to machine learning: How machines learn - Data storage, Abstraction, Generalisation, Evaluation, Machine learning in practice - Types of machine learning algorithms.

Lazy learning: Classification using K-Nearest Neighbour algorithm - Measuring similarity with distance, Choice of k, Preparing data for use with k-NN.

Probabilistic learning: Understanding Naive Bayes - Conditional probability and Bayes theorem, Naive Bayes algorithm for classification, The Laplace estimator, Using numeric features with Naive Bayes.

Module 3 (9 Hours)

Decision tree learning: Concept of decision tree, Divide and conquer approach, C5.0 Decision tree algorithm, Choosing the best split, Pruning the decision tree.

Classification rules learning: Concept of classification rules, Separate and conquer approach, The 1R algorithm, Rules from decision trees.

Regression methods: Concept of regression, Simple linear regression, Ordinary least squares estimation, Correlations, Multiple linear regression.

Module 4 (9 Hours)

Neural network learning: Artificial neurons, Activation functions, Network topology, Training neural networks with backpropagation.

Support vector machines: Hyperplanes, Classification using hyperplanes, Maximum margin hyperplanes in linearly separable data, Using kernels for non-linear spaces.

Module 5 (9 Hours)

Clustering: The k-means clustering algorithm, Using distance to assign and update clusters, Choosing number of clusters.
Evaluating model performance: Confusion matrices, Precision and recall, Sensitivity and specificity, Precision and recall, F-measure, ROC curves, Cross validation - K-fold cross validation, Bootstrap sampling.
Improving model performance - Bagging, Boosting, Random forests.

Text Books

1. Vijay Kotu, Bala Deshpande, Data Science Concepts and Practice, Morgan Kaufmann Publishers 2018 (Module 1)
2. Brett Lantz, Machine Learning with R, Second edition, PackT publishing 2015 (Modules 2 to 5)

Reference Books

1. Michael Steinbach, Pang-Ning Tan, and Vipin Kumar, Introduction to Data Mining, Pearson 2016.
2. Jiawei Han, Micheline Kamber and Jian Pei, Data mining Concepts and techniques, Morgan Kaufmann Publishers 2012
3. Peter Harrington, Machine Learning in action, Dreamtech publishers 2012
4. Dr M Gopal, Applied Machine learning, McGraw Hill Education Private Limited
5. E. Alpayidin, Introduction to Machine Learning, Prentice Hall of India (2005)
6. T. Hastie, RT Ibrashiran and J. Friedman, The Elements of Statistical Learning, Springer 2001
7. Data Science from Scratch: First Principles with Python, Joel Grus, O'Reilly, First edition, 2015
8. Introducing Data Science, Davy Cielen, Arno D. B. Meysman, Mohamed Ali, Manning Publications Co., 1st edition, 2016

Web Resources:

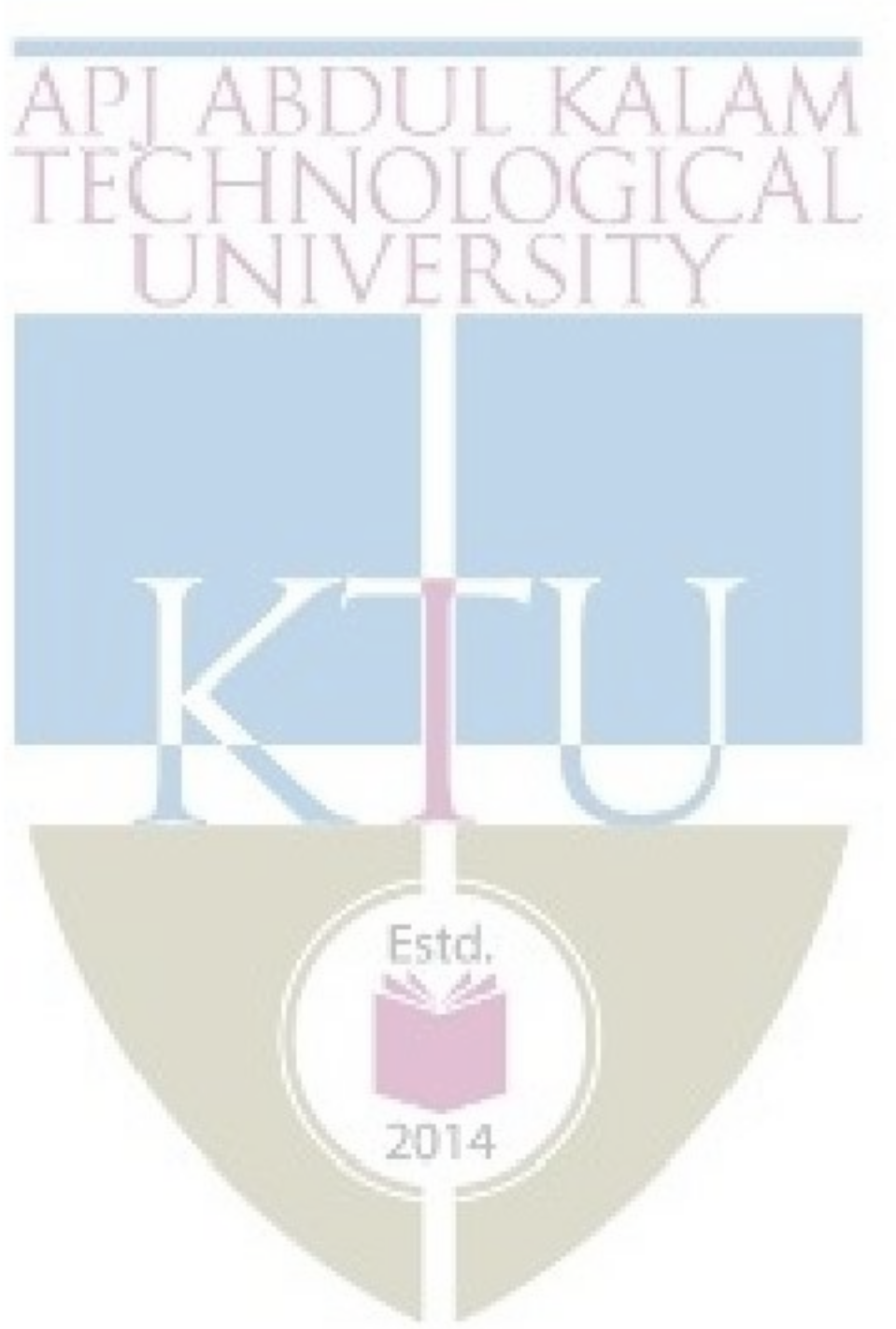
1. https://www.coursera.org/learn/machine-learning
2. https://www.coursera.org/learn/data-scientists-tools

Course Contents and Lecture Schedule

	Topic	No. of Lectures
1	**Module 1**	**9 hrs**
1.1	Introduction to data science - What is data science? Why data science?	2 hrs
1.2	Data science classification	1 hr
1.3	Data science process - Prior knowledge, Data preparation, Modelling, Application	2 hrs
1.4	Data exploration- Data sets, Descriptive statistics for univariate and multivariate data	2 hrs
1.5	Data visualization – Histogram, Quartile plot, Distribution chart, Scatter plot, Bubble chart, Density chart	2 hrs
2	**Module 2**	**9 hrs**

2.1	How machines learn – Data storage – Abstraction – Generalisation – Evaluation	1hr
2.2	Machine learning in practice – Types of machine learning algorithms.	1hr
2.3	Classification: Lazy learning - K-Nearest Neighbour algorithm	2 hrs
2.4	Measure of similarity, Choice of k	1 hr
2.5	Preparing data for use with k-NN	1 hr
2.6	Probabilistic Learning: Conditional probability and Bayes theorem.	1 hr
2.7	Naive Bayes algorithm	2 hrs
3	**Module 3**	**9 hrs**
3.1	Concept of decision tree, Divide and conquer approach	1 hr
3.2	C5.0 Decision tree algorithm	1 hr
3.3	Choosing the best split, Pruning the decision tree	2 hrs
3.4	Classification rules learning: Concept of classification rules, Separate and conquer approach	1 hr
3.5	The 1R algorithm, Rules from decision trees	1 hr
3.6	Regression methods: Concept of regression, Correlations	1 hr
3.7	Simple linear regression, Ordinary least squares estimation	1 hr
3.8	Multiple linear regression	1 hr
4	**Module 4**	**9 hrs**
4.1	Understanding neural networks - Artificial neurons	1 hr
4.2	Activation functions, Network topology	2 hrs
4.3	Training neural networks with back propagation	1hr
4.4	Understanding Support Vector Machines, Classification with hyperplane	1 hr
4.5	Linearly separable data, Nonlinearly separable data	1 hr
4.6	Methods to find maximum margin hyperplanes in linearly separable data	1 hr
4.7	Using kernels for non-linear spaces	2 hrs
5	**Module 5**	**9 hrs**
5.1	Understanding Clustering - The k-means clustering algorithm	1 hr

5.2	Using distance to assign and update clusters, Choosing the appropriate number of clusters	1 hr
5.3	Evaluating model performance: Confusion matrices, Precision and recall, Sensitivity and specificity, Precision and recall, F-measure, ROC curves.	2 hrs
5.4	Cross validation: K-fold cross validation, Bootstrap sampling	2 hrs
5.5	Improving model performance: Bagging, Boosting	2 hrs
5.6	Random forests	1 hr



CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
20MCA203	DESIGN & ANALYSIS OF ALGORITHMS	CORE	3	1	0	4

Preamble: The syllabus is prepared with a view to provide a strong foundation to students in design and analysis of computer algorithms and to introduce them the advanced topics such as Network Flows, Approximation algorithms and Randomised algorithms.

Prerequisite: Knowledge in Data Structures

Course Outcomes: After completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Discuss the basic concepts in computer algorithms and their analysis & design using Divide and Conquer.	Level 2: Understand
CO 2	Explain the concepts of Greedy Strategy and Dynamic Programming to use it in solving real world problems.	Level 3: Apply
CO 3	Explain the Branch & Bound technique, Backtracking technique and Lower bounds.	Level 2: Understand
CO 4	Describe the fundamental concepts of Computational Complexity and Network Flows.	Level 2: Understand
CO 5	Discuss the concepts of Approximation and Randomised Algorithms.	Level 2: Understand

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	3	1	2			2					
CO 2	3	3	1	2			2					
CO 3	3	3	1	2			2					
CO 4	3	3	1	2			2					
CO 5	3	3	1	2			2					

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category Levels	Continuous Assessment Tests		End Semester Examination
	1	2	
Level 1: Remember	20	20	20
Level 2: Understand	20	30	30
Level 3: Apply	10		10
Level 4: Analyse			
Level 5: Evaluate			
Level 6: Create			

Mark distribution

Total Marks	Continuous Internal Evaluation (CIE)	End Semester Examination (ESE)	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test (2 numbers) : 20 marks
Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be *two* parts; **Part A** and **Part B**. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer *all* questions. Part B contains 2 questions from each module of which student should answer *any one*. Each question can have a maximum 2 subdivisions and carry 6 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO 1):

1. Define "Time Complexity" of an algorithm?
2. What is the need for analysing an algorithm?
3. Define Big Oh Notation.
4. Define the terms Best Case, Worst Case and Average case complexities.
5. Explain the Merge Sort algorithm with an example.

Course Outcome 2 (CO 2):

1. Explain the Greedy Control abstraction.
2. Write the Prim's algorithm and illustrate with an example.
3. State and illustrate the Principle of Optimal Substructure.
4. Explain a solution to the Travelling Salesman problem using Dynamic Programming.

Course Outcome 3 (CO 3):

1. Explain the N-Queen's problem and its solution using Backtracking.
2. Explain the 8-puzzle problem and illustrate how it can be solved using Branch and Bound.
3. Bring out the notion of Decision Trees.
4. What is the lower bound of the time complexity of Comparison based sorting algorithms?

Course Outcome 4 (CO 4):

1. Define class P and NP.
2. What is Polynomial Time Reduction?
3. Show that the Clique problem is NP-Complete.
4. Define the Terms - Flow Network and Network Flow.
5. Explain the Ford-Fulkerson Algorithm.

Course Outcome 5 (CO 5):

1. What is an Approximation algorithm?
2. Describe the 2-approximation algorithm for Vertex Cover problem.
3. What is a Randomised algorithm?
4. Explain the Schwartz-Zippel Lemma. How this Lemma can be used to test the identity of two polynomials.

Model Question Paper
Course Code: 20MCA203

Course Name: Design and Analysis of Algorithms

Max. Marks :60 Duration: 3 Hrs

Part A

Answer all questions. Each question carries 3 marks (10 * 3 = 30 Marks)

1. Define Big Oh notation.
2. Write the control abstraction for a typical Divide and Conquer algorithm.
3. Explain a Greedy strategy which can give the optimal solution for the Knapsack problem.
4. Write a dynamic programming algorithm to compute the factorial of a number.
5. How does Backtracking differ from Branch and Bound?
6. Using a decision tree, show that any search algorithm which searches a given key within an array of n elements must perform at least O(ln n) comparisons in the worst case.
7. What do you mean by the term Polynomial time reduction?
8. Define the term Network Flow and illustrate with an example.

9. What do you mean by approximation ratio of an Approximation algorithm?

10. What is meant by a Randomised Algorithm?

Part B

Answer all questions. Each question carries 6 marks. (5 * 6 = 30 Marks)

11	Write the Linear Search Algorithm and analyse the best, worst and average case complexities of the algorithm.	6
	OR	
12	Explain the Merge Sort algorithm and give its worst-case analysis.	6
13	Write Kruskal's algorithm to compute the minimum cost spanning tree.	6
	OR	
14	Explain the dynamic programming algorithm for the Travelling Salesman problem.	6
15	Write the Backtracking algorithm for N-Queen Problem.	6
	OR	
16	Explain the 8-puzzle problem and its solution using branch and bound technique.	6
17	Show that the Clique problem is NP-Complete.	6
	OR	
18	Describe the Ford Fulkerson's procedure to compute the Max-Flow within a given Flow Network.	6
19	Explain the 2-approximation algorithm for Vertex Cover and justify its approximation ratio.	6
	OR	
20	Describe Randomised Quick sort.	6

Syllabus

Module 1: (8 Hours)
Review of Algorithm Analysis: Time and Space Complexity, Asymptotic Notations, Recurrence Equations, Solving Recurrence Equations- Substitution method and Iteration method. **Divide and Conquer**: Control Abstraction, Merge Sort, Quick Sort, Matrix Multiplication.
Module 2: (9 Hours)
Greedy Strategy: Control Abstraction, Knapsack Problem, Minimal Spanning Tree Algorithms- Prim's and Kruskal's Algorithm, Job Scheduling with deadlines **Dynamic Programming**: Control Abstraction, Principle of Optimal Substructure, All Pairs shortest path problem, Travelling Salesman Problem, Bellman-Ford Algorithm
Module 3:(7 Hours)
Backtracking: Control Abstraction, N-Queens problem, Sum of Subsets Problem **Branch and Bound**: Control Abstraction, 8- Puzzle problem **Lower Bounds:** The Decision Tree method, Lower Bounds for Comparison based Sort and Searching *(Analysis not required)*
Module 4: (11 Hours)
Complexity Theory: Class P and NP, Polynomial time reductions, Class NP Hard and NP-Complete, Example Problems- Vertex Cover problem, Clique Problem. **Network Flows**: Flow Networks and Network Flow, Max- Flow Min Cut Theorem, Ford Fulkerson method, Bipartite matching (*Analysis not required)*
Module 5: (10 Hours)
Introduction to Approximation Algorithms: Approximation Ratio, 2-approximation algorithm for Vertex Cover problem, Vertex Cover Approximation using Linear Programming and LP Rounding Algorithm. **Introduction to Randomised Algorithms**: Review of Basic Probability, Schwartz-Zippel Lemma and Polynomial Identity Testing, Randomized Quick Sort (*Proof of Expected Worst Case Analysis not required)*

Text Books

1. Thomas H. Cormen, et al., "Introduction to Algorithms", Prentice Hall, 3rd Edition (2010)
2. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms", Orient Longman, Universities Press, 2nd Edition (2008)

Reference Books

1. Richard Neapolitan, Kumarss Naimipour, "Foundations of Algorithms", Jones and Bartlett Publishers, Inc, 4th Edition (2011).
2. Sara Baase, Allen Van Gelder, "Computer Algorithms: Introduction to Design and Analysis", Pearson India, 3rd Edition (2002).
3. A. Levitin, "Introduction to the Design & Analysis of Algorithms", Pearson Education, 3rd Edition (2008).

Course Contents and Lecture Schedule

Sl. No.	Topic	No. of Lectures
1	**Review of Algorithm Analysis and Divide & Conquer**	**8 Hours**
1.1	Time and Space Complexity	1
1.2	Asymptotic Notations	1
1.3	Recurrence Equations, Solving Recurrence Equations- Substitution method	1
1.4	Iteration method	1
1.5	Divide and Conquer: Control Abstraction, Merge Sort, Merge Sort Analysis	2
1.6	Quick Sort, Quicksort analysis	1
1.7	Matrix Multiplication	1
2	**Greedy Strategy and Dynamic Programming**	**9 Hours**
2.1	Greedy Strategy: Control Abstraction, Knapsack Problem	1
2.2	Minimum Cost Spanning Tree	1
2.3	Prim's algorithm	1
2.4	Kruskal's algorithm	1
2.5	Job Scheduling with deadlines	1
2.6	Dynamic Programming: Control Abstraction, Principle of Optimal substructure	1
2.7	All Pairs shortest path problem	1
2.8	Travelling Salesman Problem	1
2.9	Bellman-Ford Algorithm	1

3	**Backtracking, Branch & Bound, Lower Bounds**	**7 Hours**
3.1	Backtracking: Control Abstraction N- Queens problem	1
3.2	Sum of subsets problem	1
3.3	Branch and Bound: Control Abstraction 8- Puzzle problem	1
3.4	Lower Bounds: The Decision Tree method	2
3.5	Lower Bounds for Comparison based Sorting	1
3.6	Lower bounds for searching	1
4	**Computational complexity, Network Flows**	**11 Hours**
4.1	Class P, NP	1
4.2	Polynomial Time Reductions	1
4.3	Class NP-Hard and NP-Complete	2
4.4	Vertex Cover Problem	1
4.5	Clique problem	1
4.6	Flow Networks and Network Flows	2
4.7	Max Flow Min Cut Theorem	1
4.8	Ford Fulkerson's method	1
4.9	Bipartite matching	1
5	**Approximation & Randomised Algorithms**	**10 Hours**
5.1	Approximation algorithms- introduction, Approximation Ratio	1
5.2	2- approximation algorithm for Vertex Cover problem	1
5.3	Vertex Cover Approximation using Linear Programming and LP Rounding Algorithm	2
5.4	Randomized Algorithms: introduction, Review of Basic Probability	1
5.5	Review of Basic probability	2
5.6	Schwartz-Zippel Lemma and Polynomial Identity Testing	2
5.7	Randomized Quick Sort	1

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
20MCA261	OPERATIONS RESEARCH	ELECTIVE	3	1	0	4

Preamble: This course introduces the concepts of linear programming problems. The topics treated in this course have applications in real life problems.

Prerequisite: Nil

Course Outcomes: After completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Solve different types of Linear Programming Problems.	Level 3: Apply
CO 2	Apply the concept of linear programming problems in real life.	Level 3: Apply
CO 3	Solve different decision-making problems using optimization techniques.	Level 3: Apply
CO 4	Use PERT and CPM to analyse project network management.	Level 3: Apply
CO 5	Identify suitable queuing model and solve queuing problems.	Level 3: Apply

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	3	1	-	-	-	2	-	-	-	-	-
CO 2	3	3	3	-	-	-	2	-	-	-	-	-
CO 3	3	3	3	-	-	-	2	-	-	-	-	-
CO 4	3	3	1	1	-	-	2	2	-	-	-	-
CO 5	3	3	3	-	-	-	2	-	-	-	-	-

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category Levels	Continuous Assessment Tests		End Semester Examination
	1	2	
Level 1: Remember	10	10	10
Level 2: Understand	20	20	20
Level 3: Apply	20	20	30
Level 4: Analyse			
Level 5: Evaluate			
Level 6: Create			

Mark Distribution

Total Marks	Continuous Internal Evaluation (CIE)	End Semester Examination (ESE)	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test (2 numbers) : 20 marks
Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks

Sample Course Level Assessment Questions

Course Outcome 1 (CO 1):

1. Define slack variable, surplus variable and optimal basic feasible solution.
2. Obtain all basic feasible solution of the set of equations:
   a) $2x_1 + 3x_2 + 4x_3 + x_4 = 2$
   b) $x_1 + x_2 + 7x_3 + x_4 = 4$
3. Solve by Big M method

Maximise $Z = 6x_1 - 3x_2 + 2x_3$
Subject to $2x_1 + x_2 + x_3 \leq 16$
$3x_1 + 2x_2 + x_3 \leq 18$

$$x_1 - 2x_2 \geq 8$$

$$x_1,\ x_2,\ x_3 \geq 0$$

Course Outcome 2 (CO 2):

1. Construct the dual of

$$\begin{aligned} &\text{Maximise} && Z = 3x_1 + 17x_2 + 9x_3 \\ &\text{Subject to} && x_1 - x_2 + x_3 \geq 3 \\ & && -3x_1 + 2x_2 \leq 1 \\ & && x_1,\ x_2,\ x_3 \geq 0 \end{aligned}$$

2. Prove that the dual of the dual is the primal
3. Solve using the principle of duality

$$\begin{aligned} &\text{Minimise} && Z = 3x_1 + 5x_2 \\ &\text{Subject to} && 2x_1 + 8x_2 \geq 40 \\ & && 3x_1 + 4x_2 \geq 50 \\ & && x_1,\ x_2 \geq 0 \end{aligned}$$

Course Outcome 3 (CO 3):

1. Explain North West Corner method
2. Solve the following transportation problem

	1	2	3	Supply
1	2	7	4	5
2	3	3	1	8
3	5	4	7	7
4	1	6	2	14
Demand	7	9	18	34

3. Solve the assignment problem

	I	II	III	IV
A	16	10	14	11
B	14	11	15	15
C	15	15	13	12
D	13	12	14	15

Course Outcome 4 (CO 4):

1. Explain critical path analysis.
2. A project consists of series of tasks labelled A, B, …, H, I with the following relationships (W < X, Y means X and Y cannot start until W is completed; X, Y < W means W cannot start until both X and Y are completed). With this notation construct the network diagram having the following constraints:

   A < D, E; B, D < F; C < G; B, G< H; F, G < I.

   Find also the minimum time of completion of the project, when the time (in days) of completion of each task is as follows:

Task :	A	B	C	D	E	F	G	H	I
Time :	23	8	20	16	24	18	19	4	10

3. A project consists of eight activities with the following relevant information.

Activity	Immediate predecessor	Estimated duration (days)		
		Optimistic	Most likely	Pessimistic
A	--	1	1	7
B	--	1	4	7
C	--	2	2	8
D	A	1	1	1
E	B	2	5	14
F	C	2	5	8
G	D, E	3	6	15
H	F, G	1	2	3

(i) Draw the PERT network and find out the expected project completion time.
(ii) What duration will have 95% confidence for project completion?
(iii) If the average duration for activity F increases to 14 days, what will be its effects on the expected project completion time which will have 95% confidence?

(For standard normal Z = 1.645, area under the standard normal curve from 0 to Z is 0.45)

Course Outcome 5 (CO 5):

1. Explain Birth-death process.
2. In a railway marshalling yard, goods trains arrive at a rate of 30 trains per day. Assuming that the inter-arrival time follows an exponential distribution and the

service time distribution is also exponential with an average 36 minutes. Calculate the following:

i. The mean queue size (line length), and
ii. The probability that the queue size exceeds 10.
iii. If the input of trains increases to an average 33 per day, what will be the change in (i) and (ii)?

3. At a railway station, only one train is handled at a time. The railway yard is sufficient only for two trains to wait while other is given signal to leave the station. Trains arrive at the station at an average rate of 6 per hour and the railway station can handle them on an average of 12 per hour. Assuming Poisson arrivals and exponential service distribution, find the steady-state probabilities for the various number of trains in the system. also find the average waiting time of a new train coming into the yard

Model Question Paper
Course Code: 20MCA261
Course name: Operations Research

Max. Marks: 60 Duration: 3hrs

Part A

Answer all questions, each carries 3 marks (10×3 = 30)

1. Write down the basic structure of a linear programming problem in the mathematical form.
2. Define slack and surplus variables in LPP.
3. State the fundamental theorem of duality.
4. Write the dual of the following

$$Max\ Z = x_1 - x_2 + 3x_3$$
$$subject\ to\ x_1 + x_2 + x_3 \leq 10$$
$$2x_1 - x_3 \leq 2$$
$$2x_1 - 2x_2 + 3x_3 \leq 6$$
$$x_1, x_2, x_3 \geq 0$$

5. Obtain the IBFS using north west corner method

	D_1	D_2	D_3	D_4	Supply
O_1	2	4	3	6	20
O_2	7	3	8	2	10
O_3	2	2	9	11	15
Demand	15	15	8	7	

6. Describe the Matrix Minima method.
7. What is queue discipline?
8. Explain single serve Poisson queuing model with infinite capacity.
9. Activities P, Q and R instantly follow activity M, and their current starting times are 12, 19 and 10. So, what is the latest finishing time for activity M?
10. What is the difference between PERT and CPM.

Part B

Answer all questions, each carries 6 marks (5×6 = 30)

11. Solve the following problem by Simplex method

$$Max\ Z = 5x_1 + 3x_2$$
$$subject\ to\ 4x_1 - x_2 \leq 10$$
$$2x_1 + 2x_2 \leq 50$$
$$x_1, x_2 \geq 0$$

or

12. Solve by Big-M method

$$Max\ Z = 6x_1 - 3x_2 + 2x_3$$
$$subject\ to\ 2x_1 + x_2 + x_3 \leq 16$$
$$3x_1 + 2x_2 + x_3 \leq 18$$
$$x_2 - 2x_3 \geq 8$$
$$x_1, x_2, x_3 \geq 0$$

13. Prove that the dual of a dual is the primal.

or

14. Solve the following by using the dual principle

$$Max\ Z = 40x_1 + 35x_2$$
$$subject\ to\ 2x_1 + 3x_2 \leq 60$$
$$4x_1 + 3x_2 \leq 96$$
$$x_1, x_2 \geq 0$$

15. Solve the following Assignment problem

	I	II	III	IV
A	2	3	4	5
B	4	5	6	7
C	7	8	9	8
D	3	5	8	9

or

16. Solve the following transportation problem

	D_1	D_2	D_3	D_4	Supply
O_1	5	2	4	3	22
O_2	4	5	1	6	15
O_3	4	6	7	5	8
Demand	7	12	17	9	

17. Explain critical path analysis.

or

18. A project consists of eight activities with the following relevant information.

Activity	Immediate predecessor	Estimated duration (days)		
		Optimistic	Most likely	Pessimistic
A	--	1	1	7
B	--	1	4	7
C	--	2	2	8
D	A	1	1	1
E	B	2	5	14
F	C	2	5	8
G	D, E	3	6	15
H	F, G	1	2	3

(iv) Draw the PERT network and find out the expected project completion time.
(v) What duration will have 95% confidence for project completion?
(vi) If the average duration for activity F increases to 14 days, what will be its effects on the expected project completion time which will have 95% confidence?

(For standard normal Z = 1.645, area under the standard normal curve from 0 to Z is 0.45)

19. Explain birth-death process.

or

20. At a railway station, only one train is handled at a time. The railway yard is sufficient only for two trains to wait while other is given signal to leave the station. Trains arrive at the station at an average rate of 6 per hour and the railway station can handle them on an average of 12 per hour. Assuming Poisson arrivals and exponential service distribution, find the steady-state probabilities for the various number of trains in the system. also find the average waiting time of a new train coming into the yard.

Syllabus

Module 1: (9 Hours)
Linear programming problem- Slack and surplus variable- Standard form- Solution of Linear programming problem- Basic solution- Basic feasible solution- Degenerate- and Non-degenerate solutions- Optimal solution- Solution by simplex method- Artificial variables- Big- M method.
Module 2: (9 Hours)
Duality in Linear Programming Problem- Statement of duality theorem- Statement of complementary slackness theorem. The primal- Duality solutions using simplex method- Revised simplex method
Module 3: (9 Hours)
Transportation problem- Solution of Transportation problem- Finding an initial basic feasible solution- North West Corner method- Matrix minima method- Vogel's Approximation method- Test for Optimality- Modi method- Unbalanced Transportation problem- Maximisation in Transportation problem. Assignment problem- Optimal solution- Hungarian method of assignment- Maximization in assignment problem.
Module 4: (9 Hours)
Network analysis- Project scheduling- Construction of project networks- Critical path method (CPM)- Identification of critical path using CPM- Estimation of Floats- Total float- Independent float- Project Evaluation and Review Technique (PERT) - Computation of expected completion times by PERT.
Module 5: (9 Hours)
Queuing theory- Elements of Queuing System- Kendall's notation- Operating characteristics- Poisson process- Exponential distribution- Mean and variance- Birth and Death process. Queuing models based on Poisson process- Single server models with finite and infinite capacity- Multi server model with finite and infinite capacity.

Note:

- Programming Assignments using Python and appropriate Case Studies may be given at the end of each module.

- Linear Programming Problems in module 1 and module 2 and Transportation problems in module 3 can be solved using Python library PuLP. Using Numpy, PERT/CPM problems in module 4 can be solved.

Text Book

1. KantiSwarup, P.K. Gupta and Man Mohan, Operation Research, Sultan Chand (2010)

Reference Books

1. Hamdy A Taha, Operations Research- an introduction, Eighth Edition, Prentice Hall of India.
2. Ravindran, Philips and Solberg, Wiley, Operation Research, Second edition (2007)

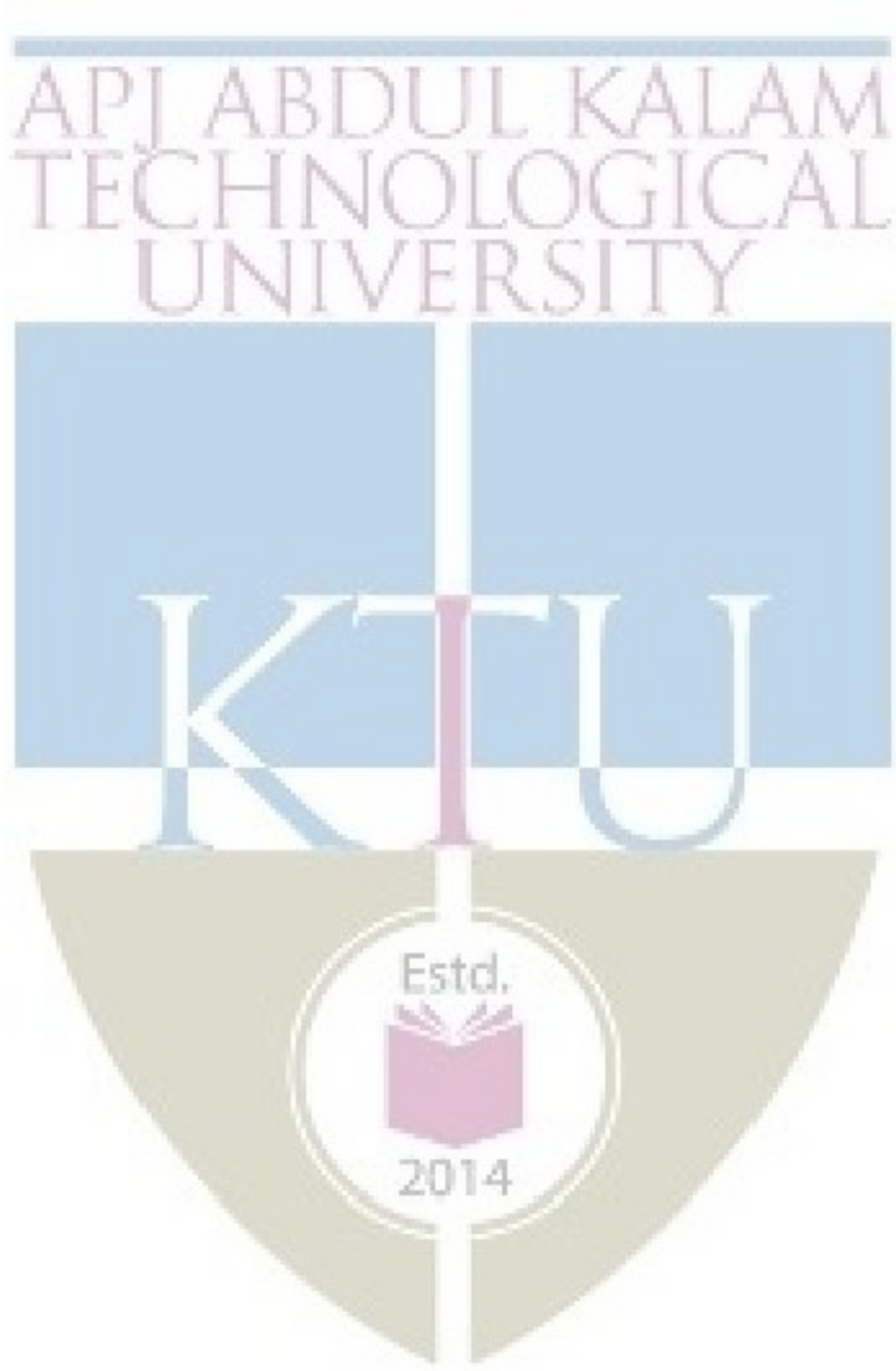
Web References

1. https://pypi.org/project/PuLP/
2. https://numpy.org/

Course Contents and Lecture Schedule

Sl. No.	Topic	No. of Lectures
1	**Module 1**	**9 Hours**
1.1	Linear programming problem- Slack and surplus variable- Standard form	1
1.2	Solution of Linear programming problem- Basic solution- Basic feasible solution- Degenerate- and Non-degenerate solutions- Optimal solution	2
1.3	Solution by simplex method	3
1.4	Artificial variables- Big- M method	3
2	**Module 2**	**9 Hours**
2.1	Duality in Linear Programming Problem	1
2.2	Statement of duality theorem- Statement of complementary slackness theorem	2
2.3	The primal- Duality solutions using simplex method	3
2.4	Revised simplex method	3
3	**Module 3**	**9 Hours**
3.1	Transportation problem- Solution of Transportation problem- Finding an initial basic feasible solution- North West Corner method	2
3.2	Matrix minima method- Vogel's Approximation method	1
3.3	Test for Optimality- Modi method- Unbalanced Transportation problem- Maximisation in Transportation problem	3
3.4	Assignment problem- Optimal solution- Hungarian method of assignment- Maximization in assignment problem	3
4	**Module 4**	**9 Hours**
4.1	Network analysis- Project scheduling- Construction of project networks	1
4.2	Critical path method (CPM)- Identification of critical path using CPM	2
4.3	Estimation of Floats- Total float- Independent float	3
4.4	Project Evaluation and Review Technique (PERT)	2
4.5	Computation of expected completion times by PERT	1

5	**Module 5**	**9 Hours**
5.1	Queuing theory- Elements of Queuing System- Kendall's notation- Operating characteristics- Poisson process	1
5.2	Exponential distribution- Mean and variance- Birth and Death process	2
5.3	Queuing models based on Poisson process	3
5.4	Single server models with finite and infinite capacity	1
5.5	Multi server model with finite and infinite capacity	2



CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
20MCA263	CYBER SECURITY & CRYPTOGRAPHY	ELECTIVE	3	1	0	4

Preamble: This course is designed to provide theoretical concepts used in cryptography and to introduce the students to various cryptographic algorithms and techniques used for implementing data security and protection. This course also discusses common web application security vulnerabilities.

Prerequisite: Student is expected to have studied mathematics courses that cover Elementary Number Theory, Finite Field, Discrete Logarithm and Euclidean Algorithm.

Course Outcomes: After completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Explain various types of security attacks, security mechanisms, security services and classical encryption techniques.	Level 2: Understand
CO 2	Make use of Symmetric and Asymmetric encryption techniques to solve cryptographic problems.	Level 3: Apply
CO 3	Describe the concepts of message authentication codes, hash functions and digital signing techniques for ensuring secure transactions.	Level 2: Understand
CO 4	Discuss security services in Application, Transport and Network layers.	Level 2: Understand
CO 5	Explain common web application security vulnerabilities and various prevention mechanisms.	Level 2: Understand

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	2	1	1				1					
CO 2	2	2	2	1			1					
CO 3	2	1	1				1					
CO 4	2	1	1			2	1					
CO 5	2	1	1			2	1					

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category Levels	Continuous Assessment Tests		End Semester Examination
	1	2	
Level 1: Remember	15	15	20
Level 2: Understand	35	35	40
Level 3: Apply			
Level 4: Analyse			
Level 5: Evaluate			
Level 6: Create			

Mark distribution

Total Marks	Continuous Internal Evaluation (CIE)	End Semester Examination (ESE)	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test (2 numbers) : 20 marks
Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have a maximum of 2 sub-divisions and carry 6 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO 1):

1. Briefly explain each component of OSI security architecture.
2. Compare Substitution and Transposition techniques in cryptography.
3. Explain how steganography is used in cryptography.

Course Outcome 2 (CO 2):

1. Explain block cipher modes of operation.
2. Compare DES and AES

3. Perform encryption and decryption using RSA Algorithm with parameters: P=17, q = 11, e = 7, M = 88.

Course Outcome 3 (CO 3):

1. Compare the features of HMAC and CMAC algorithms.
2. Explain important steps in DSS.
3. Describe the terms (a) birthday attack (b) hashcash (c) blind signature

Course Outcome 4 (CO 4):

1. Explain any one protocol used in E-mail for security.
2. Explain how security is provided in Network Layer using IPsec.
3. Describe the process of securing electronic transactions.

Course Outcome 5 (CO 5):

1. Discuss any four Application Security Risks.
2. Which are the different forms of XSS and how to prevent these?
3. Explain the attack scenario of any four web application security vulnerabilities.

Model Question Paper

Course Code: 20MCA263

Course Name: CYBER SECURITY & CRYPTOGRAPHY

Max. Marks :60 **Duration: 3 Hrs**

Part A

Answer all questions.

Each question carries 3 marks (10 x 3 = 30 Marks)

.

1. Compare phishing and ransomware attacks.
2. What is OSI security architecture?
3. List out the advantages and disadvantages of Output Feed Back mode.
4. Explain round functions used in DES.
5. Explain important steps in DSS.
6. Describe the terms (a) birthday attack (b) hashcash (c) blind signature.
7. Describe security association of IPSec.
8. Explain about S/MIME.

9. How can we prevent Injection attack?
10. What is XXE? How to prevent it?

(10 x 3=30 marks)

Part B

Answer all questions. Each question carries 6 marks. (5 * 6 = 30 Marks)

11. Explain Network security model with the help of a neat diagram (6)

OR

12. Describe the working of Playfair cipher and Hill cipher. (6)

13. Apply Diffie-Hellman key exchange algorithm to compute the shared private key using the values P = 23, g = 9, a= 4, b= 3. Explain the steps in detail. (6)

OR

14. Perform encryption and decryption using RSA Algorithm with parameters: P=17, q = 11, e = 7, M = 88. Explain the steps in detail. (6)

15. Compare HMAC and CMAC protocol with suitable diagrams. (6)

OR

16. Compare various signature schemes with suitable diagrams. (6)

17. Explain PGP cryptographic functions with diagram. (6)

OR

18. Explain Secure Electronic Transaction Protocol. (6)

19. Briefly explain any four Application Security Risks. (6)

OR

20. Explain the attack scenarios of any four web application security vulnerabilities. (6)

(5 x 6=30 Marks)

Syllabus

Module 1: (7 Hours)
Introduction to Cryptography, OSI security architecture: Security Services, Mechanisms and attacks- Phishing, Ransomware, DoS attack. Network security model. Classical Encryption techniques - Symmetric cipher model, substitution techniques, transposition techniques. Steganography.
Module 2: (10 Hours)
Conventional Symmetric Key Encryption: Block ciphers and Stream Ciphers, Block Cipher Design Principles, Modes of operation, Data Encryption Standard, Advanced Encryption Standard (AES), Multiple Encryption, Triple DES. **Public key cryptography:** Principles of public key cryptosystems-The RSA algorithm-Key management – Diffie Hellman Key exchange - Elliptic curve arithmetic - Elliptic curve cryptography.
Module 3: (10 Hours)
Hash Functions and MAC: Properties of hash functions, birthday attack, hashcash, Message Authentication Code Algorithms, MAC protocols: HMAC, CMAC. **Digital Signatures:** Classification of signature schemes: RSA signature, Digital Signature Standard, Overview of ElGamal and Schnorr schemes, One time signature schemes, Attacks on Digital Signatures, Blind Signatures.
Module 4: (10 Hours)
Introduction to Cyber Security: Email Security: Security Services for email, Attacks possible through email, Establishing keys privacy, authentication of the source, Message Integrity, Non-repudiation, Pretty Good Privacy, S/MIME. **IP Security:** Overview of IPSec, IPv4 and IPv6, Authentication Header, Encapsulation Security Payload (ESP), Internet Key Exchange. **Transport Level Security:** SSL/TLS Basic Protocol, computing the keys, client authentication, PKI as deployed by SSL, Attacks fixed in v3, Exportability, Encoding, Secure Electronic Transaction (SET).
Module 5: (8 Hours)
Common web application security vulnerabilities: Injection flaws, Broken authentication, Sensitive data exposure, XML External Entities (XXE), Broken access control, Security misconfiguration, Cross-Site Scripting (XSS), Insecure deserialization, Using components with known vulnerabilities, Insufficient logging & monitoring. Example attack scenarios of each of the vulnerabilities listed; how to prevent them

Text Book

1. William Stallings, "Cryptography and Network Security," 6th Edition, Pearson Education, March (2013).
2. Behrouz A. Forouzan, "Introduction to Cryptography and Network Security", Tata McGraw-Hill Publishing 2nd Edition (2011).

Reference Books

1. Charlie Kaufman, Radia Perlman and Mike Speciner, "Network Security", Prentice Hall of India, 2002.
2. Manuel Mogollon, "Cryptography and Security Services – Mechanisms and Applications", Cybertech Publishing, 2008
3. William R. Cheswick, Steven M. Bellovin, Aviel D. Rubin, "Firewalls and Internet Security" Addison- Wesley, 2003

Web References

1. http://www.hashcash.org/hashcash.pdf [Reference for hashcash]
2. https://owasp.org/www-pdf-archive/OWASP_Top_10-2017_%28en%29.pdf. [Reference for Module 5]
3. https://www.coursera.org/learn/crypto
4. https://www.coursera.org/learn/crypto2

Course Contents and Lecture Schedule

Sl. No.	Topic	No. of Lectures
1	**Introduction to Cryptography**	**7 Hours**
1.1	What is cryptography, Related Terms, Need of cryptosystems	1
1.2	OSI security architecture: Security Services, Mechanisms	1
1.3	Security attacks- Phishing, Ransomware, DoS attack.	1
1.4	Network security model	1
1.5	Classical Encryption techniques, Symmetric cipher model	1
1.6	Substitution techniques	1
1.7	Transposition techniques, Steganography	1
2	**Conventional Symmetric and Public Key Encryption**	**10 Hours**
2.1	Block ciphers and Stream Ciphers, Block Cipher Design Principles	1
2.2	Modes of operation	1
2.3	Data Encryption Standard	1
2.4	Advanced Encryption Standard (AES)	1
2.5	Multiple Encryption, Triple DES	1
2.6	Public key cryptography: Principles of public key cryptosystems	1
2.7	The RSA algorithm	1

2.8	Key management	1
2.9	Diffie Hellman Key exchange	1
2.10	Elliptic curve arithmetic - Elliptic curve cryptography.	1
3	**Hash Functions and MAC**	**10 Hours**
3.1	Properties of hash functions, birthday attack	1
3.2	Hashcash, Message Authentication Code Algorithms	1
3.3	MAC protocols: HMAC, CMAC	1
3.4	Digital Signatures: Classification of signature schemes	1
3.5	RSA signature	1
3.6	Digital Signature Standard	1
3.7	Overview of ElGamal and Schnorr schemes	1
3.8	One time signature schemes	1
3.9	Attacks on Digital Signatures	1
3.10	Blind Signatures	1
4	**Introduction to Cyber Security**	**10 Hours**
4.1	Email Security: Security Services for email, Attacks possible through email	1
4.2	Establishing keys privacy, authentication of the source, Message Integrity, Non-repudiation	1
4.3	Pretty Good Privacy, S/MIME	1
4.4	IP Security: Overview of IPSec	1
4.5	IPv4 and IPv6, Authentication Header	1
4.6	Encapsulation Security Payload (ESP), Internet Key Exchange	1
4.7	Transport Level Security: SSL/TLS Basic Protocol	1
4.8	computing the keys, client authentication, PKI as deployed by SSL	1
4.9	Attacks fixed in v3, Exportability, Encoding	1
4.10	Secure Electronic Transaction (SET)	1
5	**Common web application security vulnerabilities**	**8 Hours**
5.1	Common web application security vulnerabilities	1
5.2	Injection flaws, Broken authentication	1
5.3	Sensitive data exposure, XML External Entities (XXE)	1
5.4	Broken access control, Security misconfiguration	1
5.5	Cross-Site Scripting (XSS), Insecure deserialization	1
5.6	Using components with known vulnerabilities, Insufficient logging & monitoring.	1
5.7	Example attack scenarios of each of the vulnerabilities listed	1
5.8	How to prevent each of the vulnerabilities.	1

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
20MCA265	Cloud Computing	ELECTIVE	3	1	0	4

Preamble: The syllabus is prepared with a view to equip the students to learn basic concepts in cloud computing - compute, storage, networking. They should gain basic understanding of orchestration, HA and failover.

Prerequisite: Awareness in Virtualisation and Containers is desirable.

Course Outcomes: After completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Understand the basic concepts in cloud computing and OpenStack logical architecture	Level 2: Understand
CO 2	Discuss OpenStack cloud controller and common services	Level 3: Apply
CO 3	Compare different OpenStack compute service components and storage types	Level 2: Understand
CO 4	Describe the OpenStack Networking- Connection types and networking services	Level 2: Understand
CO 5	Discuss orchestration, HA and failover in OpenStack	Level 2: Understand

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	2				3		2					1
CO 2	2				3		2					1
CO 3	2				3		2					1
CO 4	2				3		2					1
CO 5	2				3		2					1

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category Levels	Continuous Assessment Tests		End Semester Examination
	1	2	
Level 1: Remember	20	15	20
Level 2: Understand	20	35	30
Level 3: Apply	10		10
Level 4: Analyse			
Level 5: Evaluate			
Level 6: Create			

Mark distribution

Total Marks	Continuous Internal Evaluation (CIE)	End Semester Examination (ESE)	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test (2 numbers) : 20 marks
Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have a maximum 2 subdivisions and carry 6 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO 1):

1. List and explain various components of Nova compute service.
2. Explain the neutron architecture?
3. Briefly describe keystone identity management.

Course Outcome 2 (CO 2):

1. Explain the telemetry services in OpenStack.
2. Explain the steps involved in bringing up a working OpenStack Ansible on the deployment host.
3. Explain the steps in network configuration

Course Outcome 3 (CO 3):

1. Explain briefly swift architecture
2. Briefly explain how data is handled in the cluster by swift
3. What is meant by CPU over commitment?

Course Outcome 4 (CO 4):

1. Explain steps in associating a floating IP to a virtual machine.
2. Briefly explain the steps in creating a virtual network with two subnets
3. Briefly explain Linux bridge-based connectivity?

Course Outcome 5 (CO 5):

1. Briefly explain the major components in heat?
2. Explain the different metrics that can be measured in a highly available infrastructure?
3. Explain the need for Service level agreement.

Model Question Paper
Course Code: 20MCA265
Course Name: Cloud Computing

Max. Marks :60 **Duration: 3 Hrs**

Part A

Answer all questions. Each question carries 3 marks (10 * 3 = 30 Marks)

1. What are the different components in OpenStack logical architecture?
2. Differentiate between private cloud and public cloud.
3. Explain asymmetric clustering and symmetric clustering.
4. List out the functionalities handled by the cloud controller.
5. Briefly explain docker containers.
6. Compare object storage with NAS/SAN based storage.
7. Describe the steps in connecting two networks using a virtual router.
8. Write a short note on firewall as a service
9. List the HA levels in OpenStack.
10. Explain the purpose of HA proxy.

Part B

Answer all questions. Each question carries 6 marks. (5 * 6 = 30 Marks)

11	List and explain the different components in OpenStack Architecture.	6
	OR	
12	a. Explain the provisioning of VM in OpenStack using a diagram	4
	b. Describe the best practices used in Physical mode design	2
13	Explain the keystone architecture	6
	OR	
14	Explain the steps involved in running OpenStack playbooks	6
15	Explain in detail the multiple services involved in launching an instance	6
	OR	
16	Explain the steps in deploying swift service	6
17	Explain the architecture of neutron in detail	6
	OR	
18	Explain the categorization of neutron virtual networks in detail	6
19	Explain stacking in OpenStack	6
	OR	
20	Explain in detail steps involved in setting a database with high availability	6

Syllabus

Module 1: Overview of OpenStack (7 Hours)
Introduction to cloud computing, private cloud, public cloud, hybrid cloud architecture. Cloud Services - Infrastructure as a Service, Platform as a Service, Storage as a Service. Designing OpenStack Cloud Architectural Consideration - OpenStack - The new data centre paradigm - OpenStack logical architecture - Nova - Compute Service-Neutron - Networking services - Gathering the pieces and building a picture - A sample architecture setup.
Module 2: OpenStack cluster - Controller and common services (6 Hours)
OpenStack Cluster – The Cloud Controller and Common Services- Asymmetric clustering, Symmetric clustering, The cloud controller - The keystone service. The nova-conductor service, The nova-scheduler service, The API services, Image management, The network service, The horizon dashboard, The telemetry services.

Module 3: OpenStack compute and Storage (12 Hours)
OpenStack Compute -The compute service components - Deciding on the hypervisor - OpenStack Magnum Project - Segregating the compute cloud - Overcommitment considerations - Storing instances' alternatives - Understanding instance booting - Planning for service recovery. OpenStack Storage - Block, Object, and File Share - Understanding the storage types - Ephemeral Storage - Persistent storage - A spotlight on Swift - Deploying Swift service - Using block storage service: Cinder.
Module 4: OpenStack Networking (10 Hours)
The architecture of Neutron - Implementing virtual networks - Connecting virtual networks with routers - Implementing network security in OpenStack. OpenStack Networking - The architecture of Neutron - Implementing virtual networks - VLAN, Tunnel based, Virtual Switches, The ML2 Plugin. Neutron Subnets - Connecting virtual networks with routers - Configuring the routing service - connecting networks using a virtual router, connecting to the external world, connectivity from the external world, associating a floating IP - Implementing network security in OpenStack
Module 5: OpenStack Orchestration, HA and failover (10 Hours)
Orchestration in OpenStack - Heat and its Components, stacking in OpenStack, OpenStack Orchestration with Terraform. OpenStack HA and failover: Scope of HA in OpenStack, HA in the database, HA in the Queue, Implementing HA on RabbitMQ.

Text Book

1. Omar Khedher, Chandan Datta Chowdhury, Mastering OpenStack, 2nd Edition, Packt Publishing, 2017

Reference Books

1. Tom Fifield, Diane Fleming, Anne Gentle,Lorin Hochstein, Jonathan Proulx, Everett Toews, and Joe Topjian, OpenStack Operations Guide, O'REILY,1/e, 2014.
2. Uchit Vyas, Applied OpenStack Design Patterns, Apress, 1/e, 2016.
3. V. K. Cody Bumgardner, OpenStack in action, Manning, 2016.
4. Amar Kapadia, Sreedhar Varma, Kris Rajana, Implementing Cloud Storage with OpenStack Swift, Packt Publishing, 2014.
5. https://docs.openstack.org/wallaby/?_ga=2.231002015.1428061357.1620834394-1139122985.1620834394

Course Contents and Lecture Schedule

Sl. No.	Topic	No. of Lectures
1	**Overview of OpenStack**	**7 Hours**
1.1	Introduction to cloud computing, private cloud, public cloud, hybrid cloud architecture.	1
1.2	Cloud Services - Infrastructure as a Service, Platform as a Service, Storage as a Service	1
1.3	Designing OpenStack Cloud Architectural Consideration - OpenStack - The new data center paradigm -OpenStack logical architecture	1
1.4	Nova - Compute service	1
1.5	Neutron - Networking services	1
1.6	Gathering the pieces and building a picture	1
1.7	A sample architecture setup	1
2	**OpenStack cluster - Controller and common services**	**6 Hours**
2.1	OpenStack Cluster – The Cloud Controller and Common Services-Asymmetric clustering, Symmetric clustering	1
2.2	The cloud controller - The keystone service	2
2.3	The nova-conductor service, The nova-scheduler service, The API services, Image management.	1
2.4	The network service, The horizon dashboard, The telemetry services	2
3	**OpenStack compute and Storage**	**12 Hours**
3.1	The compute service components-Deciding on the hypervisor-OpenStack Magnum project	1
3.2	Segregating the compute cloud	1
3.3	Overcommitment considerations	1
3.4	Storing instances' alternatives	1
3.5	Understanding instance booting	1
3.6	Planning for service recovery	1
3.7	OpenStack Storage - Block, Object, and File Share-Understanding the storage types	1
3.8	A spotlight on swift	2
3.9	Deploying swift service	1

3.10	Using Block Storage Service Cinder	2
4	**OpenStack Networking**	**10 Hours**
4.1	The architecture of Neutron	1
4.2	Implementing virtual networks - VLAN, Tunnel based	1
4.3	Virtual Switches, The ML2 Plugin	1
4.4	Neutron Subnets	2
4.5	Connecting virtual networks with routers - Configuring the routing service	1
4.6	Connecting networks using a virtual router, Connecting to the external world	1
4.7	Connectivity from the external world, Associating a floating IP	1
4.8	Implementing network security in OpenStack	2
5	**OpenStack Orchestration, HA and Failover**	**10 Hours**
5.1	Orchestration in OpenStack, Heat and its Components	1
5.2	Stacking in OpenStack	2
5.3	OpenStack Orchestration with Terraform	2
5.4	Scope of HA in OpenStack	2
5.5	HA in the database	1
5.6	HA in the Queue, Implementing HA on RabbitMQ	2

Suggested Assignments

1) Create VMs in your physical machine using OpenStack to set up the following services: Moodle, MySQL Server, Samba. Design the desired configuration of the physical machine to handle the requirements of the entire college.
2) Set up storage services for storing external files for Moodle.
3) Set up firewall rules for samba, MySQL server, allow the connection to MySQL server only to Moodle VM.
4) Set up recovery plans for the above services
5) Convert the MySQL server to HA MySQL server.
6) Setup a load balancer for the Moodle server.

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
20MCA267	CYBER FORENSICS	ELECTIVE	3	1	0	4

Preamble: This course helps the learner to understand the fundamentals of cyber forensics. Student will learn common approaches, practices and techniques used for collecting and preserving digital evidences in this course.

Prerequisite: Basic knowledge in operating systems & computer networks.

Course Outcomes: After completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Explain a computer crime and the concept of rules or policy violations.	Level 2: Understand
CO 2	Gather evidences and preserve the collected evidence with the required knowledge on various storage format choices.	Level 3: Apply
CO 3	Describe digital storage and file systems and extract data using Autopsy.	Level 3: Apply
CO 4	Explain mobile device forensics and practice data acquisition procedures for network forensics using Wireshark.	Level 3: Apply
CO 5	Prepare forensics reports both using tools and manually and explain ethics and code for expert witness.	Level 2: Understand

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	2	1		1		2	1		2			
CO 2	2	1		1	2		1					
CO 3	2	1		1	2		1					
CO 4	2	1		1	2		1					
CO 5	2	1		1	2	3	1		1			

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category Levels	Continuous Assessment Tests		End Semester Examination
	1	2	
Level 1: Remember	15	15	20
Level 2: Understand	35	35	40
Level 3: Apply			
Level 4: Analyse			
Level 5: Evaluate			
Level 6: Create			

Mark distribution

Total Marks	Continuous Internal Evaluation (CIE)	End Semester Examination (ESE)	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test (2 numbers) : 20 marks
Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have a maximum of 2 sub-divisions and carry 6 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO 1):

1. How to perform industrial espionage investigation?
2. Explain the various steps involved in cyber forensics investigation.
3. Identify real-time scenarios that is considered as company policy violation.

Course Outcome 2 (CO 2):

1. What are the advantages of proprietary evidence format over row format?
2. Differentiate static acquisitions from live acquisitions.
3. How to ensure the integrity of collected digital evidence? List the techniques employed to validate the collected evidence.

Course Outcome 3 (CO 3):

1. Describe the various activities involved while starting a windows operating system.
2. Explain the various file system organization used in windows operating system.
3. What is the importance of windows registry analysis in forensic investigation?
4. How to ensure the integrity of collected evidences?
5. Differentiate the forensics procedure in Linux and MacOS.

Course Outcome 4 (CO 4):

1. Explain the mobile forensics procedure in detail.
2. Illustrate the use of Wireshark packet analyser.
3. How the forensics acquisition method in mobile differs from that in computer system?

Course Outcome 5 (CO 5):

1. How to write an investigation report that can sustain in court of law?
2. Discuss how Autopsy tool is used to generate forensics report.
3. Address the difficulties that occurred while preparing an Expert Testimony.

Model Question Paper
Course Code: 20MCA267

Course Name: Cyber Forensics

Max. Marks :60 Duration: 3 Hrs

Part A
Answer all questions.
Each question carries 3 marks (10 x 3 = 30 Marks)

1. Categorize the formats used to store the collected digital evidences.
2. What do you mean by a computer crime? Which activities are considered as company policy violations?
3. Enumerate the features of Resilient File System.
4. Write down the operations involved in boot sequence.
5. Differentiate between soft link and hard link.
6. Which are the data acquisition tools available in Linux that is forensically sound?
7. List the features of Wireshark tool.

8. List different types of mobile forensic acquisition procedures.
9. State the guidelines for writing a report which is admissible in a court of law.
10. What are the different types of forensics reports?

(10 x 3=30 marks)

Part B

Answer all questions. Each question carries 6 marks. (5 * 6 = 30 Marks)

11. What is evidence bag? Describe standard operating procedures for securing evidence before transporting it to forensic lab. (6)

OR

12. How the retention policy of evidence related to evidence storage mediums? (6)

13. Explain the importance of Windows Registry in forensics analysis. (6)

OR

14. What is a solid-state storage device? Discuss the usage of Microsoft BitLocker tool. (6)

15. Explain the file structures of Linux and MacOS. (6)

OR

16. Define write blocker? Explain the use of Sleuth Kit tool. (6)

17. Explain the standard operating procedures used for mobile forensics. (6)

OR

18. Discuss the standard operating procedures used for network forensics. (6)

19. List and explain the steps involved in generating forensics report using Autopsy tool. (6)

OR

20. What are the responsibilities of a Computer Forensic Investigator? (6)

(5 x 6=30 Marks)

Syllabus

Module 1: (8 Hours)

Overview of computer crime, Overview of company policy violation, Preparing a case - Planning an investigation, Securing evidence. Industrial espionage investigation.

Conducting an investigation: Gathering evidence, Bit-stream copy of evidence.

Storage formats for storing collected digital evidence - Raw format, Proprietary formats, Advanced Forensic Format (AFF). Acquisition tools and methods. Digital evidence validation methods and tools.

Storing Digital evidence- Evidence Retention.

Familiarizing Autopsy for Windows - a free forensics tool.

Module 2: (10 Hours)

Understanding Digital data and storage systems: Understanding boot sequence, Understanding Disk Drives - Solid-state Storage Devices (SSDs).

Microsoft File Systems - Disk partitions, Understanding FAT, Understanding NTFS, MFT - file attributes, file data, NTFS compressed files, NTFS encrypted file system, Deleting NTFS file system, ReFS.

Whole disk encryption, Microsoft BitLocker. Understanding Windows Registry. Microsoft Windows startup tasks.

A practical assignment may be given in encrypting a partition of your computer hard disk drive/ encrypting USB flash drive to avoid firm-level attack.

Module 3: (10 Hours)

Linux file structures - File structures in Ext4, Hard links and Symbolic links.

Macintosh (MacOS) file structures - Forensic procedures in MacOS.

Setting up Sleuth Kit and Autopsy - Examining a case with Sleuth Kit and Autopsy, Importance of Write-blocker.

Acquiring data with a Linux boot CD - Preparing a target drive for data acquisition, Using dd and dcfldd commands.

Validating data acquisitions - Linux validation methods, Windows validation methods.

Following practical assignments may be given:

i. *Recover deleted files from pen drive*
ii. *Extract camera information from recovered images*
iii. *Extract deleted internet browsing history*
iv. *Recover deleted files from unallocated space using Autopsy*

Module 4: (10 Hours)

Understanding Mobile Device forensics - Mobile phone basics, Understanding Mobile phone hardware.

Acquisition procedures for Mobile devices, Mobile Forensic equipment, SIM card readers, Mobile phone Forensics tools and methods.

Network Forensics - The Need for Established Procedures, Securing a Network, Developing Procedures for Network Forensics, Wireshark packet analyser.

Practical assignments may be given:

i. *Identify students who use college lab facility to browse shopping websites*
ii. *Identify the hacking attempt on a closed port using ping sweep*
iii. *Using Wireshark retrieve the username and password of users who browse less secure website with Wi- Fi connection*

Module 5: (7 Hours)

Understand the importance of Forensics Reports, Types of reports, Guidelines for writing reports, Layout and presentation of reports, Generating reports with Autopsy.

Ethics and codes for Expert Witness - Forensics Examiner's role in testifying, Considerations in disqualification, Determining admissibility of evidence. Ethical difficulties in Expert Testimony, Ethical responsibilities.

Text Book

1. Bill Nelson, Amelia Phillips, Christopher Steuart, "Guide to Computer Forensics and Investigations", Cengage Learning, 6th Edition.

Reference Books

1. Marjie T. Britz, "Computer Forensics and Cyber Crime", Pearson Third Edition 2013.
2. Marie - Helen Maras "Computer Forensics: Cybercriminals, Laws, and Evidence", Jones & Bartlett Learning, Second Edition 2015.

Web References

1. https://www.wireshark.org/download/docs/user-guide.pdf (Reference for Wireshark)
2. http://www.open.edu/openlearn/futurelearn/cyber-security

Course Contents and Lecture Schedule

Sl. No.	Topic	No. of Lectures
1	**Module 1**	**8 Hours**
1.1	An overview of computer crimes and company policy violations	1
1.2	Preparing a case - Planning an investigation, Securing evidence. Industrial espionage investigation	1
1.3	Conducting an investigation: Gathering evidence, Bit-stream copy of evidence	1
1.4	Storage formats for storing collected digital evidence - Raw format, Proprietary formats, Advanced Forensic Format (AFF)	1
1.5	Acquisition tools and methods	1
1.6	Digital evidence validation methods and tools	1
1.7	Storing Digital evidence -Evidence Retention	1
1.8	Familiarizing Autopsy for Windows - a free forensics tool	1
2	**Module 2**	**10 Hours**
2.1	Understanding Digital data and storage systems, Understanding boot sequence	1
2.2	Understanding Disk Drives	1
2.3	Solid-state Storage Devices (SSDs)	1
2.4	Microsoft File Systems - Disk partitions	1
2.5	Understanding FAT	1
2.6	Understanding NTFS, MFT - file attributes, file data	1
2.7	NTFS compressed files, NTFS encrypted file system	1
2.8	Deleting NTFS file system, ReFS	1
2.9	Whole disk encryption, Microsoft BitLocker	1
2.10	Understanding Windows Registry, Microsoft Windows startup tasks	1
3	**Module 3**	**10 Hours**
3.1	Linux file structures - File structures in Ext4	1
3.2	Hard links and Symbolic links	1
3.3	Macintosh (MacOS) file structures - Forensic procedures in MacOS	1
3.4	Setting up Sleuth Kit and Autopsy - Examining a case with Sleuth Kit and Autopsy	1
3.5	Importance of Write-blocker	1

3.6	Acquiring data with a Linux boot CD	1
3.7	Preparing a target drive for data acquisition	1
3.8	Using dd and dcfldd commands	1
3.9	Validating data acquisitions - Linux validation methods	1
3.10	Windows validation methods	1
4	**Module 4**	**10 Hours**
4.1	Understanding Mobile Device forensics - Mobile phone basics	1
4.2	Understanding Mobile phone hardware	1
4.3	Acquisition procedures for Mobile devices	1
4.4	Mobile Forensic equipment	1
4.5	SIM card readers	1
4.6	Mobile phone Forensics tools and methods	1
4.7	Network Forensics - The Need for Established Procedures	1
4.8	Securing a Network	1
4.9	Developing Procedures for Network Forensics	1
4.10	Wireshark packet analyser	1
5	**Module 5**	**7 Hours**
5.1	Understand the importance of Forensics Reports, Types of reports	1
5.2	Guidelines for writing reports, Layout and presentation of reports	1
5.3	Generating reports with Autopsy	1
5.4	Ethics and codes for Expert Witness - Forensics Examiner's role in testifying	1
5.5	Considerations in disqualification, Determining admissibility of evidence	1
5.6	Ethical difficulties in Expert Testimony	1
5.7	Ethical responsibilities	1

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
20MCA269	COMPILER DESIGN	ELECTIVE	3	1	0	4

Preamble: The objective of this course is to explore the principles, algorithms and data structures involved in the design of compilers. It includes lexical analysis, parsing techniques, generating grammars, intermediate code generation, code optimization and code generation phases.

Prerequisite: Nil

Course Outcomes: After completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Explain different phases of compiler and perform lexical analysis using the concepts of regular expressions and finite automata.	Level 2: Understand
CO 2	Develop top down and bottom-up parsers to perform syntax analysis using context free grammar.	Level 3: Apply
CO 3	Explain syntax directed translation schemes and type checking for a given grammar.	Level 2: Understand
CO 4	Distinguish different intermediate code representations and generate intermediate code for statements in high level languages.	Level 2: Understand
CO 5	Describe various code optimization techniques and generate machine dependent code.	Level 2: Understand

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	2	2		-	-	-	2	-	-	-	-	-
CO 2	2	2	2	-	-	-	2	-	-	-	-	-
CO 3	2	2		-	-	-	2	-	-	-	-	-
CO 4	2	1		-	-	-	2	-	-	-	-	-
CO 5	2	3		-	-	-	2	-	-	-	-	-

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category Levels	Continuous Assessment Tests		End Semester Examination
	1	2	
Level 1: Remember	15	15	20
Level 2: Understand	25	35	30
Level 3: Apply	10		10
Level 4: Analyse			
Level 5: Evaluate			
Level 6: Create			

Mark distribution

Total Marks	Continuous Internal Evaluation (CIE)	End Semester Examination (ESE)	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test (2 numbers) : 20 marks
Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be *two* parts; **Part A** and **Part B**. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer *all* questions. Part B contains 2 questions from each module of which student should answer *any one*. Each question carries 6 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO 1):

1. Explain how the regular expressions and finite automata are used for specification and recognition of tokens.
2. State the role of lexical analyser. Identify the lexemes and their corresponding tokens in the following statement: printf ("Simple Interest=%f\n", si);
3. Draw the DFA for the regular expression (a | b) * (abb | a+ b).
4. Trace the output after each phase of the compiler for the assignment statement: a = b + c * 10, if variables given are of float type.

Course Outcome 2 (CO 2):

1. Find the LR (0) items for the grammar
   S->SS | a| €.
2. Show the steps involved in recursive descent parsing with backtracking for the string cad with the given grammar: S -> cAd, A -> ab | a
3. Construct the predictive parsing table for the following grammar:
   S -> (L) | a
   L -> L,S | S

Course Outcome 3 (CO 3):

1. Write the S-attributed SDD of a simple desk calculator and show annotated parse tree for the expression (3+4) *(5+6).
2. Explain bottom- up evaluation of S- attributed definitions.
3. Explain the specification of a simple type checker

Course Outcome 4 (CO 4):

1. Draw DAG for the expression (a/10 + (b -10)) *(a/10 + (b-10)). Also write the sequence of instructions used for the DAG construction.
2. Write the three-address code sequence for the statement x=y*z + y*-z. Also give its triple representation.
3. Write syntax directed definitions to construct syntax tree and three address code for assignment statements.

Course Outcome 5 (CO 5):

1. Using code generation algorithm generate code sequence for the expression x = (a – b) + (a + c).
2. With suitable example of a basic block, explain the code-improving transformations of a basic block.
3. Explain common sub expression elimination with an example.

Model Question Paper
Course Code: 20MCA269

Course Name: COMPILER DESIGN

Max. Marks :60 **Duration: 3 Hrs**

Part A

Answer all questions. Each question carries 3 marks (10 * 3 = 30 Marks)

1	State the role of lexical analyzer. Identify the lexemes and their corresponding tokens in the following statement: printf ("Simple Interest=%f\n", si);	(3)
2	Draw the transition diagram for the regular definition, relop → < | <= | = | <> | >= | >	(3)
3	Find the FIRST and FOLLOW of the non-terminals in the grammar S-> aABe A-> Abc|b B-> d	(3)
4	Demonstrate the identification of handles in operator precedence parsing?	(3)
5	What is a Syntax Directed Definition? Show an example.	(3)
6	Distinguish between synthesized and inherited attributes.	(3)
7	Write the three-address code sequence for the statement x=y*z + y*-z. Also give its triple representation.	(3)
8	Discuss about the getreg() function in code generator algorithm	(3)
9	Identify any two issues in the design of a Code Generator.	(3)
10	Explain common sub expression elimination with an example.	(3)

Part B

Answer all questions. Each question carries 6 marks. (5 * 6 = 30 Marks)

11	Explain the working of different phases of a compiler. Illustrate with a source language statement.	(6)

OR

12	Explain how the regular expressions and finite state automata are used for the specification and recognition of tokens?	(6)
13	Construct LALR parse table for the grammar S->C C->cC|d	(6)
	OR	
14	Design a recursive descent parser for the grammar E->E + T | T T->T*F | F F->(E) | id	(6)
15	Differentiate between S-attributed and L-attributed definitions with suitable examples	(6)
	OR	
16	Write the Syntax Directed Definition for a simple type declaration and draw the annotated parse tree for the declaration float a, b, c.	(6)
17	Explain the following and show an example for each. i). Three-address code iii). Triples ii). Quadruples iv). Indirect triples	(6)
	OR	

18	Write Syntax Directed Definition to produce three-address code for Boolean expressions and obtain the three-address code for the statement given below:
	
	while a < b do
	if c < d then
	x = y + z
	else
	x = y – z
		(6)

19	Explain different code optimization techniques available in local and global optimizations?	(6)
	OR	
20	Write the code generation algorithm. Using this algorithm generate code sequence for the expression x = (a – b) + (a + c).	(6)

Syllabus

Module 1 (8 Hours)
Introduction to compilers: Analysis of the source program, Phases of a compiler, Grouping of phases Lexical analysis: role of lexical analyser, input buffering, specification of tokens, recognition of tokens, Deterministic and Non-Deterministic Finite automata, Regular expression to NFA and DFA
Module 2 (12 Hours)
Syntax analysis: Role of parser, Context free grammars Top down parsing: Recursive Descent parsing, Predictive parsing, LL(1) Grammars. Bottom-up parsing: Shift Reduce Parsing, Operator Precedence Parsing (concepts only), LR parsing – Constructing SLR parsing tables, Constructing Canonical LR parsing tables and Constructing LALR parsing tables.
Module 3 (8 Hours)
Syntax directed translation: Syntax directed definitions, Bottom-up evaluation of S-attributed definitions, L- attributed definitions, Top-down translation, Bottom-up evaluation of inherited attributes. Type Checking: Type systems, Specification of a simple type checker.
Module 4 (7 Hours)
Intermediate code generation: Graphical representations, Three address code - Quadruples - triples - Indirect triples, Assignment Statements, Boolean Expressions, Control flow statements
Module 5 (10 Hours)
Code Optimization: Principal sources of optimization, Optimization of Basic blocks, Global data flow analysis. Code generation: Issues in the design of a code generator, The target machine, Basic blocks and flow graphs, A simple code generator, Peephole optimization.

Note : Programming assignments using lexical analyser generator, using parser generator.

Text Books

1. Alfred V.Aho , Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, Compilers – Principles, Techniques and Tools, Addison Wesley, 2nd Edition,2006.

Reference Books

1. V Raghavan- Principles of Compiler Design – Tata McGraw Hill, 2nd edition,2011
2. Jean Paul Tremblay and Sorenson., The Theory and Practice of Compiler Writing ,McGraw Hill,2nd Edition,2006
3. Nandini Prasad, Principles of compiler design, Elsevier, 2nd Edition,2012
4. Kenneth C. Louden, Compiler Construction-Principles and Practice, 2nd Edition, Cengage, 2010.
5. Keith Cooper and Linda Torczon, Engineering a Compiler, 2nd Edition, Elsevier, 2011

Course Contents and Lecture Schedule

Sl. No.	Topic	No. of Lectures
1	**Lexical Analysis**	**8 Hours**
1.1	Lexical Analysis: Analysis of the source program	1
1.2	Phases of a compiler, Grouping of phases	1
1.3	Lexical analysis: role of lexical analyser, input Buffering	1
1.4	specification of tokens, recognition of tokens	1
1.5	Deterministic and Non-Deterministic Finite automata	2
1.6	Regular expression to NFA and DFA	2
2	**Syntax Analysis**	**12 Hours**
2.1	Syntax analysis: Role of parser, Context free grammars	1
2.2	Top-down parsing: Recursive Descent parsing	2
2.3	Predictive parsing, LL(1) Grammars	2
2.4	Bottom-up parsing: Shift Reduce Parsing	1
2.5	Operator Precedence Parsing	1
2.6	LR parsing – Constructing SLR parsing tables	2
2.7	Constructing Canonical LR parsing tables	2
2.8	Constructing LALR parsing tables.	1

3	**Syntax directed translation and Type Checking**	**8 Hours**
3.1	Syntax directed translation: Syntax directed definitions	1
3.2	Bottom- up evaluation of S attributed definitions, L- attributed definitions	2
3.3	Top-down translation, Bottom-up evaluation of inherited attributes.	2
3.4	Type Checking: Type systems	1
3.5	Specification of a simple type checker.	2
4	**Intermediate code generation**	**7 Hours**
4.1	Intermediate code generation: Graphical representations	2
4.2	Three address code-quadruples -triples-Indirect triples	2
4.3	Assignment Statements, Boolean Expressions, Control flow statements	2
4.4	Control flow statements	1
5	**Code Optimization and Code Generation**	**10 Hours**
5.1	Code Optimization: Principal sources of optimization	2
5.2	Optimization of Basic blocks,	1
5.3	Global data flow analysis	2
5.4	Code generation: Issues in the design of a code generator.	2
5.5	The target machine, Basic blocks and Flow graphs	2
5.6	Peephole optimization	1

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
20MCA281	INTERNET OF THINGS	ELECTIVE	3	1	0	4

Preamble: This course intends to provide insight into new innovations that will build novel type of interactions among things and humans, and enables the realization of smart cities, infrastructures, and services for enhancing the quality of life and utilization of resources. An overview of IOT and its related concepts, different IOT architectures and their components, emerging paradigms such as Fog computing, Platforms and solutions supporting development and deployment of IOT applications, message passing mechanisms such as RPC, REST, and CoAP, data and knowledge management, data confidentiality, data integrity, and operation control issues faced by IOT are included in the course.

Prerequisite: Basic concepts of Information Technology and Internet.

Course Outcomes: After completion of the course the student will be able to

CO No:	Course Outcome (CO)	Blooms Category Level
CO 1	Describe the main concepts and features of the IOT paradigm.	Level 2: Understand
CO 2	Discuss Fog computing, TinyOS - nesC and programming frameworks for IOT	Level 2: Understand
CO 3	Describe the data management techniques applied to the IOT environment.	Level 2 Understand
CO 4	Explain security, and privacy in IOT environments	Level 2 Understand
CO 5	Discuss key enablers and solutions to enable practical IoT systems	Level 2 Understand

Mapping of course outcomes with program outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3						2					
CO 2	3	1					2					
CO 3	3	1					2					
CO 4	3	1					2					
CO 5	3	1	1				2					

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	20	20	20
Understand	30	30	40

Apply			
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	**CIE**	**ESE**	**ESE Duration**
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test (2 numbers) : 20 marks
Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have a maximum of 2 subdivisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Compare SOA-based architecture and API-oriented architecture.
2. Neatly sketch the open IOT architecture for IOT/CLOUD convergence.
3. List and explain the applications of device/cloud collaboration.

Course Outcome 2 (CO2)

1. What are the advantages associated with Fog computing?
2. Comment on the four broad requirements that motivate the design of TinyOS.
3. Summarize the communication paradigms and technologies used in resource-constrained environments.

Course Outcome 3(CO3):

1. Explain stream and stream processing in IOT.
2. Write and explain the algorithm for distributed anomaly detection by clustering ellipsoids.
3. Discuss the general architecture of a stream-processing system in IOT.

Course Outcome 4 (CO4):

1. Give an overview on the security requirements of IOT.
2. How can you nullify the impact of fault in high-availability cluster?
3. Explain the BCK with pre-shared keys for TinyTO.

Course Outcome 5 (CO5):

1. Give an overview on the Wired Gateway Interfaces.
2. List the features to select the gateway hardware.
3. List the steps to prepare Raspberry Pi for the execution.

Model Question Paper
Course Code: 20MCA281

Course Name: INTERNET OF THINGS

Max. Marks :60 **Duration: 3 Hrs**

Part A

Answer all questions. Each question carries 3 marks (10 * 3 = 30 Marks)

1. What do you mean by computation offloading?
2. Explain the framework that enables collaboration between smart mobile devices and cloud.
3. Outline the major challenges faced in the Fog paradigm.
4. Explain Polyglot Programming.
5. Which are the challenges faced by stream-processing systems?
6. Explain anomaly detection and categorize anomalies in the data.
7. List the different ways that an IOT gateway can extend connectivity to nodes.
8. Write the advantages of obfuscation and diversification techniques.
9. Explain Inter-Integrated Circuit (I^2C) or Two Wire Interface (TWI).
10. Write a short note on Zigbee.

Part B

Answer all questions. Each question carries 6 marks. (5 * 6 = 30 Marks)

11. Explain the taxonomy of Resource Management in IOT. (6 Marks)

OR

12. Draw and explain the state diagram of the open IOT services life cycle. (6 Marks)

13. a. Comment on the four broad requirements that motivate the design of TinyOS (3 Marks)

    b. Describe the design decisions for nesC. (3 Marks)

OR

14. List the features in coordination languages - Linda, eLinda, Orc, and Jolie (6 Marks)

15. Compare Stream Management System (DSMS) and Complex Event Processing (CEP). (6 Marks)

OR

16. Describe hyper ellipsoidal model for anomaly detection. (6 Marks)

17. Describe the error detection techniques which are applicable in the context of an IOT. (6 Marks)

OR

18. Explain the Station-to-Station protocol (STS) and the two main shortcomings of STS. (6 Marks)

19. Discuss the sensors required to build the environmental-sensing IoT gateway device for weather monitoring. (6 Marks)

OR

20. List and explain the six steps for the development of a sensor project. (6 Marks)

Syllabus

Module 1 (9 Hours)
Overview of Internet of Things: Open-source semantic web infrastructure for managing IOT resources in the Cloud - Device/Cloud Collaboration framework for intelligence applications.
Module 2 (11 Hours)
Introduction to Fog Computing: principles, architectures, and applications. TinyOS – NesC, Programming frameworks for Internet of Things
Module 3 (8 Hours)
Stream processing in IoT: foundations, state-of-the-art, and future directions - A framework for distributed data analysis for IoT
Module 4 (9 Hours)
Security and privacy in the Internet of Things- Internet of Things - robustness and reliability. TinyTO: two-way authentication for constrained devices in the Internet of Things - Obfuscation and diversification for securing the Internet of Things

Module 5 (8 Hours)
Creating a simple IoT project - Preparing Raspberry Pi – Interfacing the hardware - Internal representation of sensor values- Persisting data - Creating the actuator project - Creating a controller.

More detailed knowledge may be acquired through seminars, assignments and talks by eminent external experts and also by implementing a micro project.

Any one of the following or similar micro projects may be given as part of the course.

1. Smart Gas Leakage Detector
2. Night Patrol at home

Text Books

1. RajkumarBuyya; Amir VahidDastjerdi , "Internet of Things", Morgan Kaufmann, 2016

Reference Books

1. Peter Waher, "Learning Internet of Things", Packt Publishing, 2015
2. S. SitharamaIyengar; Nandan Parameswaran; Vir V. Phoha; N. Balakrishnan; Chuka Okoye, "Fundamentals of Sensor Network Programming: Applications and Technology", Wiley, December 14, 2010
3. Robert Stackowiak, Art Licht, VenuMantha, Louis Nagode, "Big Data and The Internet of Things: Enterprise Information Architecture for A New Age", Apress, 2015

Web Resources

1. https://www.coursera.org/specializations/internet-of-things
2. http://web.mit.edu/professional/digital-programs/courses/IoT

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Introduction (9 Hours)**	
1.1	Internet of things- definition, evolution. Applications -Smart home applications, Health care, Elder care, Traffic surveillance.	1
	SOA -Based Architecture, API oriented Architecture, Resource Management.	1
	Computational Offloading, Identification and Resource/Service Discovery, IOT Data Management and Analytics, IOT and the CLOUD	1

1.2	Open IOT architecture for IOT/Cloud convergence, Sensor middleware, Cloud computing infrastructure, Directory service, Global Scheduler, Local Scheduler component, Service delivery and utility manager	1
	Workflow of open IOT platform, Scheduling process and IOT Services lifecycle, State diagram of the Open IOT Services lifecycle within the scheduler module	1
	Scheduling and resource management, Resource optimization schemes, Caching technique	1
	Service creation flowchart, Comparison of cost - with cache server and public cloud data-score	1
1.3	Runtime adaptation engine, Device/cloud collaboration framework	1
	applications of device/cloud collaboration, Semantic QA cache	1
2	**Programming frameworks (11 Hours)**	
2.1	Introduction to Fog Computing: principles, architectures, and Applications	1
	Motivating scenario for Fog Computing, Advantages of Fog Computing, Reference architecture of Fog Computing	1
	Software-Defined Resource management layer, Services of Software-Defined Resource management layer, Applications of Fog Computing.	1
2.2	History of TinyOS, Implementation, Requirements motivating the design of TinyOS, Component Model, Interfaces. TinyOS computational concepts	1
	Overview of TinyOS Execution Model, Concurrency, TinyOS Theory of Execution: Events & Tasks, TinyOS Architecture. TinyOS-Programming Model.	1
2.3	nesC design, Component Implementation, Design Decisions for nesC, Module Components, Configuration Components	1
	Whole-Program Analysis, Detecting Race Conditions, Dealing with Race Conditions, Issues for nesC.	1
2.4	Overview of Embedded Programming Languages- nesC, Keil C, Dynamic C, B#, Message Passing in Devices-Remote Procedure Call (RPC), Lightweight RPC (LRPC)	1
	Representational state transfer (REST), Computational REST (CREST), Constrained Application Protocol(CoAP), Comparison of HTTP and CoAP, Advantages of CoAP	1

	Coordination Languages- Orchestration, Choreography, Linda and eLinda, Orc, Features of Orc, Java Orchestration Language Interpreter Engine (Jolie), Polyglot Programming, Inverse pyramid for Polyglot Programming.	1
	Features of programming frameworks for IOT, IOT programming approaches, Existing IOT frameworks	1
3	**Data management techniques (8 Hours)**	
3.1	Stream, Stream Processing, Data Stream Management System (DSMS)	1
	Complex Event Processing (CEP), differences between two use-cases of Stream Processing: DSMS and CEP	1
	The characteristics of stream data in IOT, general architecture of a stream-processing system in IOT	1
	Continuous logic processing system, challenges in stream-processing systems.	1
3.2	Anomaly detection, problem statement and definitions	1
	Hyper ellipsoidal anomaly detection	1
	Distributed anomaly detection	1
	Clustering ellipsoids, incremental local modeling	1
4	**Security and privacy (9 Hours)**	
4.1	IOT security threats, IOT security requirements, security frameworks for IOT, IOT security overview, IOT gateways and security, IOT routing attacks	1
	Security frameworks for IOT - Lightweight cryptography, asymmetric LWC algorithms, privacy in IOT networks	1
4.2	IOT characteristics and reliability issues, reliability challenges	1
	Addressing reliability, security aspects and solutions	1
4.3	TinyTO: Two-way authentication for constrained devices in the Internet of Things	1
	TinyTO protocol	1
	BCK with pre-shared keys for TinyTO, handshake implementation	1
4.4	IOT network stack and access protocols**,** Obfuscation and diversification techniques	1

	Enhancing the security in IOT using obfuscation and diversification techniques, motivations and limitations, different use-case scenarios on software diversification and obfuscation.	1
5	**IoT Implementation (8 Hours)**	
5.1	Three key components to an IOT architecture, Sensor to gateway communication - wired gateway interfaces, wireless gateway interfaces	1
	Sensors - sensors required to build the environmental-sensing IOT gateway device for weather monitoring	1
	Gateway, Gateway hardware, Gateway software	1
	Data transmission - advanced message queuing protocol, backend processing, to CLOUD or not to cloud	1
5.2	Creating a simple sensor project - Preparing Raspberry Pi – Clayster libraries	1
	Hardware, Interfacing the hardware - Internal representation of sensor values- Persisting data	1
	External representation of sensor values, Exporting sensor data	1
	Creating the actuator project – Hardware, Interfacing the hardware, Creating a controller	1

20MCA283	DEEP LEARNING	CATEGORY	L	T	P	CREDIT
		ELECTIVE	3	1	0	4

Preamble: This course intends to provide insight into deep learning. This topic is currently a much sought-after skill and is under active research. Students have to refer appropriate research papers and multiple books to get in-depth knowledge about the topics**.** Instructors may give suitable programming assignments to augment the material covered in the classroom.

Prerequisite: Basic concepts of linear algebra, probability and optimization.

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Explain the basic concepts of deep learning.	Level 2: Understand
CO 2	Design neural networks using TensorFlow	Level 3: Apply
CO 3	Solve real world problems with CNN.	Level 3: Apply
CO 4	Solve real world problems with RNN.	Level 3: Apply
CO 5	Describe the concepts of GAN.	Level 2: Understand

Mapping of course outcomes with program outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	2	2										
CO 2	3	3	3		3		3					
CO 3	3	3	3		3		3					
CO 4	3	3	3		3		3					
CO 5	2	3			2		2					

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	15	15	10
Understand	25	25	30
Apply	10	10	20
Analyze			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test (2 numbers) : 20 marks
Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Describe the model of a biological neuron.
2. Explain Perceptron learning algorithm.
3. Explain the role of batch normalization in training a neural network.

Course Outcome 2 (CO2)

1. Draw and demonstrate the VGG-16 architecture.
2. Sketch the AlexNet architecture and explain its functionalities.

Course Outcome 3(CO3):

1. Design a convolutional neural network which can classify MNIST handwritten data.
2. An input image has been converted into a matrix of size 12 X 12 along with a filter of size 3 X 3 with a Stride of 1. Determine the size of the convoluted matrix.
3. Why do we prefer Convolutional Neural networks (CNN) over Artificial Neural networks (ANN) for image data as input?

Course Outcome 4 (CO4):

1. You are given an image data set with 10 classes. Describe how you will use deep learning to build a classifier.
2. Design a system to generate deep fakes from an image.

Course Outcome 5 (CO5):

1. Describe auto encoders and how they help in dimensionality reduction.
2. Explain how GANS work.

Model Question Paper
Course Code: 20MCA283

Course Name: DEEP LEARNING

Max. Marks :60 **Duration: 3 Hrs**

Part A

Answer all questions. Each question carries 3 marks (10 * 3 = 30 Marks)

1. Describe sigmoid activation functions.
2. Write the gradient descent algorithm.
3. Explain with an example how graphs are stored and represented in TensorFlow.
4. Discuss how graph representation can accelerate computing models.
5. Describe the VGG 16 architecture.
6. What is max pooling in the context of CNN?
7. Explain ReLU.
8. Explain the problem of vanishing gradients.
9. Write a note on auto encoders.
10. Explain the idea behind cross entropy.

Part B

Answer one full question from each module, each carries 6 marks.

11. (a) Describe the model of a biological neuron. 3 marks

    (b) Explain perceptron learning algorithm. 3 marks

OR

12. With a suitable example explain how backpropagation works 6marks

13. Explain the role of batch normalization in training a neural network and describe how to find out overfitting from training and validation curves 6 marks

OR

14. Explain the ideas of Rank, Shape and Type with an example in the context of a Tensor Data Structure 6 marks

15. With a suitable numerical example illustrate convolution operation. 6 marks

OR

16. Explain the architecture of AlexNet. 6 marks

17. Explain the idea of Truncated backpropagation through time. 6 marks

OR

18. Describe how LSTM works. 6 marks

19. Distinguish between generative and discriminative models 6 marks

OR

20. Explain how a GAN is trained. 6 marks

Syllabus

Module I (8 Hours)
Review of Neural Networks: Model of a biological neuron, McCulloch Pitts Neuron, Activation Functions, Perceptron, Perceptron Learning Algorithm and Convergence, Multilayer Perceptron, Back propagation, Learning XOR, Sigmoid Neurons, Gradient Descent, Feed forward Neural Networks.
Module II (10 Hours)
Training Neural Networks: Initialization, dropout, batch normalization and dropout, overfitting, underfitting, training and validation curves. Data Visualization: Feature and weight visualization, tSNE. Introduction to TensorFlow: graphs, nodes, Tensor data structures - rank, shape, type, Building neural networks with TensorFlow, Introduction to Keras.
Module III (10 Hours)
Convolutional Neural Networks: Convolution operation, Convolutional layers in neural network, pooling, fully connected layers. Case study: Architecture of Lenet, Alexnet and VGG 16
Module IV (8 Hours)
Recurrent Neural Networks: Back propagation, vanishing gradients, exploding gradients, truncated backpropagation through time, Gated Recurrent Units (GRUs), Long Short-Term Memory (LSTM) cells, solving the vanishing gradient problem with LSTMs.
Module V (9 Hours)
Autoencoders, variational autoencoders. Generative Adversarial Networks (GAN): Discriminative and generative models, GAN discriminator, GAN generator, upsampling, GAN Training, GAN challenges, loss functions, cross entropy, minimax loss, Wasserstein loss.

Programming assignments using TensorFlow maybe given at the end of each module to get hands on experience.

Textbooks.

1. Generative Deep Learning: David Foster, OReily, (2019)
2. Deep Learning, Ian Goodfellow, Yoshua Bengio and Aaron Courville, MIT press (2016)
3. Hands on Machine Learning with Scikit Learn and TensorFlow, Aurélien Géron (2019)
4. Deep Learning Illustrated, Jon Krohn, Grant Beyleveld, Aglae Bassens, Pearson, 1st Edn., (2020)
5. Online book Dive Deep into Machine Learning at https://d2l.ai/

References

Module 1

a. https://www.cse.iitm.ac.in/~miteshk/CS6910/Slides/Lecture2.pdf
b. https://www.cse.iitm.ac.in/~miteshk/CS6910/Slides/Lecture3.pdf

Module 2

a. http://neuralnetworksanddeeplearning.com
b. Hands on Machine Learning with Scikit Learn and TensorFlow, Aurélien Géron
c. Probabilistic Machine Learning: An Introduction, Kevin Murphy
d. https://www.researchgate.net/publication/228339739_Viualizing_data_using_t-SNE

Module 3

a. https://www.cse.iitm.ac.in/~miteshk/CS7015/Slides/Teaching/pdf/Lecture11.pdf
b. Convolutional neural networks for visual computing (Chapter 4), Ragav Venkatesan and Baoxin Li CRC press

Module 4

a. On the difficulty of training RNNs: https://arxiv.org/pdf/1211.5063.pdf
b. LSTM: A Search Space Odyssey: https://arxiv.org/abs/1503.04069
c. Understanding Deriving and Extending the LSTM: https://r2rt.com/written-memories-understanding-deriving-and-extending-the-lstm.html
d. Understanding LSTM Networks: http://colah.github.io/posts/2015-08-Understanding-LSTMs/
e. https://www.cse.iitm.ac.in/~miteshk/CS7015/Slides/Teaching/pdf/Lecture14.pdf
f. https://www.cse.iitm.ac.in/~miteshk/CS7015/Slides/Teaching/pdf/Lecture15.pdf

Module 5

a. GANs in Action: Deep Learning with Generative Adversarial Network Jakub Langgr, Vladimir Bok
b. Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play David Foster
c. https://developers.google.com/machine-learning/gan

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Module 1**	**8 Hours**
1.1	Review of Neural Networks: Model of a biological neuron	1
1.2	McCulloch Pitts Neuron, Activation functions	1
1.3	Perceptron, Perceptron Learning Algorithm	1
1.4	Convergence, Multilayer Perceptron	1
1.5	Back propagation	1
1.6	Learning XOR, Sigmoid Neurons	1
1.7	Gradient Descent, Feed forward Neural Networks	2
2	**Module 2**	**10 Hours**
2.1	Training Neural Networks	1
2.2	Initialization, Dropout	1
2.3	Batch normalization and drop out	1
2.4	Over fitting, under fitting, training and validation curves, data visualization, feature and weight visualization, tSNE	2
2.5	Introduction to TensorFlow, graphs, nodes, Tensor Data Structures - rank, shape, type	2
2.6	Building neural networks with tensor flow	2
2.7	Introduction to Keras	1
3	**Module 3**	**10 Hours**
3.1	Convolutional neural networks	1
3.2	Convolution operation	2
3.3	Back propagation in multilayer neural networks	1
3.4	Convolutional layers in neural network, pooling	2
3.5	Fully connected layers	2
3.6	Case study: Architecture of Lenet, Alexnet and VGG 16	2
4	**Module 4**	**8 Hours**
4.1	Recurrent neural networks	1
4.2	Back propagation: vanishing gradients, exploding gradients	1
4.3	Truncated Backpropagation Through Time	1
4.4	LSTM	1
4.5	Gated Recurrent Units (GRUs)	1
4.6	Long Short-Term Memory (LSTM) Cells	1
4.7	Solving the vanishing gradient problem with LSTMs	2
5	**Module 5**	**9 Hours**
5.1	Autoencoders, Variational autoencoders	2
5.2	Generative Adversarial Networks (GAN)	1
5.3	Discriminative and generative models	2
5.4	GAN Discriminator, GAN Generator, upsampling,	1
5.5	GAN Training	1
5.6	GAN challenges, Loss functions, cross entropy, minimax loss, Wasserstein loss	2

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
20MCA285	DIGITAL IMAGE PROCESSING	ELECTIVE	3	1	0	4

Preamble: This course introduces the techniques of simulating human vision into computer vision based on feature extraction to develop applications in different areas. The concept of enhancement, transforms, smoothing, restoration, compression, morphological image analysis, classification & segmentation in two-dimensional space are introduced. This course serves as a prerequisite for many advanced courses in computer vision areas.

Prerequisite: Nil

Course Outcomes: After completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Discuss the fundamental concepts of digital image processing, image formation and representation of images.	Level 2: Understand
CO 2	Summarise image enhancement methods in the spatial domain.	Level 2: Understand
CO 3	Explain image transforms and image smoothing & sharpening using various kinds of filters in frequency domain.	Level 2: Understand
CO 4	Describe various methods in image restoration and compression.	Level 2: Understand
CO 5	Discuss morphological basics and image segmentation methods.	Level 2: Understand

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	2	2					2					
CO 2	2	2					2					
CO 3	2	2					2					
CO 4	2	2					2					
CO 5	2	2					2					

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category Levels	Continuous Assessment Tests		End Semester Examination
	1	2	
Level 1: Remember	15	15	20
Level 2: Understand	35	35	40
Level 3: Apply			
Level 4: Analyse			
Level 5: Evaluate			
Level 6: Create			

Mark distribution

Total Marks	Continuous Internal Evaluation (CIE)	End Semester Examination (ESE)	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test (2 numbers) : 20 marks
Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have a maximum of 2 sub-divisions and carry 6 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO 1):

1. List out various components of an Image Processing System.
2. Define Electromagnetic Spectrum.
3. Illustrate the image formation in the eye. Calculate the size of the retinal image of a tree, if the observer is looking at a tree 20 m high at a distance of 100.

Course Outcome 2 (CO 2):

4. Describe the basic relationships and distance measures between pixels in a digital image.
5. List and explain steps in Histogram Processing.
6. List and explain various Intensity transformation functions used in grey scale images.

7. Explain the process of Unsharp masking?

Course Outcome 3 (CO 3):

1. Explain the properties of Unitary transform.
2. Compare and contrast 1D-DFT and 2D-DFT.
3. Design a basic Laplacian filter using first order and second order derivatives.
4. Describe various image smoothing techniques using frequency domain filters.

Course Outcome 4 (CO 4):

1. Explain image noise models and list out different noise probability density functions used in image processing applications.
2. Describe Wiener filtering technique.
3. Draw the functional block diagram of image compression system. List various types of redundancy in compression?

Course Outcome 5 (CO 5):

1. Differentiate erosion and dilation in morphological processing.
2. Compare Global thresholding and Otsu's method.
3. Explain how does Hough transform works.

Model Question Paper

Course Code: 20MCA285

Course Name: Digital Image Processing

Max. Marks :60 **Duration: 3 Hrs**

Part A

Answer all questions.

Each question carries 3 marks (10 x 3 = 30 Marks)

.

1. Describe the elements of visual perception.
2. Define Toeplitz & Circulant matrices
3. Explain histogram equalization in detail.
4. Differentiate linear spatial filter and non-linear spatial filter.
5. Explain the properties of 2D DFT.
6. List the steps involved in frequency domain filtering.
7. Write note on Point Spread Function.
8. List the components of a compression system.
9. Compare opening and closing in morphological processing of images.
10. Explain the merits and demerits of edge thresholding in segmentation.

(10 x 3=30 marks)

Part B

Answer all questions. Each question carries 6 marks. (5 * 6 = 30 Marks)

11. Explain fundamental steps in Digital Image Processing. (6)

OR

12. Differentiate sampling and quantization in image processing. (6)

13. Explain basic grey level transformation in spatial domain. (6)

OR

14. Compare Unsharp masking and High-boost filtering in Spatial filtering (6)

15. Explain Discrete Cosine Transform and its properties. (6)

OR

16. Explain the working of Homomorphic filtering with an example. (6)

17. Explain image restoration process in detail. (6)

OR

18. Differentiate lossy and lossless image compression methods (6)

19. Compare erosion and dilation in Morphological image analysis (6)

OR

20. Explain canny edge detector in detail (6)

(5 x 6=30 Marks)

Syllabus

Module 1: Overview of Digital Image Processing (9 Hours)
Digital Image Processing: Basic concepts, Difference between image processing and computer vision, Components of an image processing system. Image processing applications. Mathematical preliminaries: Basic Vector and Matrix operations, Toeplitz, Circulant, Unitary & Orthogonal matrices. Elements of Visual Perception: Structure of the human eye and image formation, Brightness adaptation and discrimination. Types of Images: Binary, Gray scale and Color Images. Image Sampling and Quantization: Digital image as a 2D array, Spatial and Intensity resolution, 2D-sampling theorem. RGB and HSI color models.
Module 2: Concept of Image enhancement & Spatial filtering (10 Hours)
Concept of Image enhancement, Basic grey level transformation functions: Image negative, Log transformation, Power-law transformation, Piecewise linear transformations. Histogram of an Image, Histogram equalization with illustration. Fundamentals of Spatial Filtering: Mechanics of Spatial filtering, 2D correlation and convolution. Smoothing spatial filters: Linear and Nonlinear types. Sharpening spatial filters: Laplacian operator, Unsharp masking and High-boost filtering, Gradient based operators for image sharpening.
Module 3: Image Transform & Filtering in frequency domain (8 Hours)
Image Transform-representation of an image in frequency domain, Unitary transformation of an Image-transform pair equations in matrix form, Properties of unitary transforms. 1D-DFT, 2D-DFT of an image- Properties of 2D-DFT. DCT and its properties, Filtering an Image in the Frequency Domain– Steps of frequency domain filtering. Basic concept and illustration of frequency domain image smoothing and sharpening.
Module 4: Image Restoration & Compression (8 Hours)
Image Restoration: Concept of Image restoration, A Model of the Image Degradation/Restoration Process, Image Noise Models, Point Spread Function, Restoration using Inverse filtering, Wiener filtering. Image compression: Need for compression, redundancy, classification of image compression schemes, A general image compression system, Huffman coding, Transform based compression, JPEG standard, Digital image watermarking-basic concept.
Module 5: Basics of morphological image processing & image segmentation (10 Hours)
Morphological image processing basics: erosion and dilation, opening and closing, Hit or Miss transformation.

Image segmentation: Fundamentals, Point detection, Line detection, Basic steps of edge detection, Hough transform, Edge detectors - Marr-Hildreth edge detector & Canny edge detector. Thresholding: Basics of intensity thresholding, Global thresholding and Otsu's method. Region-based segmentation: Region growing, Region Splitting and Merging.

Text Books

1. Rafael C., Gonzalez & Woods R.E., "Digital Image Processing", Pearson Education.

2. Jain A.K, "Fundamentals of Digital Image Processing", Prentice Hall, Eaglewood Cliffs, NJ.

Reference Books

1.Schalkoff R. J., "Digital Image Processing and Computer Vision", John Wiley 5. Pratt W.K., "Digital Image Processing", John Wiley

2.Al Bovick , "Handbook of Image and Video Processing" , Academic Press, 2000

Course Contents and Lecture Schedule

Sl. No.	Topic	No. of Lectures
1	**Overview of Data Image Processing**	**9 Hours**
1.1	Basic concepts of digital image processing, Image processing application	2
1.2	Mathematical preliminaries: Basic Vector and Matrix operations	1
1.3	Toeplitz, Circulant, Unitary & Orthogonal matrices	1
1.4	Elements of Visual Perception, Structure of human eye	1
1.5	Brightness adaptation and discrimination, Types of Images	1
1.6	Sampling and Quantization	1
1.7	Spatial and Intensity resolution, 2D-sampling theorem.	1
1.8	RGB and HSI color models.	1
2	**Concept of Image enhancement & Spatial filtering**	**10 Hours**
2.1	Concepts of Image enhancement, Basic grey level transformation functions: Image negative, Log transformation, Power-law transformation, Piecewise linear transformations.	2
2.2	Histogram of an Image, Histogram equalization with illustration	1

2.3	Fundamentals of Spatial Filtering: Mechanics of Spatial filtering	1
2.4	2D correlation and convolution	1
2.5	Smoothing spatial filters: Linear and Nonlinear types	1
2.6	Sharpening spatial filters: Laplacian operator	1
2.7	Unsharp masking	1
2.8	High-boost filtering	1
2.9	Gradient based operators for image sharpening	1
3	**Image Transform & Filtering in frequency domain**	**8 Hours**
3.1	Image Transform-representation of an image in the transform domain.	1
3.2	Unitary transformation of an Image, Properties of unitary transforms	1
3.3	1D-DFT	1
3.4	2D-DFT and its properties	1
3.5	DCT and its properties	1
3.6	Filtering an Image in the Frequency Domain– Steps of frequency domain filtering	1
3.7	Image smoothing using frequency domain filters – Ideal Lowpass Filters, Butterworth Lowpass filter & Gaussian Lowpass filter.	1
3.8	Image sharpening using frequency domain filters – Ideal Highpass Filters, Butterworth Highpass filter & Gaussian Highpass filter.	1
4	**Image Restoration & Image compression**	**8 Hours**
4.1	Image Restoration: Concept of Image restoration	1
4.2	A Model of the Image Degradation/Restoration Process	1
4.3	Image Noise Models	1
4.4	Point Spread Function	1
4.5	Restoration using Inverse filtering, Wiener filtering	1
4.6	Image compression: Need for compression, redundancy, classification of image compression schemes	1
4.7	A general image compression system Huffman coding, Transform based compression.	1
4.8	JPEG standard, Digital image watermarking-basic concept.	1
5	**Basics of morphological image processing & image segmentation**	**10 Hours**
5.1	Morphological image processing basics: erosion and dilation, opening and closing, Hit or Miss transformation.	1
5.2	Image segmentation: Fundamentals, Point detection, Line detection	1
5.3	Basic steps of edge detection - Hough transform	2
5.4	Edge detectors: Marr-Hildreth edge detector	1

5.5	Canny edge detector	1
5.6	Thresholding: Basics of intensity thresholding.	1
5.7	Global thresholding	1
5.8	Otsu's method	1
5.9	Region-based segmentation: Region growing, Region Splitting and Merging.	1

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
S20MCA287	BIOINFORMATICS	ELECTIVE	3	1	0	4

Preamble: This course helps to understand the concepts of computational biology and bioinformatics. The students will learn Database tools and their uses, various algorithms for biological sequence analysis, Genomics and Gene Recognition, Protein structure and to use various visualization techniques, data mining & machine learning in bioinformatics.

Prerequisite: Nil

Course Outcomes: After completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Explain the fundamentals of Computational Biology and Bioinformatics.	Level 2: Understand
CO 2	Classify various biological databases.	Level 2: Understand
CO 3	Use suitable algorithm for Biological Sequence Analysis and make use of database search tools.	Level 3: Apply
CO 4	Discuss Gene structure and expression of Prokaryotic and Eukaryotes.	Level 2: Understand
CO 5	Apply data mining & machine learning methods to analyse and visualize biological data.	Level 3: Apply

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	1	1				1					
CO 2	3	3	1				2					
CO 3	3	3	2				2					
CO 4	3	2	1				1					
CO 5	3	3	2		2		2					

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category Levels	Continuous Assessment Tests		End Semester Examination
	1	2	
Level 1: Remember	10	10	10
Level 2: Understand	30	30	30
Level 3: Apply	10	10	20
Level 4: Analyse			
Level 5: Evaluate			
Level 6: Create			

Mark distribution

Total Marks	Continuous Internal Evaluation (CIE)	End Semester Examination (ESE)	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks

Continuous Assessment Test (2 numbers) : 20 marks

Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be *two* parts; **Part A** and **Part B**. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer *all* questions. Part B contains 2 questions from each module of which student should answer *any one*. Each question can have a maximum 2 subdivisions and carry 6 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO 1):

1. Explain the concept of DNA
2. Explain the concept of RNA.
3. Illustrate the concept of translation and transcription.
4. Discuss Gnome project and its impact on bioinformatics

Course Outcome 2 (CO 2):

1. Explain the features of biological databases?
2. Discuss primary sequence databases and secondary sequence databases.
3. Classify the two important classification schemes of structure classification databases.
4. Retrieve the sequence from primary / secondary databases.
5. Use of BLAST for comparing sequences.

Course Outcome 3 (CO 3):

1. Explain the importance of scoring matrices in sequence alignment.
2. Explain the different algorithms used for sequence alignment .
3. Illustrate Local and global alignment Algorithm for the sequence CGTGAATTCAT (sequence#1 or A) GACTTAC (sequence #2 or B)
4. Compute the best alignment of these two sequences: ACTGATTCA ACGCATCA Using -2 as a gap penalty, -3 as a mismatch penalty, and 2 as the score for a match.

Course Outcome 4 (CO 4):

1. Explain the Prokaryotic gene structure
2. Explain the Eukaryotic gene structure
3. Demonstrate the usage of Open Reading Frame with an example
    a. 5′-ATCTAAAATGGGTGCC-3′
4. Explain the working principle of microarray

Course Outcome 5 (CO 5):

1. Differentiate between the different protein molecular structure visualizations.
2. Use Web-based Map Viewer program, RasMol, PyMol data visualization techniques in bioinformatics .
3. Use PubMed to search for a particular pattern to specify the importance of mining the biomedical literature for data on functions to complement the sequence and structure data mined from nucleotide and protein databases.
4. Compare any three machine learning technologies and their applicability to data mining methods.

Model Question Paper

Course Code: S20MCA287

Course Name: BIOINFORMATICS

Max.Marks :60 Duration: 3 Hrs

Part A

Answer all questions. Each question carries 3 marks (10 * 3 = 30 Marks)

1. Write short note on “how genome carries hereditary data from organisms.”
2. What is antisense RNA?
3. Write a short note on primary database.

4. Write note on : (a) SCOP (b) CATH.
5. How many times faster is to find the best alignment for the sequences "RQQEPURSTC"
   and "QQESGPVRST" using N_W algorithm compared to assessing each possible alignment one by one?
6. Define raw score, bit score and e-value in BLAST.
7. Write a short note on of process is gene expression in Prokaryotic.
8. Justify the reasons for the high Prokaryotic gene density compared to Eukaryotes.
9. Differentiate between centralized and distributed data mining Infrastructure. Use diagrams if necessary.
10. Explain the significance of Hidden Markov Model in bioinformatics. Draw a sample Markov chain which is the basis of HMM, also mention how HMM is different from Markov chain.

Part B

Answer all questions. Each question carries 6 marks. (5 * 6 = 30 Marks)

11	What is the Central dogma of Molecular biology? How can Molecular biology be considered as an information science?	6
	OR	
12	With a neat diagram describe the structural and functional differences between DNA and RNA?	6
13	Explain different types of protein databases and its applications in bioinformatics.	6
	OR	
14	Differentiate between Composite protein sequence database and secondary databases .	6
15	Align the following sequence using Needleman and Wunch algorithm for global alignment ATTGC and AGGC with match +1, mismatch -1 and gap penalty -2. What is the score of the optimal global alignment?	6
	OR	
16	Find the best local alignment between ACCTAGG and GGCTCAATCA with +2 for a match, -1 for a mismatch and -2 for a gap using Smith Waterman Algorithm. Explain Prokaryotic Gene structure with neat diagram.	6
17	What is GC content? How it differs in eukaryotic and prokaryotic genomes.	6
	OR	
18	Describe with the help of a diagram the generation of cDNAs. Mention its use and also write notes on ESTs.	6

19	Illustrate with the help of a neat diagram the pattern recognition and the label discovery process.	6
	OR	
20	Justify the importance of user interfaces in data visualization. With the support of a representative block diagram explain the structure of a 3D protein visualization tool. Also explain the UI components of the same.	6

Syllabus

Module 1: Computational Biology and Bioinformatics (7 Hours)
Computational Biology: Cell - Central Dogma of Molecular Biology - Structure of DNA, RNA and Protein - Coding and Non-coding RNAs - mRNA, tRNA, miRNA and siRNA. Bioinformatics: Nature & Scope of Bioinformatics, Gnome projects, Importance of bioinformatics, Pattern recognition and prediction.
Module 2: Biological Databases (8 Hours)
Biological Databases, Primary Sequence Databases, Composite protein sequence databases, Secondary Databases, Composite protein pattern databases, Structure classification databases. //Tutorial class may be arranged to the introduction and use of sequence retrieval from the databases.
Module 3: Data Searches and pairwise Alignment (10 Hours)
Dot Plots, Concept of Simple Alignment, Scoring matrices: Introduction to PAM & Blosum, Needleman and Wunsch Algorithm, Global and Local Alignments, Smith Waterman Algorithm, Multiple Sequence Alignment. Familiarize Database search tools: BLAST & FastA //Tutorial class may be arranged to the introduction and use of sequence alignment and BLAST.
Module 4: Genomics and Gene recognition (10 Hours)
Introduction to Gene expression in prokaryotes, Prokaryotic Gene structure, GC content in prokaryotic genomes, Gene Density. Eukaryotic Genomes: Gene structure, GC content in eukaryotic genomes, Gene Expression – Introduction to Microarrays.
Module 5: Data Visualization, Data mining and Machine learning (10 Hours)
Data Visualization - Introduction, Sequence Visualization, Structure Visualization, User Interface, Animation Versus Simulation, General-Purpose Technologies.

Data Mining using biological data, Methods, Infrastructure, Pattern recognition and discovery, Genetic Algorithms, Neural networks using biological data, Statistical methods using biological data, Introduction to Hidden Markov Models and Text mining.

//Tutorial class may be arranged to introduce and use - RasMol and PyMol .

Text Books

1. Dan. E. Krane and M. L. Raymer, "Fundamental Concepts of Bioinformatics", Pearson Education, 2003(Module 4)

2. Bryan Bergeron, M.D, "Bioinformatics Computing", Pearson Education, 2015. (Module 1,5)

3. Attwood T. K. and D. J. Parry-Smith," Introduction to Bioinformatics", Pearson Education, 2003 (Module 2,3)

4. Neil C Jones and Pavel A Pevzner, "An Introduction to Bioinformatics Algorithms", MIT Press, 2004

Reference Books

1. Jean-Michel Claverie and Cedric Notredame, "Bioinformatics For Dummies" , 2nd Edition,Wiley Publishing
2. David W Mount, "Bioinformatics- Sequence and Genome Analysis " , 2/e, Cold Spring Harbor
3. Laboratory Press, New York.
4. "Bioinformatics for Dummies" J. Claverie & C. Notredame ,Wiley India..

Web Reference

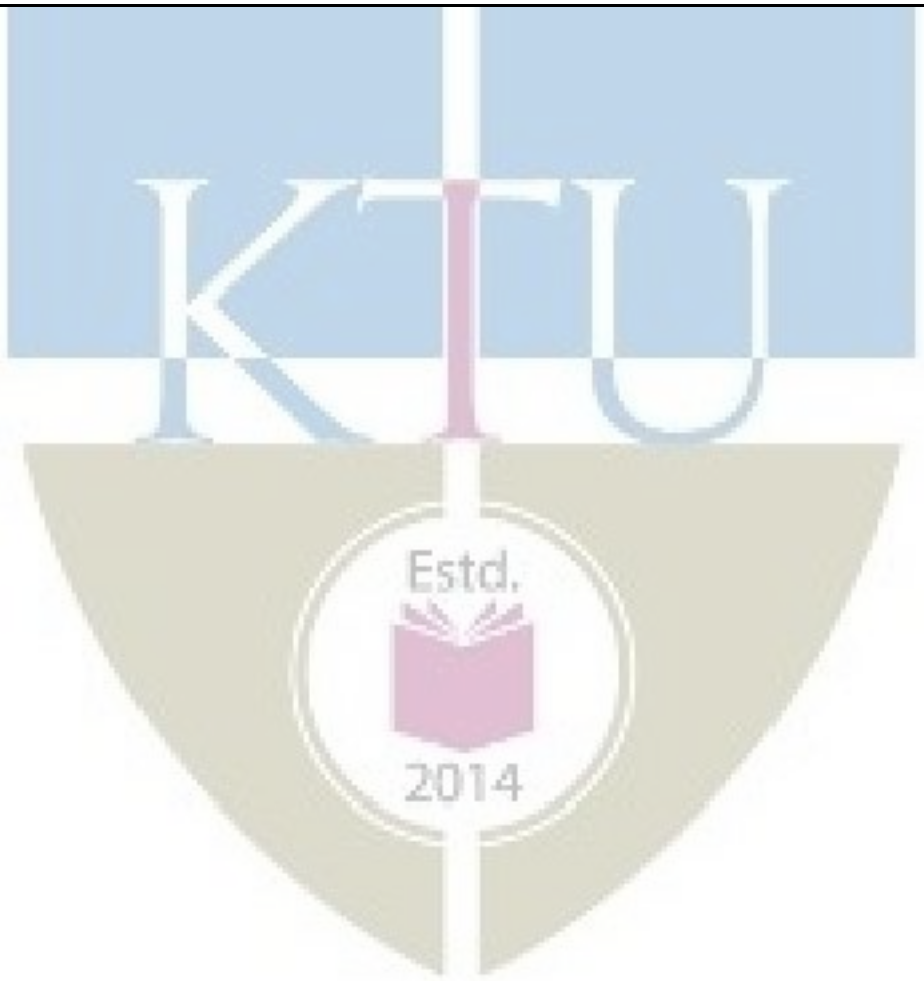
1. https://nptel.ac.in/courses/102/106/102106065/

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Computational Biology and Bioinformatics**	**(7 Hours)**
1.1	Cell - Central Dogma of Molecular Biology, Structure of DNA	1
1.2	RNA and Protein: Coding and Non-coding RNAs -mRNA	1
1.3	tRNA, miRNA and siRNA	1
1.4	Nature & Scope of Bioinformatics, Gnome projects	1
1.5	Importance of bioinformatics, Pattern recognition and prediction	1
1.6	Folding problem	1
1.7	Sequence analysis, homology and analogy	1

2	**Biological Databases**	**(8 Hours)**
2.1	Primary Sequence databases: Nucleic acid and Protein sequence: PIR, MIPS, SWIS-PROT	1
2.2	Protein sequence: TrEMBL, NRL-3D	1
2.3	Composite protein sequence Databases: NRDB, OWL, MIPSX and SWISS-PROT+TrEMBL	1
2.4	Secondary Databases, Need for Secondary databases	1
2.5	Prosite	1
2.6	Prints	1
2.7	Blocks, Profile, Pfam, Identify	1
2.8	Composite Protein Pattern Database and Structure Classification Databases	1
3	**Data Searches and pairwise Alignment**	**(10 Hours)**
3.1	Dot Plots	1
3.2	Concept of Simple Alignment, GAPS	1
3.3	Scoring matrices	1
3.4	Introduction to PAM	1
3.5	Introduction to Blosum	1
3.6	Needleman and Wunsch Algorithm	1
3.7	Global and Local Alignments: Semiglobal alignment	1
3.8	Smith Waterman Algorithm	1
3.9	Alignment scores and statistical significance of database search , Multiple Sequence Alignment.	1
3.10	Familiarize Database search tools: BLAST & FastA	1
4	**Gene structure and expression of Prokaryotic and Eukaryotes.**	**(10 Hours)**
4.1	Introduction to Gene expression in Prokaryotes	1
4.2	Prokaryotic Gene structure	1
4.3	GC content in prokaryotic genomes	1
4.4	Prokaryotic Genomes -Gene Density	1
4.5	Eukaryotic Genomes	1
4.6	Gene structure , ORF in Prokaryotic	1

4.7	GC content in Eukaryotic Genomes	1
4.8	Gene Expression - cDNAs & ESTs,	1
4.9	Serial Analysis of Gene Expression	1
4.10	Introduction to Microarrays.	1
5	**Data Visualization, Data mining & Machine learning**	**(10 Hours)**
5.1	Data Visualization Introduction	1
5.2	Sequence Visualization- Sequence Map	1
5.3	Structure Visualization- Rendering tools	1
5.4	User Interface - User Interface Components, Alternative Metaphors, Display Architecture	1
5.5	Animation Versus Simulation, General-Purpose Technologies.	1
5.6	Data Mining, Methods, Infrastructure	1
5.7	Pattern recognition and discovery	1
5.8	Genetic Algorithms	1
5.9	Neural networks, Statistical methods	1
5.10	Hidden Markov Models and Text mining	1



20MCA289	SOCIAL NETWORK ANALYSIS	CATEGORY	L	T	P	CREDIT
		ELECTIVE	3	1	0	4

Preamble: This course intends to provide insight into social network analysis. The objective of this course is to enable students analyse and visualize network data. This course will create an understanding about the semantic web, structure of various social networks and the structure of search engines.

Prerequisite: Basic concepts of graph theory and networks

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Explain the basic concepts of semantic web and social network analysis.	Level 2: Understand
CO 2	Describe the ontology-based knowledge representation techniques in social network.	Level 2: Understand
CO 3	Discuss aggregation of social network information and representation of social individuals and social relationships.	Level 2: Understand
CO 4	Describe the structure of the Web and Facebook as a graph and the algorithms for searching and community discovery.	Level 2: Understand
CO 5	Explain the general architecture of a search engine and specifically the Google search engine architecture.	Level 2: Understand

Mapping of course outcomes with program outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	2	2					1					
CO 2	2	2					1					
CO 3	2	2					2					
CO 4	2	3		2	2	2	2			2		
CO 5	2	3		2	2		2					

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	15	15	20

Understand	35	35	40
Apply			
Analyze			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test (2 numbers) : 20 marks
Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Explain the development of semantic Web and the emergence of Social Web.
2. Describe the global structure social networks.
3. Discuss in detail about the macro-structure of social networks.
4. "Most network analysis methods work on an abstract, graph-based representation of real-world networks". Justify this statement.

Course Outcome 2 (CO2)

1. Describe the characteristics of Resource Description Framework (RDF).
2. Compare the features of Web Ontology Language (WOL) and Unified Modeling Language (UML).
3. Compare the features of Web Ontology Language (WOL) and Entity Relationship (ER) Model.

Course Outcome 3(CO3):

1. Describe the ontological representation of social individuals.
2. Explain the generic architecture of Semantic Web applications.
3. Discuss how semantic web applications can be built with social network features?

Course Outcome 4 (CO4):

1. Describe Zipf's Law.
2. Write the limitations of HyperANF Algorithm and explain how it can be sorted out using the Iterative Fringe Upper Bound (iFUB) Algorithm.
3. What is meant by Degree Assortativity? What is the use of this measure?
4. "A user who logs in more generally has more friends on Facebook", describe how can we conclude this statement.

Course Outcome 5 (CO5):

1. Draw the architecture of a general search engine and explain how it works.
2. Explain how the HITS Algorithm works to assign ranks to web pages.
3. Compare the HITS Algorithm and the Page Rank Algorithm.

Model Question Paper
Course Code: 20MCA289

Course Name: SOCIAL NETWORK ANALYSIS

Max. Marks :60 **Duration: 3 Hrs**

Part A

Answer all questions. Each question carries 3 marks (10 * 3 = 30 Marks)

1. What is meant by semantic web?
2. Write notes on personal networks.
3. Define Electronic discussion networks.
4. List out the features of blogs that can be used for social network extraction.
5. Explain how the reasoning with instance equality is done in social network data?
6. What is meant by Evaluating Smushing?
7. Define "Power Law".
8. What is "Spid"? How it is used to differentiate between web-network and social network?
9. What are the basic functions of the storage repository of a search engine?
10. How can we identify web spam pages?

Part B

Answer all questions. Each question carries 6 marks. (5 * 6 = 30 Marks)

11. List and explain various measures in network analysis. 6 marks

OR

12. Describe the macro-structure of social networks. 6 marks

13. What is meant by ontology-based knowledge representation? Explain its role in the semantic web. 6 marks

OR

14. Compare the features of Web Ontology Language (WOL) and Extensible Markup Language (XML). 6 marks

15. Describe how aggregating and reasoning can be done on social network data. 6 marks

OR

16. Discuss the ontological representation of social relationships. 6 marks

17. Define the following with suitable example:
    a) Rank exponent 2 marks
    b) Hop plot exponent 2 marks
    c) Eigen exponent 2 marks

OR

18. Explain how to generate in-degree and out-degree distributions on the graph of the Web crawl. 6 marks

19. Describe how the web crawler module in a search engine does the page selection and page refresh. 6 marks

OR

20. Draw the architecture of Google search engine and comment on each of its components. 6 marks

Syllabus

Module I (9 Hours)
Introduction to the Semantic Web and Social Networks: The Semantic Web, Limitations of the current Web, The semantic solution, Development of the Semantic Web, The emergence of the social web, Social Network Analysis**,** Development of Social Network Analysis, The global structure of networks, The macro-structure of social networks, Personal networks.
Module II (8 Hours)
Electronic sources for network analysis: Electronic discussion networks, Blogs and online communities, Web-based networks.

Knowledge Representation on the Semantic Web: Ontologies and their role in the Semantic Web, Ontology languages for the Semantic Web, The Resource Description Framework (RDF) and RDF Schema, The Web Ontology Language (OWL), Comparison of Ontology languages with the Unified Modelling Language (UML), Comparison to the Entity/Relationship (E/R) model and the Relational model, Comparison to the Extensible Markup Language (XML) and XML Schema.
Module III (8 Hours)
Modelling and aggregating social network data: Network data representation, Ontological representation of social individuals, Ontological representation of social relationships, Aggregating and reasoning with social network data, Representing identity, On the notion of equality, Determining equality, Reasoning with instance equality, Evaluating smushing.
Module IV (10 Hours)
Graph Structure of the Web: Breadth First Search (BFS) Algorithm, Strongly Connected Components (SCC) Algorithm, Weakly Connected Components (WCC) Algorithm, In-degree and out- degree distributions, Connected Components, Zipf's Law, Rank Exponent R, Out-Degree Exponent O, Hop Plot Exponent H, Eigen Exponent E. **Graph Structure of Facebook:** Hyper ANF Algorithm, Iterative Fringe Upper Bound (iFUB) Algorithm, Spid, Degree Distribution, Path Length, Component Size, Clustering Coefficient and Degeneracy, Friends-of-Friends, Degree Assortativity, Login Correlation, Effects of Age, Gender and Country of Origin.
Module V (10 Hours)
Link Analysis: Search Engine – Search engine architecture, Crawling, Storage, Indexing, Ranking, HITS Algorithm, Page rank algorithm, Random walk, SALSA Algorithm, Bayesian Algorithm; Google - Google architecture, Data Structures, Crawling, Searching, Web Spam Pages.

Textbooks.

1. Social Networks and the Semantic Web, Peter Mika, Springer, 2007. (For Modules 1, 2 & 3)
2. Practical Social Network Analysis with Python, Krishna Raj P. M., Ankith Mohan, K. G. Srinivasa, Springer, 2018. (For Modules 4 & 5)

References

1. Social Network Analysis, John Scott, SAGE Publications, 4th Edition (2017)
2. Social Network Analysis - Interdisciplinary Approaches and Case Studies, Xiaoming Fu, Jar-Der Luo and Margarete Boos, CRC Press (2017)
3. Handbook of Social Network Analysis, John Scott and Peter J. Carrington, SAGE Publications (2011)

4. Social Network Analysis - Methods and Applications, Stanley Wasserman and Katherine Faust, Cambridge University Press (2012)

Web - References

1. https://onlinecourses.nptel.ac.in/noc20_cs78/preview
2. https://www.coursera.org/learn/social-network-analysis
3. https://www.coursera.org/learn/python-social-network-analysis

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Module 1**	**9 Hours**
1.1	The Semantic Web, Limitations of the current Web	1
1.2	The semantic solution	1
1.3	Development of the Semantic Web	1
1.4	The emergence of the Social Web	1
1.5	Social Network Analysis	1
1.6	Development of Social Network Analysis	1
1.7	The global structure of networks	1
1.8	The macro-structure of social networks	1
1.9	Personal networks	1
2	**Module 2**	**8 Hours**
2.1	Electronic sources for network analysis, Electronic discussion networks	1
2.2	Blogs and online communities	1
2.3	Web-based networks	1
2.4	Knowledge Representation on the Semantic Web	1
2.5	Ontologies and their role in the Semantic Web	1
2.6	Ontology languages for the Semantic Web, The Resource Description Framework (RDF) and RDF Schema	1
2.7	The Web Ontology Language (OWL), Comparison of Ontology languages with the Unified Modelling Language (UML)	1
2.8	Comparison to the Entity/Relationship (E/R) model and the Relational model, Comparison to the Extensible Markup Language (XML) and XML Schema	1
3	**Module 3**	**8 Hours**
3.1	Modelling and aggregating social network data, Network data representation	1
3.2	Ontological representation of social individuals	1
3.3	Ontological representation of social relationships	1
3.4	Aggregating and reasoning with social network data	1
3.5	Representing identity	1
3.6	Notion of equality, Determining equality	1

3.7	Reasoning with instance equality	1
3.8	Evaluating smushing	1
4	**Module 4**	**10 Hours**
4.1	Graph Structure of the Web	1
4.2	Breadth First Search (BFS) Algorithm	1
4.3	Strongly Connected Components (SCC) Algorithm, Weakly Connected Components (WCC) Algorithm	1
4.4	In-degree and out- degree distributions, Connected Components	1
4.5	Zipf's Law	1
4.6	Rank Exponent R, Out-Degree Exponent O, Hop Plot Exponent H, Eigen Exponent E	1
4.7	Graph Structure of Facebook: HyperANF Algorithm	1
4.8	Iterative Fringe Upper Bound (iFUB) Algorithm, Spid, Degree Distribution, Path Length	1
4.9	Component Size, Clustering Coefficient and Degeneracy, Friends-of-Friends	1
4.10	Degree Assortativity, Login Correlation, Effects of Age, Gender and Country of Origin	1
5	**Module 5**	**10 Hours**
5.1	Link Analysis: Search Engine – Search engine architecture	1
5.2	Crawling, Storage, Indexing	1
5.3	Ranking, HITS Algorithm	1
5.4	Page rank algorithm	1
5.5	Random walk	1
5.6	SALSA Algorithm	1
5.7	Bayesian Algorithm	1
5.8	Google - Google architecture	1
5.9	Data Structures, Crawling, Searching	1
5.10	Web Spam Pages	1

20MCA241	DATA SCIENCE LAB	CATEGORY	L	T	P	CREDIT
		LAB	0	1	3	2

Preamble: This is an introductory practical course on Data Science and student will learn how to use various scientific libraries in python to implement data mining techniques and machine learning algorithms.

Prerequisite: Fundamentals of programming, python programming fundamentals, Machine learning, fundamentals of web programming,

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Use different python packages to perform numerical calculations, statistical computations and data visualization	Level 3: Apply
CO 2	Use different packages and frameworks to implement regression and classification algorithms.	Level 3: Apply
CO 3	Use different packages and frameworks to implement text classification using SVM and clustering using k-means	Level 3: Apply
CO 4	Implement convolutional neural network algorithm using Keras framework.	Level 3: Apply
CO 5	Implement programs for web data mining and natural language processing using NLTK	Level 3: Apply

Mapping of course outcomes with program outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	3	3	1	3	2	3		2			
CO 2	3	3	3	2	3	2	3		2			
CO 3	3	3	3	2	3	2	3		2			
CO 4	3	3	3	2	3	2	3		2			
CO 5	3	3	3	2	3	3	3		2			

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember (K1)			
Understand (K2)			
Apply (K3)	50	50	50
Analyse (K4)			
Evaluate (K5)			
Create(K6)			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	50	50	3 hours

Continuous Internal Evaluation Pattern:

Maximum Marks: 50	
Attendance	7½
Maintenance of daily lab record and GitHub management	10
Regular class viva voce	7½
Timely completion of day-to-day tasks	10
Tests/Evaluation	15

End Semester Examination Pattern:

Maximum Marks: 50			
Verification of Daily program record and Git Repository			5 marks
Viva			10 marks
Problem solving (Based on difficulty level, one or more questions may be given)	Flowchart / Algorithm / Structured description of problem to explain how the problem can be solved / Interface Design	15%	35 marks
	Program correctness	50%	
	Code efficiency	15%	
	Formatted output	20%	

Course Level Assessment Questions

Course Outcome 1 (CO1):

- Review of python programming – Programs review the fundamentals of python (simple python programs ice breaker) – (at most one lab session)

- Matrix operations (using vectorization) and transformation using python and SVD using Python.
- Programs using matplotlib / plotly / bokeh / seaborn for data visualisation.
- Programs to handle data using pandas.

Course Outcome 2 (CO2)

- Program to implement k-NN classification using any standard dataset available in the public domain and find the accuracy of the algorithm.
- Program to implement Naïve Bayes Algorithm using any standard dataset available in the public domain and find the accuracy of the algorithm
- Program to implement linear and multiple regression techniques using any standard dataset available in the public domain and evaluate its performance.

Course Outcome 3(CO3):

- Program to implement text classification using Support vector machine.
- Program to implement decision trees using any standard dataset available in the public domain and find the accuracy of the algorithm.
- Program to implement k-means clustering technique using any standard dataset available in the public domain

Course Outcome 4 (CO4):

- Programs on feedforward network to classify any standard dataset available in the public domain.
- Programs on convolutional neural network to classify images from any standard dataset in the public domain.

*[Note] : Encourage students to refer standard neural network architectures such as LeNet5, ResNet, GoogLeNet etc. and use these as starting points for their models.

Course Outcome 5 (CO5):

<u>Web Data Mining</u>

- Implement a simple web crawler (ensure ethical conduct).
- Implement a program to scrap the web page of any popular website – suggested python package is scrappy (ensure ethical conduct).

Natural Language Processing

Problems may be designed for the following topics so that students can get hands on experience in using python for natural language processing:

- Part of Speech tagging
- N-gram and smoothening
- Chunking

Syllabus
Review of python programming, Matrix operations, Data Visualisation using matplotlib / plotly / bokeh / seaborn, Data handling using pandas, Classification k-NN algorithm, Naïve Bayes algorithm, Implementation of linear and multiple regression techniques, Text classification using Support vector machine, Implementation of Decision Trees, Clustering using k-means algorithm, Convolutional Neural Network to classify images using Keras framework, Web Crawler and Scrapping web pages, Implementation of NLP - Part of Speech tagging, N-gram & smoothening and Chunking using NLTK.

Reference Books

1. Christopher M Bishop, “Pattern Learning and Machine Learning”, Springer, 2006
2. E. Alpayidin, “Introduction to Machine Learning”, Prentice Hall of India (2005)
3. T. Hastie, RT Ibrashiran and J. Friedman, “The Elements of Statistical Learning”, Springer 2001
4. Toby Segaran, “Programming Collective Intelligence: Building Smart Web 2.0 Applications”, O' Reilly Media; 1 edition (16 August 2007).
5. Drew Conway, John Myles White, “Machine Learning for Hackers: Case Studies and
6. Algorithms to Get You Started”, O' Reilly Media; 1 edition (13 February 2012)
7. Simon Rogers, Mark Girolami, “A First course in Machine Learning”, CRC Press, First Indian reprint, 2015.
8. Tom Mitchell, “Machine Learning”, McGraw Hill, 1997.
9. Bing Liu, Web Data Mining - Exploring Hyperlinks, Contents and Usage Data, Second edition, Springer 2011

Course Contents and Lab Schedule

Sl No.	Topic	No. of hours
1	Review of python programming, Matrix operations, Programs using matplotlib / plotly / bokeh / seaborn for data visualisation and programs to handle data using pandas.	8
2	Program to implement k-NN classification using any standard dataset available in the public domain and find the accuracy of the algorithm	2
3	Program to implement Naïve Bayes Algorithm using any standard dataset available in the public domain and find the accuracy of the algorithm	2
4	Program to implement linear and multiple regression techniques using any standard dataset available in the public domain and evaluate its performance.	4
5	Program to implement text classification using Support vector machine.	4
6	Program to implement decision trees using any standard dataset available in the public domain and find the accuracy of the algorithm	4
7	Program to implement k-means clustering technique using any standard dataset available in the public domain	2
8	Program on convolutional neural network to classify images from any standard dataset in the public domain using Keras framework.	6
9	Program to implement a simple web crawler and scrapping web pages.	6
10	Implement problems on natural language processing - Part of Speech tagging, N-gram & smoothening and Chunking using NLTK	8

20MCA243	MOBILE APPLICATION DEVELOPMENT LAB	CATEGORY	L	T	P	CREDIT
		LAB	0	1	3	2

Preamble: This is a practical course on Mobile Application Development and student will learn how to program in Android Platform and develop applications using SQLite that run on Andriod Operating System.

Prerequisite: Basic knowledge on programming and database concepts.

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Design and develop user interfaces for mobile apps using basic building blocks, UI components and application structure using Emulator	Level 3: Apply
CO 2	Write simple programs and develop small applications using the concepts of UI design, layouts and preferences	Level 3: Apply
CO 3	Develop applications with multiple activities using intents, array adapter, exceptions and options menu.	Level 3: Apply
CO 4	Implement activities with dialogs, spinner, fragments and navigation drawer by applying themes	Level 3: Apply
CO 5	Develop mobile applications using SQLite.	Level 3: Apply

Mapping of course outcomes with program outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	3	3	1	3	2	3		2			
CO 2	3	3	3	2	3	2	3		2			
CO 3	3	3	3	2	3	2	3		2			
CO 4	3	3	3	2	3	2	3		2			
CO 5	3	3	3	2	3	3	3		2			

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember(K1)			
Understand(K2)			
Apply(K3)	50	50	50
Analyse(K4)			
Evaluate(K5)			
Create(K6)			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	50	50	3 hours

Continuous Internal Evaluation Pattern:

Maximum Marks: 50	
Attendance	7½
Maintenance of daily lab record and GitHub management	10
Regular class viva voce	7½
Timely completion of day-to-day tasks	10
Tests/Evaluation	15

End Semester Examination Pattern:

Maximum Marks: 50			
Verification of Daily program record and Git Repository			5 marks
Viva			10 marks
Problem solving (Based on difficulty level, one or more questions may be given)	Flowchart / Algorithm / Structured description of problem to explain how the problem can be solved / Interface Design	15%	35 marks
	Program correctness	50%	
	Code efficiency	15%	
	Formatted output	20%	

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Design a Login Form with username and password using LinearLayout and toast valid credentials
2. Write a program that demonstrates Activity Lifecycle.
3. Implementing basic arithmetic operations of a simple calculator
4. Implement validations on various UI controls

Course Outcome 2 (CO2)

1. Design a registration activity and store registration details in local memory of phone using Intents and SharedPreferences
2. Design a simple Calculator using GridLayout and Cascaded LinearLayout
3. Create a Facebook page using RelativeLayout; set properties using .xml file
4. Develop an application that toggles image using FrameLayout

Course Outcome 3(CO3):

1. Implement Adapters and perform exception handling
2. Implement Intent to navigate between multiple activities
3. Develop application that works with explicit intents
4. Implement Options Menu to navigate to activities
5. Develop an application that uses ArrayAdapter with ListView.

Course Outcome 4 (CO4):

1. Develop an application that use GridView with images and display Alert box on selection
2. Develop an application that implements Spinner component and perform event handling
3. Apply themes via code and manifest file
4. Develop application using Fragments
5. Implement Navigation drawer

Course Outcome 5 (CO5):

1. Create database using SQLite and perform INSERT and SELECT
2. Perform UPDATE and DELETE on SQLite database
3. Develop an application as a micro project which uses SQLite database as an assignment

Syllabus

Fundamentals: Basic Building blocks – Activities, Services, Broadcast Receivers and Content providers, UI Components – Views and notifications Components for communication -Intents and Intent Filters
Application Structure: AndroidManifest.xml, user-permission – sdk, Resources and R.java, Assets, Layouts and Drawable Resources, Activities and Activity lifecycle.
Emulator-Android Virtual Device: Launching emulator, Editing emulator settings, Emulator shortcuts, Logcat usage, Introduction to DDMS
Basic UI design: Form widgets, Text Fields, Validation of EditText, Layouts, [dip, dp, sip, sp] versus px
Preferences: Shared Preferences, Preferences from xml
Menu: Option menu, Context menu, menu from xml, menu via code
Intents: Explicit Intents, Implicit intents
UI design: Time and Date, Images and media, Android Adapter and ListView, Composite, Alert Dialogs and Toast, Popup, Fragments, Navigation drawer
Tabs, Tab Activity Styles & Themes: styles.xml, drawable resources for shapes, gradients (selectors), style attribute in layout file, Applying themes via code and manifest file
Content Providers: SQLite Programming, SQLite Open Helper, SQLite Database, Cursor, Reading and updating Contacts, Reading bookmarks

Reference Books

1. Joseph Annuzzi Jr, Lauren Darcey, Shane Condor, "Advanced Android Application Development, Developers Library", Pearson Education, 4th Edition (2015)

2. Lauren Darcey, Shane Condor, "Android, Wireless Application Development", Pearson Education, 3rd Edition.

3. Paul Deitel, Harvey Deitel, Alexander Wald, "Android 6 for programmers, An AppDriven Approach", Pearson Education

4. Rap Payne, "Beginning App Development with Flutter: Create Cross-Platform Mobile Apps", Apress (2019)

Course Contents and Lecture Schedule

Sl No	Topic	No. of hours
1	Fundamentals – Basic building blocks	3
2	Application structure, layout and resources	3
3	Android Virtual Device, Activity Lifecycle	3
4	Basic UI Design and EditText Validation	4
5	Shared Preferences, RelativeLayout, FrameLayout, GridLayout and Preferences from xml	9
6	ArrayAdapter, ListView and Exception handling	3
7	Various Menu options	3
8	Explicit and Implicit Intents	3
9	Images and media, Dialogs, Spinner component, Popups, Fragments, Navigation drawer	6
10	Applying themes and styles .xml	3
11	SQLite Programming	6

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
20MCA245	MINI PROJECT	PROJECT	-	-	4	2

Preamble: This project work aims to enable the students to apply the software engineering principles on a real software project, to make the students familiar with the stages of a deployment pipeline and to develop a software product using the latest software development methodology.

Prerequisite: Knowledge in software engineering principles and programming skills.

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Identify a real-life project which is useful to society / industry	Level 2: Understand
CO 2	Interact with people to identify the project requirements	Level 3: Apply
CO 3	Apply suitable development methodology for the development of the product / project	Level 3: Apply
CO 4	Analyse and design a software product / project	Level 4: Analyse
CO 5	Test the modules at various stages of project development	Level 5: Evaluate
CO 6	Build and integrate different software modules	Level 6: Create
CO 7	Document and deploy the product / project	Level 3: Apply

Mapping of course outcomes with program outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	2	3	3	3	1	2	3	3	3	3	3	3
CO 2	2	3	2	3	2	3	2	1	3	2	3	
CO 3	3	3	3	3	3	1	3	3	1		2	
CO 4	3	3	3	3	3		3	3	1	1	2	
CO 5	3	3	3	3	3		2	3			1	
CO 6	3	3	3	3	3	2	3	3		2	3	3
CO 7	1	1	3	3	3	2	3	3	2	1	2	

3/2/1: High/Medium/Low

Mark distribution

Total Marks	CIE	ESE
100	100	-

Assessment Criteria

Class participation and attendance	10%
Evaluation	50%
Class work	40%

Marks Division

Continuous evaluation by Supervisor, Scrum Master and Project Guide	50 Marks
Interim evaluation by the Project Assessment Board	25 Marks
Final evaluation by the Project Assessment Board	25 Marks
Total	**100 Marks**

Guidelines:

- Students shall identify Real-Life Projects which are relevant and useful to the society or industry.
- The project shall be an individual project and must be done in-house. The student has to spend time in the lab for the project work.
- Attendance as per MCA regulations is applicable for submitting the project for final evaluation.
- Students shall submit project synopsis and get prior approval from the Project (Faculty) Supervisor before the project work begins.
- If there is a customer for the project then he/she will be the Product Owner (External Guide) and a faculty from the department will be the Internal Guide. If there is no such customer then the Internal Guide himself/herself shall act as the Product Owner.
- A faculty / technical staff shall act as the Scrum Master to continuously monitor the project development. Periodic meetings, of less than 15 minutes, at the convenience of

the Scrum Master are to be highly encouraged. Ensure such meetings occur once in three days.

- Set a sprint as two weeks, ensure biweekly reviews. A review shall not exceed 30 minutes. A demo to the Product Owner (Project Guide) is mandatory in every review.

- The student shall maintain a Scrum Book (Rough Record) which has to be divided into 4 parts – (i) Product Backlog (ii) Database & UI Design (iii) Testing & Validation and (iv) Details of Versions. Make dated entries in the corresponding part at regular intervals. The corrections and comments from Product Owner and Scrum Master should be clearly indicated with the Date.

- Test Driven Development methodology may be practiced for the project development. BugZilla, BackLog or any such tool may be used for Bug Tracking.

- Git shall be used for Version Control and Git commit history may be verified as part of project evaluation .

- LaTeX or an equivalent tool shall be used for preparing Presentations and Project Report.

- Interim evaluations of project's progress shall be conducted as part of Internal Assessment. Project Evaluation Board may consist of Project Supervisor, Product Owner, Scrum Master and one other Faculty Member from the department. Scrum reviews shall not be sacrificed for such presentations.

- At the end of the semester entire project development activities shall be evaluated internally by the Project Evaluation Board.

Week	Schedule
1	Familiarisation with build tools (editor/IDE, compiler such as gcc with commonly used options/switches, debugger like dgb). Familiarisation with an IDE (Eclipse, NetBeans…), that supports build tools and common version control operations using Git . Familiarisation with Docker Selection of Topic, Formation of Development Team, Feasibility analysis.
2	Topic Approval, Meeting of Development Team including Scrum Master with Product Owner. Informal, preliminary discussions of requirements. Creating user stories in the rough record.

	Commencement of the Project.
3	Identifying modules, Initial Design of Database & UI. Creating a Docker container for the environment Creating an empty git repository by Scrum Master / one member of the Development team and setting permission to other members. Pushing the first version of the Project along with a Readme file containing contact details of team members. Creating pull requests for sample update of Readme by each member and merging the pull requests of one by another.
4-5	Setting up systems for development, testing and production. Design of the basic model of a simple deployment pipeline Creating a suitable folder structure (Maven's folder structure is desirable). Creating Unit tests using an XUnit framework, Writing the build and code analysis script, Writing acceptance test scripts and test cases, Setting up a Continuous Integration System like Jenkins. Automating acceptance tests with Selenium, Karate or an equivalent tool, writing a simple deployment script that uses scp/rsync or Ansible for copying the Dockerfile and running Docker with ssh. First Scrum Review. (Here onwards, the Scrum reviews are conducted on every other week)
7	Project Presentation - Interim Evaluation to be based on Git History
14	Submission of Project Report, with Scrum Book Project Presentation – Final Evaluation to be based on Git History, Scrum Book, Project Report and Presentation

References

1. Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation (Addison-Wesley Signature Series (Fowler)) 1st Edition
2. Alistair Cockburn, Agile Software Development: The Cooperative Game, Addison Wesley, 2nd Edition (2006).
3. Andrew Hunt, David Thomas, The Pragmatic Programmer: From Journeyman to Master, Pearson India, 1st Edition (2008).

4. Ken Schwaber, Mike Beedle, Agile Software Development with Scrum, Pearson (2008).
5. Lisa Crispin, Janet Gregory, Agile Testing: A Practical Guide for Testers and Agile Teams, Addison Wesley Professional, 1st Edition (2008).
6. Mike Cohn, User Stories Applied: For Agile Software Development, Addison Wesley, 1st Edition, (2004).
7. Pressman, R.S., Software Engineering: A Practitioner's Approach, McGraw Hill SE, 7th Edition, (2010).
8. Robert C. Martin, Agile Software Development, Principles, Patterns and Practices, Prentice Hall Imprint, Pearson Education, 2nd Edition (2002).\
9. Rod Stephens, Beginning Software Engineering, Wrox Series, Wiley India Pvt Ltd (2015).
10. RyPress Ry's Git Tutorial (Free e-book)

Web Reference

1. Introduction to DevOps (https://www.edx.org/course/introduction-devops-microsoft-dev212x)

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
20MCANC3	Domain Expertise Workshops	Non-Credit Course	-	-	1	Nil

Preamble: This course intends to give insight into various application domains and technology domains in IT industry.

Prerequisite: Nil

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Associate real-life problems with IT solutions	Level 2: Understand
CO 2	Describe latest developments in IT field	Level 2: Understand
CO 3	Interact with technical experts	Level 3: Apply
CO 4	Prepare technical documents	Level 3: Apply
CO 5	Present a topic before an audience	Level 2: Understand

Mapping of course outcomes with program outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	1	2				3	3		3	3	3	3
CO 2	2	2				3	3		3	3	3	3
CO 3		2				3	3		3			3
CO 4						3			3			3
CO 5						3			3			3

3/2/1: High/Medium/Low

Mark distribution

Total Marks	CIE	ESE
-	-	-

Guidelines:

- As part of this course following activities shall be done:
  - Expert talks shall be arranged to explain about various Application domains like Retail, Finance, Healthcare, Automotive, Manufacturing… and IT domains like IoT, AI, Bigdata, Full Stack Development, Robotic Process Automation…
  - Instruct students to research and submit reports about any of these domains.
  - Instruct students to study about these domains and take seminars…
- One hour in every week or two hours in alternate weeks shall be used for this course
- Staff-in-charge shall maintain a file with the records, documents and reports as hardcopies or e-copies of all the activities done on this course.
- These documents shall be submitted for verification during the academic audit.

SEMESTER -4

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
20MCA242	COMPREHENSIVE VIVA	VIVA	-	-	-	6

Preamble: Comprehensive Viva intends to assess the knowledge gained by a student in the core courses of this programme and to make the student aware of his/her knowledge level and where he/she stands after completing this programme. This course will help the student in preparing for comprehensive examinations and improve the confidence in answering questions in objective mode.

Prerequisite: Thorough knowledge in all the courses he/she learned during this programme.

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Articulate the concepts in the core courses learned through this programme.	Level 2: Understand
CO 2	Attend technical interviews with confidence.	Level 2: Understand
CO 3	Interpret questions and answer them with clarity.	Level 2: Understand
CO 4	Make use of the concepts learned through this programme in future.	Level 3: Apply

Mapping of course outcomes with program outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	3		2	2		2		3			
CO 2	3	3	1	2	3	2	3		3			
CO 3	1	2				2	2		3			
CO 4	3	2	3	2	2	3	3		2			

Mark distribution

Total Marks	CIE	ESE
100	-	100

Guidelines:

- Comprehensive viva shall be conducted within the first 20 days of the fourth semester.

- Viva shall be conducted by a panel of examiners consisting of:
    1. Head of the department
    2. A senior faculty in the department
    3. External examiner appointed by the university

- Viva shall be conducted for each student for a minimum of 20 minutes

- Knowledge level of the student shall be assessed on the following topics.
    - 20MCA105 - Advanced Data Structures
    - 20MCA107 - Advanced Software Engineering
    - 20MCA102 - Advanced Database Management Systems
    - 20MCA201 - Data Science & Machine Learning
    - 20MCA203 - Design & Analysis of Algorithms
    - Code snippets in Java or Python to solve simple problems.
    - Technologies used in the project work
    - Recent developments in the field of computer science.

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
20MCA244	SEMINAR	SEMINAR	-	-	2	2

Preamble: This course intends to enable the students to gain knowledge in any of the technically relevant current topics on Computer Science or Information Technology, and to acquire confidence in presenting the topic and preparing a report.

Prerequisite: Nil

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Annotate the ideas presented in technical papers	Level 2: Understand
CO 2	Comprehend a concept by referring different technical documents	Level 2: Understand
CO 3	Prepare technical documents	Level 3: Apply
CO 4	Present a topic before an audience	Level 3: Apply
CO 5	Interact with the audience	Level 2: Understand

Mapping of course outcomes with program outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	2	3	1	3	2		3		3	2		2
CO 2	2	3	1	3	2		3		3	2		2
CO 3	2		1	2	3	2	3		3	2		2
CO 4	2	2			3	3			3		2	
CO 5	2	2			3	3			3		2	

Mark distribution

Total Marks	CIE	ESE
50	50	-

Assessment Criteria

Scope and relevance of topic	20%
Quality of presentation slides	10%
Presentation skills	30%
Knowledge in the topic	20%
Report	20%

Marks Division

Evaluation by Faculty Guide	20 Marks
Evaluation by the Faculty Committee	30 Marks
Total	**50 Marks**

Guidelines:

- Students shall conduct detailed study on a technically relevant current topic in Computer Science / Information Technology under the supervision of a Faculty Guide and present it as a seminar at the end of the study.
- The study may be conducted on
    - articles published in reputed journals/conference proceedings
    - recent development in Computer Science / Information Technology
    - recent research and development activity in a research lab
    - latest software tool or framework
- Students shall submit an abstract on identified topic and get prior approval from the Faculty Guide before the study begins.
- The student shall submit a seminar report, based on the study and their findings. The report shall not be a reproduction of original paper or manual.
- The study and its findings shall be presented in the class taking a duration of 15-20 minutes.
- LaTeX or an equivalent tool shall be used for preparing Presentations and Seminar Report.
- Students shall be encouraged to publish their study in journals and due credit shall be given to such students.
- A committee of three senior faculty members shall constituted by the head of the department and the seminar presentation shall be evaluated by that committee.

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
20MCA246	MAIN PROJECT	PROJECT	-	-	27	12

Preamble: This project work aims to enable the students to apply the software engineering principles on a real software project, to make the students familiar with the stages of a deployment pipeline and to develop a software product using the latest software development methodology.

Prerequisite: Knowledge in software engineering principles and programming skills.

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Identify a real-life project which is useful to society / industry	Level 2: Understand
CO 2	Interact with people to identify the project requirements	Level 3: Apply
CO 3	Apply suitable development methodology for the development of the product / project	Level 3: Apply
CO 4	Analyse and design a software product / project	Level 4: Analyse
CO 5	Test the modules at various stages of project development	Level 5: Evaluate
CO 6	Build and integrate different software modules	Level 6: Create
CO 7	Document and deploy the product / project	Level 3: Apply

Mapping of course outcomes with program outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	2	3	3	3	1	2	3	3	3	3	3	3
CO 2	2	3	2	3	2	3	2	1	3	2	3	
CO 3	3	3	3	3	3	1	3	3	1		2	
CO 4	3	3	3	3	3		3	3	1	1	2	
CO 5	3	3	3	3	3		2	3			1	
CO 6	3	3	3	3	3		3	3		2	3	3
CO 7	1	1	3	3	3	2	3	3	2	1	2	

Mark distribution

Total Marks	CIE	ESE
100	70	30

Marks Division

Continuous evaluation by Supervisor, Guide(s) and Scrum Master	30 Marks (Internal)
Evaluation by the Project Assessment Board	40 Marks (Internal)
Evaluation by the External expert	30 Marks (External)
Total	**100 Marks**

Guidelines:

- Students shall identify Real-Life Projects which are relevant and useful to the society or industry.

- The project shall be an individual project and must be done in-house. The student has to spend time in the lab for the project work. Attendance as per MCA regulations is applicable for submitting the project for final evaluation.

- However, in exceptional cases students shall be given permission to work on the project outside the campus and at the industry premises if the organization offering the project belongs to anyone of the following categories.
    - CMM Level 5 Certified Company
    - Publicly listed company in India
    - National Research Institute
    - Central / State Government Department
    - Project funded by the Central / State Government Agency

- In such cases, the student is required to produce a letter from the organisation before starting the project and a committee constituted by the head of the department shall make the decision on permission. Industries and training institutes that offer project work for a fee shall not be permitted.

- Students shall submit project synopsis and get prior approval from the Project (Faculty) Supervisor before the project work begins.

- If there is a customer for the project then he/she will be the Product Owner (External Guide) and a faculty from the department will be the Internal Guide. If there is no such customer then the Internal Guide himself/herself shall act as the Product Owner.

- A faculty / technical staff shall act as the Scrum Master to continuously monitor the project development. Periodic meetings, of less than 15 minutes, at the convenience of the Scrum Master are to be highly encouraged. Ensure such meetings occur once in three days.

- The student shall maintain a Scrum Book (Rough Record) which has to be divided into 4 parts – (i) Product Backlog (ii) Database & UI Design (iii) Testing & Validation and (iv) Details of Versions. Make dated entries in the corresponding part at regular intervals. The corrections and comments from Product Owner and Scrum Master should be clearly indicated with the Date.

- Test Driven Development methodology may be practiced for the project development. BugZilla, BackLog or any such tool may be used for Bug Tracking.

- Git shall be used for Version Control and Git commit history may be verified as part of project evaluation .

- LaTeX or an equivalent tool shall be used for preparing Presentations and Project Report.

- Students shall be encouraged to publish their work in journals and due credit shall be given to such students.

- For the externally done projects, periodic confidential progress report and attendance statement shall be collected from the External Guide and be reviewed by the Project Supervisor.

- Set a sprint as two weeks, ensure biweekly reviews. A review shall not exceed 30 minutes. A demo to the Product Owner (Project Guide) is mandatory in every review.

- Interim evaluations of the project's progress shall be conducted by a Project Assessment Board as part of internal assessment. Two such evaluations are desirable. Scrum reviews shall not be sacrificed for such presentations.

- The Project Assessment Board shall be constituted by the Head of the Department with the following five members.
  Chairman:
  1. Head of the Department

  Members:
  2. Project supervisor/s of the student

3. One faculty member from the Department
4. One faculty member from a sister Department
5. An external expert, either from an academic/research institute or Industry. (For the externally done projects, the external guide shall be invited as external expert.)

- At the end of the semester, two evaluations shall be there on the entire project development activities. First an internal evaluation by the Project Assessment Board and second an external evaluation by an External Examiner.

- An External Examiner either from an academic institute or industry shall be appointed by the University for the External Evaluation.

Week	Schedule
(May be scheduled inline with the KTU academic calendar)	
1	Selection of Topic, Submission of project synopsis and getting approval Meeting of Development Team including Scrum Master with Product Owner (Project Guide)
2	Commencement of the Project.
4	First Sprint release and Scrum Review by the Product Owner (Project Guide)
6	Second Sprint release and Scrum Review by the Project Guide First interim evaluation by the Project Assessment Board
8	Third Sprint release and Scrum Review by the Project Guide
10	Fourth Sprint release and Scrum Review by the Project Guide
11	Second interim evaluation by the Project Assessment Board
12	Fifth Sprint release and Scrum Review by the Project Guide
13	Submission of project report, with Scrum Book Final project presentation Evaluation by the Project Assessment Board
14	Final evaluation by the External Examiner.