ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES (A)
(UGC Autonomous)
Approved by AICTE, Affiliated to Andhra University, Accredited by
N.B.A. & NAAC with 'A' Grade
(Estd : 2001)

2021-22

Academic Regulations (R19)

Curriculum & Syllabi (III Year I&II Semesters)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

VISION

Our vision is to emerge as a world class Computer Science and Engineering department through excellent teaching and strong research environment that responds swiftly to the challenges of changing computer science technology and addresses technological needs of the stakeholders.

MISSION

To enable our students to master the fundamental principles of computing and to develop in them the skills needed to solve practical problems using contemporary computer-based technologies and practices to cultivate a community of professionals who will serve the public as resources on state-of-the-art computing science and information technology.

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

PEO-1	Employability	Work as Competent Computer Engineer either globally or locally by engaging in professional practice in a variety of roles with ability to serve as a team or individual.
PEO-2	Higher studies	Prepared to pursue masters or research programmes in computer science or other disciplines.
PEO-3	Entrepreneurship	Become successful Entrepreneurs who demonstrate strong technical and leadership skills to bring out innovative designs/products that also addresses social issues.
PEO-4	Lifelong learning and ethics	Adapt to rapidly changing technology in engineering domains through continuous learning and practice code of ethics.

PROGRAM SPECIFIC OUTCOMES (PSOs)

1	Programming and software Development skills: Ability to acquire programming efficiency to analyze, design and develop optimal solutions, apply standard practices in software project development to deliver quality software product.
2	Computer Science Specific Skills: Ability to formulate, simulate and use knowledge in various domains like data engineering, image processing and information and network security, artificial intelligence etc., and provide solutions to new ideas and innovations

PROGRAM OUTCOMES (POs)

Graduate Attribute1:	Engineering Knowledge
PO-1	Apply the knowledge of basic engineering sciences, humanities, core engineering and computing concept in modeling and designing computer based systems.
Graduate Attribute2:	Problem Analysis
PO-2	Identify, analyze the problems in different domains and define the requirements appropriate to the solution.
Graduate Attribute3:	Design/Development of Solution
PO-3	Design, implement & test a computer based system, component or process that meet functional constraints such as public health and safety, cultural, societal and environmental considerations.
Graduate Attribute4:	Conduct Investigations of Complex Problems
PO-4	Apply computing knowledge to conduct experiments and solve complex problems, to analyze and interpret the results obtained within specified timeframe and financial constraints consistently.
Graduate Attribute5:	Modern Tool Usage
PO-5	Apply or create modern techniques and tools to solve engineering problems that demonstrate cognition of limitations involved in design choices.
Graduate Attribute6:	The Engineer and Society
PO-6	Apply contextual reason and assess the local and global impact of professional engineering practices on individuals, organizations and society.
Graduate Attribute7:	Environment and Sustainability
PO-7	Assess the impact of engineering practices on societal and environmental sustainability.
Graduate Attribute8:	Ethics
PO-8	Apply professional ethical practices and transform into good responsible citizens with social concern.
Graduate Attribute9:	Individual and Team Work
PO-9	Acquire capacity to understand and solve problems pertaining to various fields of engineering and be able to function

	effectively as an individual and as a member or leader in a team.
Graduate Attribute10:	Communication
PO-10	Communicate effectively with range of audiences in both oral and written forms through technical papers, seminars, presentations, assignments, project reports etc.
Graduate Attribute11:	Project Management and Finance
PO-11	Apply the knowledge of engineering, management and financial principles to develop and critically assess projects and their outcomes in multidisciplinary areas.
Graduate Attribute12:	Life-long Learning
PO-12	Recognize the need and prepare oneself for lifelong self learning to be abreast with rapidly changing technology.

ANITS-CSE CURRICULUM – REGULATIONS –R19

I Year Course structure – CSE

Semester - I

Course Code	Title of the course	Category	Periods						Sessionals Marks	Semester end Exam	Total Marks	Credits
			L	T	P	E	O	Total				
CSE111	EngineeringMathematics – I	BS	3	0	0	1	6	10	40	60	100	3
CSE112	Communicative English	HS	3	0	0	1	3	7	40	60	100	3
CSE113	Basic Electronics Engineering	ES	3	0	0	1	3	7	40	60	100	3
CSE114	Digital Logic Design	ES	3	0	0	1	3	7	40	60	100	3
CSE115	PROBLEM SOLVING WITH C	ES	3	0	0	1	3	7	40	60	100	3
CSE116	English Language Lab	HS	0	0	3	0	1	4	50	50	100	1.5
CSE117	Problem solving with C – lab.	ES	0	0	3	0	3	6	50	50	100	1.5
CSE118	Environmental Science (Mandatory non-credit course)	BS	3	0	0	0	1	4	50	-	50	-
Total			**18**	**0**	**6**	**5**	**23**	**52**	**350**	**400**	**750**	**18**

I Year Course structure – CSE

Semester - II

Course Code	Title of the course	Category	Periods						Sessionals Marks	Semester end Exam	Total Marks	Credits
			L	T	P	E	O	Total				
CSE121	Engineering Mathematics – II	BS	3	0	0	1	6	10	40	60	100	3
CSE122	Engineering Physics	BS	3	0	0	1	4	8	40	60	100	3
CSE123	Engineering Chemistry	BS	3	0	0	1	4	8	40	60	100	3
CSE124	ELEMENTS OF ELECTRICAL ENGINEERING	ES	3	0	0	1	4	8	40	60	100	3
CSE125	Engineering Drawing	ES	2	0	3	1	3	9	40	60	100	3.5
CSE126	Engineering Physics Lab.	BS	0	0	3	0	1	4	50	50	100	1.5
CSE127	Engineering Chemistry Lab.	BS	0	0	3	0	1	4	50	50	100	1.5
CSE128	Engineering Workshop	ES	0	0	3	0	1	4	50	50	100	1.5
CSE129	Human Values and Professional Ethics (Mandatory non-credit course)	HS	3	0	0	0	1	4	50	0	50	-

Total	**17**	**0**	**12**	**5**	**25**	**59**	**400**	**450**	**850**	**20**

II Year Course structure – CSE

Semester - I

CODE	SUBJECT NAME	Category	Periods						Sessionals Marks	Semester end Exam	Total Marks	Credits
			L	T	P	E	O	Total				
CSE 211	DATA STRUCTURES&ALGORITHMS	PC	2	1	0	1	4	8	40	60	100	3
CSE 212	COMPUTER ORGANIZATION	PC	3	0	0	1	4	8	40	60	100	3
CSE 213	JAVA PROGRAMMING	PC	3	0	0	1	4	8	40	60	100	3
CSE 214	DATA COMMUNICATION	PC	3	0	0	1	4	8	40	60	100	3
CSE 215	DISCRETE MATHEMATICAL STRUCTURES	BS	3	0	0	1	4	8	40	60	100	3
CSE 216	DESIGN THINKING & PRODUCT INNOVATION	ES	2	0	2	1	3	8	40	60	100	3
CSE 217	***JAVA PROGRAMMING LAB***	***PC***	0	0	***3***	0	2	5	***50***	***50***	***100***	1.5
CSE 218	***DATA STRUCTURES LAB USING C***	***PC***	0	0	***3***	0	2	5	***50***	***50***	***100***	1.5
Total			**16**	**1**	**8**	**6**	**27**	**58**	**340**	**460**	**800**	**21**

II Year Course structure – CSE

Semester - II

CODE	SUBJECT NAME	Category	Periods						Sessionals Marks	Semester end Exam	Total Marks	Credits
			L	T	P	E	O	Total				
CSE 221	PROBABILITY , STATISTICS AND QUEUEING THEORY	BS	3	0	0	1	6	10	40	60	100	3
CSE 222	MICROPROCESSOR & INTERFACING	PC	2	1	0	2	4	9	40	60	100	3
CSE 223	OPERATING SYSTEMS	PC	3	0	0	1	4	8	40	60	100	3
CSE 224	COMPUTER NETWORKS	PC	3	0	0	1	4	8	40	60	100	3
CSE 225	COMPUTER GRAPHICS	PC	2	1	0	1	4	8	40	60	100	3
CSE 226	FORMAL LANGUAGES AND AUTOMETA THEORY	PC	2	1	0	1	2	6	40	60	100	3
CSE 227	***MICRO PROCESSOR INTERFACING LAB***	***PC***	0	0	***3***	***0***	***1***	4	***50***	***50***	***100***	1.5
CSE 228	***OPERATING SYSTEM LAB***	***PC***	0	0	***3***	***0***	***1***	4	***50***	***50***	***100***	1.5
Total			**15**	**3**	**6**	**7**	**26**	**57**	**340**	**460**	**800**	**21**

III Year Course structure – CSE

Semester - I

CODE	SUBJECT NAME	Category	Periods						Sessionals Marks	Semester end Exam	Total Marks	Credits
			L	T	P	E	O	Total				
CSE 311	OPEN ELECTIVE-I*	OE	3	0	0	1	2	6	40	60	100	3
CSE 312	PROFESSIONAL ELECTIVE -I	PE	3	0	0	1	2	6	40	60	100	3
CSE 313	COMPETITIVE PROGRAMMING	PC	2	1	0	1	5	9	40	60	100	3
CSE 314	COMPILER DESIGN	PC	2	1	0	1	5	9	40	60	100	3
CSE 315	DATA BASE MANAGEMENT SYSTEMS	PC	3	0	0	1	4	8	40	60	100	3
CSE 316	DESIGN & ANALYSIS OF ALGORITHMS	***PC***	2	1	0	1	4	8	40	60	100	3
CSE 317	***QA-I &SOFT SKILLS***	***HS***	0	0	3	***0***	1	4	***100***	***0***	100	1.5
CSE 318	***COMPETITIVE PROGRAMMING LAB***	***PC***	0	0	***3***	***0***	1	4	***50***	***50***	100	1.5
CSE 319	***DATA BASE MANAGEMENT SYSTEMS LAB***	***PC***	0	0	***3***	1	3	7	***50***	***50***	100	1.5
CSE 320	Constitution of India & - Intellectual Property Rights (Mandatory non-credit course)	HS	2	0	0	0	1	3	50	0	50	-
Total			**17**	**3**	**9**	**7**	**28**	**64**	**490**	**460**	**950**	**22.5**

III Year Course structure – CSE

Semester - II

CODE	SUBJECT NAME	Category	Periods						Sessionals Marks	Semester end Exam	Total Marks	Credits
			L	T	P	E	O	Total				
CSE 321	OPEN ELECTIVE -II*	OE	3	0	0	1	2	6	40	60	100	3
CSE 322	PROFESSIONAL ELECTIVE -II	PE	3	0	0	1	2	6	40	60	100	3
CSE 323	PROFESSIONAL ELECTIVE -III	PE	3	0	0	1	2	6	40	60	100	3
CSE 324	OBJECT ORIENTED SOFTWARE ENGINEERING	PC	3	0	0	1	4	8	40	60	100	3
CSE 325	WEB TECHNOLOGIES	PC	2	1	0	1	4	8	40	60	100	3
CSE 326	CRYPTOGRAPHY AND NETWORK SECURITY	PC	3	0	0	1	4	8	40	60	100	3
CSE 327	***QA-II&VERBAL ABILITY***	HS	0	0	3	***0***	***1***	4	***100***	***0***	100	1.5
CSE 328	***OBJECT ORIENTED SOFTWARE ENGINEERING LAB***	***PC***	0	0	***3***	***0***	***1***	4	***50***	***50***	100	1.5

CSE 329	***WEB TECHNOLOGIES LAB***	***PC***	0	0	***3***	2	3	8	***50***	***50***	100	1.5
Total			**17**	**1**	**9**	**8**	**23**	**58**	**440**	**460**	**900**	**22.5**

IV Year Course structure – CSE (Tentative)

Semester - I

CODE	SUBJECT NAME	Category	Periods						Sessionals Marks	Semester end Exam	Total Marks	Credits
			L	T	P	E	O	Total				
CSE 411	OPEN ELECTIVE -III*	OE	3	0	0	1	2	6	40	60	100	3
CSE 412	PROFESSIONAL ELECTIVE -IV	PE	3	0	0	1	2	6	40	60	100	3
CSE 413	PROFESSIONAL ELECTIVE -V	PE	3	0	0	1	2	6	40	60	100	3
CSE 414	MANAGEMENT PRINCIPLES	HS	3	0	0	0	2	5	40	60	100	3
CSE 415	DATA ANALYTICS	PC	2	1	0	1	4	8	40	60	100	3
CSE 416	***CRYPTOGRAPHY & NETWORK SECURITY LAB***	***PC***	0	0	***3***	***0***	***1***	4	***50***	***50***	***100***	1.5
CSE 417	***DATA ANALYTICS LAB***	***PC***	0	0	***3***	***0***	***1***	4	***50***	***50***	***100***	1.5
CSE 418	***PROJECT PHASE-I***	***PR***	0	0	***3***	***0***	***1***	4	***100***	***0***	***100***	2
CSE 419	***SUMMER INTERNSHIP-INDUSTRY***	***PR***	0	0	***0***	***0***	***1***	1	***100***	***0***	***100***	1
Total			**14**	**1**	**9**	**4**	**16**	**44**	**500**	**400**	**900**	**21**

IV Year Course structure – CSE

Semester - II

CODE	SUBJECT NAME	Category	Periods						Sessionals Marks	Semester end Exam	Total Marks	Credits
			L	T	P	E	O	Total				
CSE 421	OPEN ELECTIVE -IV*	OE	3	0	0	1	3	7	40	60	100	3
CSE 422	PROFESSIONAL ELECTIVE -VI/MOOC	PE	3	0	0	1	3	7	40	60	100	3
CSE 423	***PROJECT PHASE-II***	***PR***	0	0	9	0	9	18	***100***	***100***	***200***	8
Total			**6**	**0**	**9**	**2**	**15**	**32**	**180**	**220**	**400**	**14**

Total Credits **160**

*Open Elective can be Inter Department Disciplinary Course,Emerging Courses or MOOC. Final decision will be taken by the department.

List of Professional Electives	
PE1 3rd Yr-Sem -1	•CSE 312(A)Embedded Systems •CSE312(B)Advanced Data Structures •CSE312(C)Digital ImageProcessing .CSE 312(D)Artificial Intelligence
PE2 3rd Yr-Sem -2	•CSE322(A)Human Computer Interaction •CSE 322(B) Mobile Computing .CSE322(C)No SQL Data Bases .CSE322(D)Data warehousing and Data mining
PE3 3rd Yr-Sem -2	• CSE323(A)Distributed Operating Systems • CSE323(B)Smart Systems Design & Programming • CSE323(C)Machine Learning Using R • CSE323(D)Computer vision
PE4 4th Yr-Sem -1	.CSE412(A)Natural Language Processing • CSE412(B)Bioinformatics .CSE 412(C)High Performance Computing • CSE415(D)Neural Networks
PE5 4th Yr-Sem -1	.CSE413(A)IOT • CSE413(B)AugmentedReality • CSE413(C)Semantic Web • CSE413(D)Multimedia & Animation
PE6 4th Yr-Sem -2	.CSE422(A) Deep Learning • CSE422(B)Cyber Security • CSE422(C)Social Network Analysis • CSE422(D) Cloud Computing

OPEN ELECTIVES	
OE 1 3rd Yr-Sem -1	CSE 311 Essentials of Python (as emerging subject) Python Programming Basics
OE2 3rd Yr-Sem -2	File Systems & Data Bases Computer Operating Systems Fundamentals Of Computer Networks
OE3 4th Yr-Sem -1	will be decided next year
OE4 4th Yr-Sem -2	will be decide next year

ESSENTIALS OF PYTHON (Open elective for CSE)	
Code: CSE 311	Credits : 3
Instruction :3 Periods/Week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:

- Basic Knowledge of Programming Fundamentals

Course Objectives:

- Describe the core syntax and semantics of Python programming language.
- Illustrate the process of structuring the data using lists, dictionaries, tuples, strings and sets.
- Discover the need for working with the functions, modules and packages.
- Infer the Object-oriented Programming concepts in Python.
- Familiarize the advanced concepts like regular expressions, date and time. Indicate the use of and built-in functions to navigate the file system.

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Interpret the fundamental Python syntax and semantics and able to solve, test and debug python programs
2.	Fluency in the use of Python control flow statements and Determine the methods to create and manipulate Python programs by utilizing the data structures like lists, dictionaries, tuples, strings and sets.
3.	Express proficiency in the handling of functions, modules, packages and handle abnormal termination of the programs.
4.	Articulate the Object-Oriented Programming concepts such as encapsulation, inheritance and polymorphism as used in Python
5.	List the usage and application of regular expressions, date and time. Identify the commonly used operations involving file systems.

Mapping of Course Outcomes with Program Outcomes:

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

SYLLABUS

UNIT-I: 10 periods

Introduction: Installation, Keywords and Identifiers, Statement, Indentation, Comments, Variables, Constants, Literals, Data Types, Type Conversion, I/O, Import, Operators(Arithmetic operators, Comparison operators, Logical operators, Bitwise operators, Assignment operators, Identity operators, Membership operators), Namespace and Scope.

Learning Outcomes: At the end of this Unit the student will be able to

- Analyze fundamental advantages of python over the other programming languages.
- Solve, test and debug basic problems using python script.

UNIT-II 14 periods

Flow control & Collections: If, If...else, if...elif...else, Nested if, for loop, while loop, Break, Continue and Pass. Numbers, Decimal, Fractions, Mathematics, List, Tuple, String, Set and Dictionary. Data types manipulations (create, Index, Negative indexing, Slicing, change or add elements, delete or remove elements, Methods, Comprehension, Membership Test, Iteration, Operations and Built in Functions)

Learning Outcomes: At the end of this Unit the student will be able to

- Implement Flow control statements required real world problems.
- Manipulate python programs by using the python data structures like lists, dictionaries, tuples, strings and sets.

UNIT-III: 12 periods

Functions: Function, Function argument, Recursion, Anonymous / Lambda functions, Global, Local and Nonlocal variables, Global keyword, Modules and Packages. Exception Handling in Python, What is an Exception?, Syntax for Exception Handling, Handling
Single Exception, Handling Multiple Exceptions.

Learning Outcomes: At the end of this Unit the student will be able to

- Resolve real world problems using python functions.
- Familiarize the usage of Modules and packages to enhance the problem solving and usage of Exceptions.

UNIT-IV: 12 periods

Object oriented programming: Introduction to OOPs, Class, Object, Constructors, Methods, Inheritance, Method Overriding, Multiple Inheritance, Operator overloading, Encapsulation and Polymorphism.

Learning Outcomes: At the end of this Unit the student will be able to

- Design object-oriented programs with Python classes.
- Usage of inheritance, encapsulation, inheritance and polymorphism for reusability.

UNIT-V: 12 periods

Advanced topics:The regex package, Regular expression methods: findall, finditer, match, search, split, sub, subn.Date and Time Data Types and Tools: The datetime package, Types in datetime module,Datetime format specification, Locale-specific date formatting, Time Series

Basics, Date Ranges,Frequencies, and Shifting, Base Time Series Frequencies, Time Zone Handling.Files(Open, Read, Write, Close) and File Methods.

Learning Outcomes: At the end of this Unit the student will be able to

- Interpret the advantages of advanced concepts like regular expressions, date and time.
- Identify the commonly used operation involved in files for I/O processing.

TEXT BOOKS

1. Wes McKinney, "Python for Data Analysis",O'REILLY,2012.
2. Vamsi Kurama, Pearson, "Python Programming : A Modern Approach", Pearson India, 2017

REFERENCE BOOKS

1. Downey, Allen. How to think like a computer scientist : learning with Python / Allen Downey, Jeffrey Elkner, Chris. Meyers. – 1st ed.2002
2. Gowrishankar S, Veena, "Introduction to Python Programming", CRC Press/Taylor & Francis, 2019.
3. John Hunt "A Beginners Guide to Python 3 Programming", Springer

CASE STUDIES

1. Jack and his three friends have decided to go for a trip by sharing the expenses of the fuel equally. Write a Python program to calculate the amount (in Rs) each of them need to put in for the complete (both to and fro) journey.
The program should also display True, if the amount to be paid by each person is divisible by 5, otherwise it should display False. (Hint: Use the relational operators in the print statement.)
Assume that mileage of the vehicle, amount per litre of fuel and distance for one way are given.

Test your code by using the given sample inputs.
Verify your code by using the 2nd sample input(highlighted) given below:

Sample Input			Expected Output
Mileage of the vehicle (km/litre of fuel)	Amount per litre of fuel (Rs)	Distance for one way (kms)	
12	65	96	260.0 True
12	40	190	

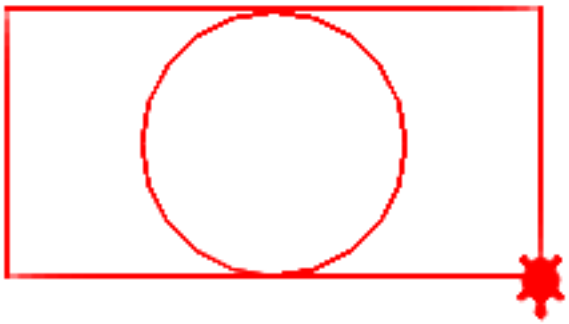
2. A three digit number is said to be an "Armstrong number" if the sum of the third power of its individual digits is equal to the number itself.
Example: 371 is an Armstrong number as 371 = 33 + 73 + 13
407 is an Armstrong number as 407 = 43 + 03 + 73
Write a pseudo-code to check whether a given three digit number is an Armstrong number.

3.A University offering degree courses to students has decided to provide scholarship based on the following details:

Branch of study	Score (%)	Scholarship %	Remarks
Arts	Score is at least 90	50	The student is eligible only for one scholarship% even if both the score conditions are valid for the given branch of study. In such cases, students are eligible for the highest scholarship% applicable among the two.
Arts	Score is an odd number	5	
Engineering	Score is more than 85	50	
Engineering	Score is divisible by 7	5	

If there are 500 students who have joined the university, write a pseudocode to calculate and display the final fees to be paid by each student. You may accept the branch of study, score and course fee as inputs for each student and calculate the final fees to be paid by each student based on formulae given below:
Scholarship amount=course fee * (scholarship%)
Final fee= course fee - scholarship amount

4. Write a program to create the following pattern:



5. Write a python program to find and display the product of three positive integer values based on the rule mentioned below: It should display the product of the three values except when one of the integer values is 7. In that case, 7 should not be included in the product and the values to its left also should not be included. If there is only one value to be considered, display that value itself. If no values can be included in the product, display -1.

Note: Assume that if 7 is one of the positive integer values, then it will occur only once. Refer to the sample I/O given below.

Sample Input	Expected Output
1, 5, 3	15
3, 7, 8	8
7, 4, 3	12
1, 5, 7	-1

6.A traveller on a visit to India is in need of some Indian Rupees (INR) but he has money belonging to another currency. He wants to know how much money he should provide in the currency he has, to get the specified amount in INR.

Write a python program to implement a currency calculator which accepts the amount needed in INR and the name of the currency which the traveller has. The program should identify and display the amount the traveller should provide in the currency he has, to get the specified amount in INR.

Note: Use the forex information provided in the table below for the calculation. Consider that only the currency names mentioned in the table are valid. For any invalid currency name, display -1.

Currency	Equivalent of 1.00 INR
Euro	0.01417
British Pound	0.0100
Australian Dollar	0.02140
Canadian Dollar	0.02027

7. Write a python program to generate and display the next date of a given date.

Assume that
Date is provided as day, month and year as shown in below table.
The input provided is always valid. Output should be day-month-year.
Hint: print(day,"-",month,"-",year) will display day-month-year

	Sample Input	Expected Output
Day	1	2-9-2020
Month	9	
Year	2020	

8. Write a python program which finds the maximum number from num1 to num2 (num2 inclusive) based on the following rules.

1.Always num1 should be less than num2
2.Consider each number from num1 to num2 (num2 inclusive). Populate the number into a list, if the below conditions are satisfied
a. Sum of the digits of the number is a multiple of 3
b. Number has only two digits
c. Number is a multiple of 5

3.Display the maximum element from the list
In case of any invalid data or if the list is empty, display -1.

9. Given a string containing uppercase characters (A-Z), compress the string using Run Length encoding. Repetition of character has to be replaced by storing the length of that run.

Write a python function which performs the run length encoding for a given String and returns the run length encoded String.

Provide different String values and test your program.

Sample Input	Expected Output
AAAABBBBCCCCCCCC	4A4B8C
AABCCA	2A1B2C1A

10. A hospital wants to know the medical speciality visited by the maximum number of patients.
Assume that the patient id of the patient along with the medical speciality visited by the patient is stored in a list. The details of the medical specialities are stored in a dictionary as follows:
{
"P":"Pediatrics",
"O":"Orthopedics",
"E":"ENT
}

Write a function to find the medical speciality visited by the maximum number of patients and return the name of the speciality.

Note:

1. Assume that there is always only one medical speciality which is visited by a maximum number of patients.
2. Perform case sensitive string comparison wherever necessary.

Sample Input	Expected Output
[101,P,102,O,302,P,305,P]	Pediatrics
[101,O,102,O,302,P,305,E,401,O,656,O]	Orthopedics
[101,O,102,E,302,P,305,P,401,E,656,O,987,E]	ENT

11. Write a python program to display all the common characters between two strings. Return -1 if there are no matching characters.
Note: Ignore blank spaces if there are any. Perform case sensitive string comparison wherever necessary.

Sample Input	Expected output
"I like Python" "Java is a very popular language"	lieyon

12.A teacher is in the process of generating a few reports based on the marks scored by the students of her class in a project based assessment.
Assume that the marks of her 10 students are available in a tuple. The marks are out of 25.

Write a python program to implement the following functions:
1.find_more_than_average(): Find and return the percentage of students who have scored more than the average mark of the class.
2. generate_frequency(): Find how many students have scored the same marks. For example, how many have scored 0, how many have scored 1, how many have scored 3….how many have scored 25. The result should be populated in a list and returned.
3. sort_marks(): Sort the marks in the increasing order from 0 to 25. The sorted values should be populated in a list and returned.

Sample Input	Expected Output
list_of_marks = (12,18,25,24,2,5,18,20,20,21)	70.0 [0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 2, 0, 2, 1, 0, 0, 1, 1] [2, 5, 12, 18, 18, 20, 20, 21, 24, 25]

13. Write a python function, check_double(number) which accepts a whole number and returns True if it satisfies the given conditions.

1. The number and its double should have exactly the same number of digits.
2. Both the numbers should have the same digits ,but in different order.
   Otherwise it should return False.

Example: If the number is 125874 and its double, 251748, contain exactly the same digits, but in a different order.

14. Given a number n, write a program to find the sum of the largest prime factors of each of nine consecutive numbers starting from n.
g(n) = f(n) + f(n+1) + f(n+2) + f(n+3) + f(n+4) + f(n+5) + f(n+6) + f(n+7) + f(n+8)
where, g(n) is the sum and f(n) is the largest prime factor of n

For example,
g(10)=f(10)+f(11)+f(12)+f(13)+f(14)+f(15)+f(16)+f(17)+f(18)
=5 + 11 + 3 + 13 + 7 + 5 + 2 + 17 + 3
=66

15. Write a python function, nearest_palindrome() which accepts a number and returns the nearest palindrome greater than the given number.

Sample Input	Expected Output
12300	12321
12331	12421

16. Assume that a poem is given. Write the regular expressions for the following:

1. Print how many times the letter 'v' appears in the poem.
2. Remove all the newlines from the poem and print the poem in a single line.
3. If a word has 'ch' or 'co', replace it with 'Ch' or 'Co'.
4. If the pattern has characters 'ai' or 'hi', replace the next three characters with **.

Test your code by using the given sample inputs.
Verify your code by using the 2nd sample input(highlighted) given below:

Sample Input	Expected Output
If I can stop one heart from breaking, I shall not live in vain; If I can ease one life the aching, Or cool one pain, Or help one fainting robin Unto his nest again, I shall not live in vain.	4 If I can stop one heart from breaking, I shall not live in vain; If I can ease one life the aching, Or cool one pain, Or help one fainting robin Unto his nest again, I shall not live in vain. If I can stop one heart from breaking, I shall not live in vain; If I can ease one life the aChing, Or Cool one pain, Or help one fainting robin Unto his nest again, I shall not live in vain. If I can stop one heart from breaking, I shall not live in vain; If I can ease one life the achi** Or cool one pain, Or help one fai**ng robin Unto hi**est again, I shall not live in vain.
It takes strength for being certain, It takes courage to have doubt. It takes strength for challenging alone, It takes courage to lean on another. It takes strength for loving other souls, It takes courage to be loved. It takes strength for hiding our own pain, It takes courage to help if it is painful for	

someone.	

17.A university wants to automate their admission process. Students are admitted based on marks scored in a qualifying exam.
A student is identified by student id, age and marks in the qualifying exam. Data are valid, if:

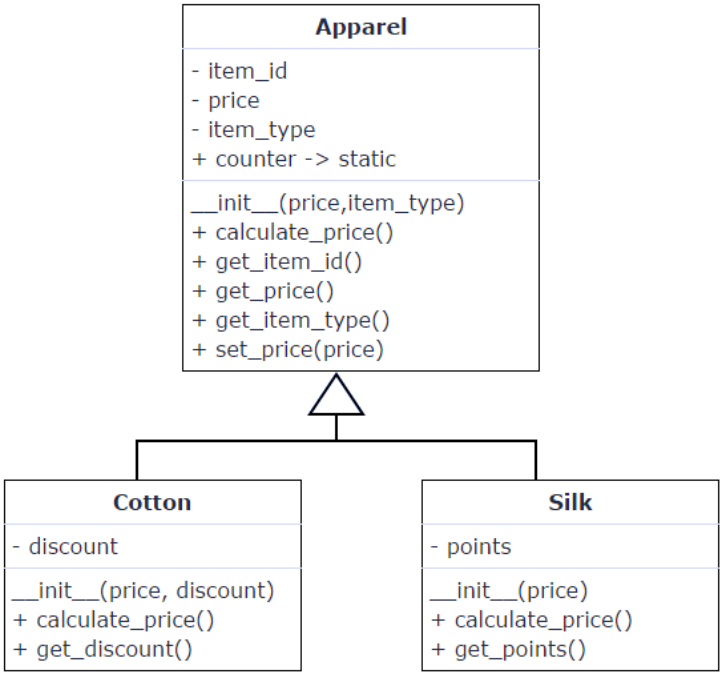
- Age is greater than 20
- Marks is between 0 and 100 (both inclusive)

A student qualifies for admission, if

- Age and marks are valid and
- Marks is 65 or more

Write a python program to represent the students seeking admission in the university.

18. An apparel shop wants to manage the items which it sells. Write a python program to implement the class diagram given below.



19. Write a python program to Find Resolution of JPEG Image
20. Royal Orchid is a florist. They want to be alerted when stock of a flower goes below a particular level.
The flowers are identified using name, price per kg and stock available (in kgs).
Write a Python program to implement the above requirement.

EMBEDDED SYSTEMS	
CSE312(A)	Credits : 3
Instruction : 3 Periods /Week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:

- Basic Knowledge of Programming Fundamentals
- Knowledge of Programming Languages (such as C, C++)

Course Objectives:

- To provide in-depth knowledge about embedded processor, its hardware and software
- To explain programming concepts and embedded programming in C and assembly language
- To explain real time operating systems, inter-task communication and an embedded software development tool

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Identify the basic components required to build an embedded system.
2.	Select an appropriate software architecture to build an embedded system..
3.	Design embedded software-using RTOS.
4.	Build embedded software using different software tools.
5.	Debug embedded software using different software and hardware tools.

Mapping of Course Outcomes with Program Outcomes:

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

SYLLABUS

UNIT I **(8 Hours)**

Introduction to embedded systems hardware needs; timing diagrams, memories (RAM, ROM, EPROM). Tristate devices, Buses, PLD's. Built-ins on the microprocessor. Interrupts basics, ISR;Context saving, shared data problem. . Atomic and critical section.

Learning Outcomes: At the end of this unit, Students are able to

- Describe different types of memories used in Embedded System Development
- Interpret the role of Interrupts in Embedded Systems Performance.

UNIT II **(8 Hours)**

Survey of software architectures, Round Robin, Function queue scheduling architecture, Use of real time operating system.RTOS, Tasks , Scheduler, Shared data reentrancy, priority inversion, mutex binary semaphore and counting semaphore.

Learning Outcomes: At the end of this unit, Students are able to

- Describe the role of software architecture in Embedded Systems Performance
- Explain the application of the Real Time Operating System in Embedded System Development.

UNIT III **(10 Hours)**

Inter task communication, message queue, mailboxes and pipes, timer functions, events. Interrupt routines in an RTOS environment.Embedded system software design using an RTOS. Hard realtime and soft real time system principles, Task division, need of interrupt routines, Interrupt latency, Introduction to Device Drivers.

Learning Outcomes: At the end of this unit, Students are able to

- Describe the different RTOS features used in Embedded Systems Development.
- Differentiate Hard real time and Soft real time systems

UNIT IV **(9 Hours)**

Embedded Software development tools. Host and target systems, cross compilers, linkers, locators for embedded systems. Getting embedded software into the target system.

Learning Outcomes: At the end of this unit, Students are able to

- Differentiate host and target systems
- Describe the need of Cross Compilers,linkers and locators for embedded system development.

UNIT V **(10 Hours)**

Debugging techniques. Testing on host machine ,Instruction set emulators, logic analysers. In-circuit emulators and monitors.

Case Study

DEVELOPING EMBEDDED C APPLICATIONS THROUGH KEIL SOFTWARE, Embedded PROGRAMMING IN c++, java.

Learning Outcomes: At the end of this unit, Students are able to

- Explain different types of debugging techniques used in Embedded System Development.
- Describe the difference between simulation and emulation.

Text Books:

1. David A. Simon, An Embedded Software Primer, Pearson Education, Inc., 1999
2. Raj Kamal, Embedded Systems, Architecture, Programming and Design, TMH, 2003

Reference Books:

1. Sriram V Iyer and Pankaj Gupta, Embedded Real Time Systems programming, TMH, 2004.

Web Resources:

1. https://www.coursera.org/learn/introduction-embedded-systems#about

ADVANCED DATA STRUCTURES	
CSE 312(B)	**Credits: 3**
Instruction : 3 Periods / Week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:

- Knowledge of Data structures.
- Students must have the knowledge of some programming languages (such as C, C++, Java).

Course Objectives:

- Understand the variety of advanced data structures (skip lists, hash tables, priority queues, balanced search trees, graphs).
- Give the advantages and dis-advantages of each of the advanced data structure.
- Learn how to apply algorithm design techniques and data structures to solve problems.
- Learn different external sorting techniques and analyze their efficiency.

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Implement the ADTs like linear list, skip list and hash tables and their operations
2.	Describe the methods to how to balance a binary search trees using Rotation methods and Color changing methods
3.	Design and apply binary heaps, leftist heaps, Binomial queues and their operations for solving the real world scenarios.
4.	Apply Algorithms for solving problems by using sorting techniques
5.	Solve problems using data structures such as graph algorithms, including single-source and all-pairs shortest paths.

Mapping of course outcomes with program outcomes:

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

SYLLABUS

UNIT-I : **12 Hours**

Skip lists and Hashing:
Sets, Map**,** Dictionaries, representation of dictionary as ADT,Linear list, skip list, hash table representation, an application-text compression using dictionary.
Learning Outcomes: At the end of this unit, the students will be able to

- Explain ADT, sets, maps and dictionaries.
- Implement lists and hashing techniques.

UNIT-II : **15 Hours**

Balanced Search Trees:
Red-black trees, Representation of Red-black tree, Insertion, Deletion and searching of nodes in Red-black tree. Splay trees, B-Trees, Indexed Sequential Access Method(ISAM), B-Trees of order m, Representation of B-Tree, Insertion, deletion and searching a node in B-Tree.
Learning Outcomes: At the end of this unit, the students will be able to

- Explore balanced search trees.
- Implement basic operations on the balanced search trees.

UNIT-III : **12 Hours**

Priority Queues:
Binary heap, Applications of priority queues, leftist heaps, Binomial queues.
Learning Outcomes: At the end of this unit, the students will be able to

- Explain the operations of the priority queue.
- Implement Priority queue operations.

UNIT-IV: **12 Hours**

Sorting:
Shell sort, Heap sort, Quick sort, Indirect sorting, decision trees, bucket sort, External sorting.
Learning Outcomes: At the end of this unit, the students will be able to

- Describe the concept of different sorting techniques.
- Apply different sorting algorithms.

UNIT-V : **12 Hours**

Graphs:
Graph algorithms-Topological sorting, shortest-path algorithms- unweighted shortest path, graphs with negative edge cost, acyclic graphs, Network flow problems, Applications of DFS. Introduction to NP-Completeness.
Learning Outcomes: At the end of this unit ,the students will be able to

- Construct the various types of graphs.
- Apply the graph concepts to solve the real time problems.

Text Books:

1.SartajSahni,"*Data Structures, Algorithms and Applications in C++",* SecondEdition,University Press

2.Mark Allen Weiss, "Data Structures and Algorithm Analysis in C++" , Third Edition, Pearson Education.

Reference Books:

1. Richard F.Gilberg, Behrouz A.Forouzan, "*Data Structures: A Pseudocode Approach with C",* Second Edition, Cengage Learning.
2. NB Venkateswarulu and EV Prasad, "*C and Data structures: A Snap Shot Oriented Treatise with Live Examples from Science and Engineering",* S Chand, 2010.

Web Resources:

1. http://nptel.ac.in/courses/106102064/
2. http://nptel.ac.in/courses/106103069/

DIGITAL IMAGE PROCESSING	
CSE 312(C)	**Credits : 3**
Instruction : 3Hours /Week	Sessional Marks : 40
End Exam : 3 Hours	Ena Exam Marks : 60

Prerequisites:

- Knowledge of linear algebra, basic probability and statistics, introductory knowledge of basic programming language, MATLAB/C are preferred.

Course Objectives:

- To make the students to be familiar with basic image processing techniques for solving real problems,
- To make the students have a general overview on digital image processing concept along with its uses and applications.
- To make the students gain knowledge about representation of a digital image in different domains and the transformations between those domains,
- To make the students learn about various morphological operations on a digital image.

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Apply the basic concepts of 2D image acquisition, sampling, quantization, relationships between pixels and components of image.
2.	Analyze the filtering techniques in spatial domain for face reorganization, pattern reorganization and segmentation.
3.	Analyze and apply the filtering techniques in frequency domain for classify the images.
4.	Apply image morphological techniques for manipulating digital images.
5.	Apply the image Segmentation techniques on Edge detection and Region-Based Segmentation.

Mapping of course outcomes with program outcomes:

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

SYLLABUS

UNIT-I : **12 Hours**

Introduction :
Digital Image Processing and Applications – Image Representation and Modeling
Digital Image Fundamentals:
Elements of Visual perception – A simple Image Model – Sampling and Quantization – Some Basic Relationship between Pixels.
Elements of digital image processing systems, Digital Camera working principles, Elements of visual perception, brightness, contrast, hue, saturation.
Learning Outcomes:

- Review the fundamental concepts of a digital image processing system.
- Describe and explain basic principles of digital image processing

UNIT-II : **15 Hours**

Image Transforms & **Color Image Processing:**
Background, Some Basic Intensity Transformation Functions, Histogram Processing, Histogram Equalization, Histogram Matching (Specification), Local Histogram Processing, Using Histogram Statistics for Image Enhancement, Color Fundamentals , Color Models , The RGB Color Model , The CMYand CMYK Color Models , The HSI Color Model , Pseudocolor Image Processing , Intensity Slicing , Intensity to Color Transformations , Basics of Full-Color Image Processing.
Learning Outcomes:

- Examine various types of images, intensity transformations and color images.

UNIT- III : **8 Hours**

IMAGE ENHANCEMENT IN SPATIAL DOMAIN:
Fundamentals of Spatial Filtering, Smoothing Spatial Filters, Sharpening Spatial Filters, Combining Spatial Enhancement Methods.
Learning Outcomes:

- Analyze images in the frequency domain using various transforms.
- Develop Fourier transform for image processing in frequency domain.

UNIT-IV : **12 Hours**

IMAGE ENHANCEMENT IN FREQUENCY DOMAIN:
Background, Preliminary Concepts, Sampling and the Fourier Transform of Sampled Functions, The Basics of Filtering in the Frequency Domain, Image Smoothing Using Frequency Domain Filters, Image Sharpening Using Frequency Domain Filters, Selective Filtering.
Learning Outcomes:

- Evaluate the techniques for image enhancement and image restoration.
- Apply image processing algorithms in practical applications.

UNIT-V : **15 Hours**

IMAGE SEGMENTATION & Morphology:
Edge detection, Edge linking via Hough transform – Thresholding - Region based segmentation – Region Growing – Region splitting and Merging .
Preliminaries, Erosion and Dilation, Opening and Closing, the Hit-or-Miss Transformation, Some Basic Morphological Algorithms, Gray-Scale Morphology

Learning Outcomes:

- Interpret image segmentation and representation techniques.
- Evaluate the methodologies for image segmentation, restoration etc.

Text Books:

1. Gonzalez Rafael C and Woods Richard E," *Digital Image Processing*", 3rd Edition, Prentice Hall, 2008.
2. Jain Anil K," *Fundamentals of Digital Image Processing"*, PrenticeHall, 1989. (TA1632.J25)

Reference Books:

1. Pratt William K, "*Digital Image Processing: PIKS Scientific Inside"*, 4th Edition, John Wiley, 2007. (TA1632.P917 2007)
2. Pitas Ioannis, Digital Image Processing Algorithms and Applications, John Wiley, 2000. (TA1637.P681)
3. Anil K. Jain, PHI. Pattern Recognition and Image Analysis, Earl Gose and Richard Johnsonbaugh Steve Jost, PHI," *Fundamentals of Digital Image Processing"*.

Web Resources:

1. https://nptel.ac.in/courses/106/105/106105032/
2. https://www.coursera.org/courses?languages=en&query=digital%20image%20processing

ARTIFICIAL INTELLIGENCE	
Course Code: CSE 312(D)	**Credits : 3**
Instruction : 3 Periods /Week	**Sessional Marks : 40**
End Exam : 3 Hours	**End Exam Marks : 60**

Prerequisites:

- Probability and Discrete Mathematics, Basic knowledge on algorithms and programming language

Course Objectives: The course should enable the students:

- To acquire knowledge on basic principles, techniques, and applications of artificial intelligence.
- To have better insight into knowledge representation, problem-solving, and learning methods in solving particular engineering problems.

Course Outcomes:

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

1.	Interpret the fundamental concept of Artificial Intelligence and solve the problems by applying suitable searching methods.
2.	Illustrate the problems using logic based knowledge representation.
3.	Apply statistical reasoning models to solve the problems .
4.	Summarize the concept of planning and natural language processing.
5.	Classify the various approaches of the Expert system.

Mapping of Course Outcomes with Program Outcomes:

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

SYLLABUS

Unit –I **(12 Hours)**

Introduction to Artificial Intelligence: History, Turing Test, AI problems, Artificial Intelligence Techniques, , Applications of AI; **Problems, Problem Spaces and Search:** Defining the problem as State space search, production systems, Problem characteristics, Production System characteristics, Issues in the Design of Search Programs; **Searching Techniques: Uninformed Search Strategies:** Breadth first search, Depth first search, **Heuristic Search Techniques:** Generate & Test, simple and Steepest-Ascent Hill Climbing, Best First search, A* Algorithm, Constraint satisfaction, Means-End Analysis.

Learning Outcomes: At the end of this unit, Student will be able to

1. Interpret the basics of Artificial Intelligence Techniques, Problem Characteristics, and Production system
2. Demonstrate searching strategies and identify the Searching algorithms to solve the real world problems.

Unit –II **(10 Hours)**

Knowledge Representation: Issues-Representation and Mappings**,** approaches to knowledge representation, Issues in Knowledge representation.

Using Predicate Logic: Representing simple facts in logic, Representing Instance and ISA relationship, Computable Functions and Predicates, Resolution, Convert into clause forms, unification algorithm.

Representing Knowledge using Rules: Procedural vs Declarative Knowledge, Logic Programming, Forward vs Backward reasoning, Matching, Control Knowledge.

Learning Outcomes: At the end of this unit, Student will be able to

1. Describe the concept of knowledge representation approaches,rules and reasonings.
2. Determine the knowledge representations in the form of predicate logic

Unit-III **(10 Hours)**

Statistical Reasoning: Probability and Bayes' Theorem, Certainty Factor and Rule based System, Bayesian Networks, Dempster-Shafer Theory

Weak Slot Filler Structures: Semantic nets, Frames.

Strong Slot Filler Structures: Conceptual dependencies, Scripts

Learning Outcomes: At the end of this unit, Student will be able to

1. Demonstrate the concepts of statistical reasoning approaches.
2. Examine the knowledge on slot filler structures

Unit-IV **(10 Hours)**

Planning: Overview, An Example Domain: The Blocks World, Components of a planning system, Types of Planning: Goal Stack planning, Non-Linear Planning, Hierarchical Planning **Introduction to Natural Language Processing:** Introduction, Syntactic Processing, Semantic Analysis, Discourse and Pragmatic Processing, Statistical Natural Language Processing, Spell Checking, Applications of NLP.

Case study: Text Data Preprocessing.

Learning Outcomes: At the end of this unit, Student will be able to

1. Compare the planning approach for the real world scenario.
2. Describe the various steps involved in Natural language processing.

Unit-V **(8 Hours)**

Strong Method Problem Solving: Introduction, Overview of Expert System Technology, Rule Based Expert System, Model Based, Case Based and Hybrid Systems.

Learning Outcomes: At the end of this unit, Student will be able to

1. Describe the concept of the Expert system.
2. Outline the various approaches of the Expert System.

Text Books:

1. Rich E & Knight K, "Artificial Intelligence", 4th Edition, Tata McGraw hill, 2010. (UNIT:I-IV)
2. George F Luger, "Artificial Intelligence: Structure and strategies of complex problem solving", 6th Edition, Addison Wisley, 2009. (UNIT-V)

References:

1. Stuart Russell and Peter Norvig, "Artificial Intelligence: A Modern Approach", 2nd Edition, Pearson Education, 2010.
2. DAN.W. Patterson, "Introduction to A.I and Expert Systems", 1st Edition, PHI, 2007.

Web References:

1. https://nptel.ac.in/courses/106/105/106105077/
2. https://www.udemy.com/course/natural-language-processing-with-python-and-nltk/

COMPETITIVE PROGRAMMING	
CSE 313	**CREDITS:3**
INSTRUCTION: 2 Periods & 1 Tutorial/Week	**SESSIONAL MARKS: 40**
FINAL EXAM: 3Hrs	**FINAL EXAM MARKS: 60**

Pre requisites:

- C and Data Structures

Course Objectives:

The focus of the course is to

- Development and implementation of advanced algorithms
- Develop the skills required for programming competitions.
- Learn to select appropriate algorithms for a given problem, integrate multiple algorithms
- Solving a complex problem, designing new algorithms, and implementing them .
- Solving problems in teams and working under time pressure.

Course Outcomes:

By the end of the course Students will be able to

1.	Judge time complexity rules and Apply sorting, searching techniques to solve problems.
2.	Select the best way for finding path using flows and cuts, string algorithms and greedy algorithms.
3.	Analyze and develop backtracking algorithms
4.	Examine bit manipulation operations and work with number theory.
5.	Solve given problems using dynamic programming.

CO-PO Mapping

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

SYLLABUS

UNIT-I **10 Hours**

Time complexity: Calculation rules, Complexity classes, Estimating efficiency, Maximum subarray sum, **Sorting:** Sorting Theory, Counting Sort, Radix Sort, Heap Sort, Bucket Sort, Ternary Search, **Case Studies** : The 3n + 1 Problem (UVa IDs: 100), Minesweeper (UVa ID: 10189).

Learning Outcomes: At the end of this unit the Students will be able to

- Compare various sorting and searching algorithms
- Apply the algorithms to solve a problem.

UNIT- II **14 Hours**

Flows and Cuts: Ford-Fulkerson algorithm, Path Covers, **Directed graphs:** Topological sorting , **Strings:** Trie, String Hashing, Z-algorithm, **Greedy Algorithms :** coin problem, tasks and deadlines, Minimizing sums, Data compression

Case Studies: Jolly Jumpers(UVa ID: 10038)

Learning Outcomes: At the end of this unit the Students will be able to

- Evaluate various flows and cut, directed graphs and string algorithms.
- Finding best path greedy approach.

UNIT-III **12 Hours**

Backtracking: Backtracking**,** Constructing All Subsets, Constructing All Permutations, Program Design Example: The Eight-Queens Problem, Pruning Search.

Case studies: 15-Puzzle Problem , Tug of War

Learning Outcomes: At the end of this unit the Students will be able to

- construct all sub sets and permutations using back tracking
- Solve given problems using back tracking.

UNIT-IV **14 Hours**

Bit Manipulation: Bit representation, bit operations, representing sets

Number theory: Primes and factors, Modular arithmetic, solving equations.

Learning Outcomes: At the end of this unit the Students will be able to

- Examine bit manipulation operations and representing sets
- Work with Primes and factors and Modular arithmetic.

UNIT-V **10 Hours**

Dynamic Programming: Coin problem, Longest increasing subsequence, Paths in a grid, Edit distance, Counting tilings.

Case Studies: Is Bigger Smarter? (UVa Ids:10131), Distinct Subsequences (UVa Ids:10069)

Learning Outcomes: At the end of this unit the Students will be able to

- Examine the different dynamic programs.
- Apply the dynamic programs to solve any scenarios

Text Books:

1. Felix Halim and Steven Halim Competitive Programming 3: The New Lower Bound of Programming Contests.
2. Steven S. Skiena Miguel A. Revilla PROGRAMMING CHALLENGES, The Programming Contest Training Manual.

Reference Books:

1. ART OF PROGRAMMING CONTEST C Programming Tutorials | Data Structures | Algorithms by Ahmed Shamsul Arefin.

Online Resources:

Books:

1. https://jadi.net/wp-content/uploads/2017/07/competetive-programmers-handbook.pdf
2. http://acm.cs.buap.mx/downloads/Programming_Challenges.pdf
3. https://www.comp.nus.edu.sg/~stevenha/myteaching/competitive_programming/cp1.pdf
4. https://www.comp.nus.edu.sg/~stevenha/database/Art_of_Programming_Contest_SE_for_uva.pdf

Problem reading & Practice:

1. https://uva.onlinejudge.org/index.php

COMPILER DESIGN	
CSE 314	**Credits : 3**
Instruction : 2Hours & 1 Tut/week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:

- Basic fundamentals of Discrete Mathematics
- Principles of Automata Theory.

Course Objectives:

- Introduce the major concept areas of language translation and compiler design.
- Learn the design of lexical analyzer, syntax analyzer.
- Enrich the knowledge in various phases of compiler and its use, intermediate code generation, optimization techniques, machine code generation, and use of symbol table.
- Provide practical programming skills necessary for constructing a compiler.

Course Outcomes:

By the end of the course, the student will be able to:	
1	Identify the challenges of theory of computations and Explain different phases of a compiler and design of lexical analyzer, and differentiate between various parsers.
2	Explain how Top down parsing is done
3	Identify the differences in the functioning of various bottom up parsers
4	Differentiate different intermediate code generation techniques and apply a variety of optimization techniques to improve the code.
5	Compare different code generation techniques, and how symbol table and run time storage are managed.

Mapping of course outcomes with program outcomes:

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

SYLLABUS

UNIT-I: **11 Hours**

The Theory of Automata: Overview of Finite Automata and Formal Languages.
Overall view of Compilers: Types of Translators, Brief discussion on various phases of Compilers, Design of lexical analyzer, LEX tool.
Learning Outcomes:

- Explain the Phases of compiler and implementation of Lexical Analyzer phase
- Explain the process of Compilation i.e., conversion from High level language to low level language basing on different phases of compiler.

UNIT-II: **10 Hours**

Design of Parsers:
Top down Parsers, Problems with Top down Parsers, Backtracking, Left recursion, Left factorial, Predictive Parser.
Learning Outcomes:

- Identify the problems of Top down Parser and explain the working of Top down Parser
- Explain the working procedure of syntax analysis phase with help of Top Down Parsing methods using Top Down Parsers

UNIT-III: **13 Hours**

Bottom up parser: Shift Reduce parser, Operator Precedence Parser, LR parser: LR(0), SLR,CLR parsers. LALR parser, parsing of string, YACC TOOL.
Learning Outcomes:

- Identify different Bottom up Parsers and able to implement Syntax analysis phase by using bottom up parsers.
- Explain the working procedure of syntax analysis phase with help of Bottom up Parsing methods using Different Bottom up Parsers

UNIT-IV: **13 Hours**

Syntax Directed Translation:
Syntax directed translation and implementation, Intermediate code, Postfix notation, DAG, Three address Code, Quadruples, and Triples, indirect triples.
Machine independent Code Optimization: The principle sources of optimization, local Optimization, Loop Optimization, DAG, Global data flow analysis.
Learning Outcomes:

- Explain different machine independent code optimization techniques.
- Explain how intermediate code can be represented and how to optimize the intermediate code.

UNIT-V: **13 Hours**

Code Generation:
Problems, Machine model, A simple code generator, Machine dependent code Optimization**,** Register allocation and assignment, Code generation from DAG, Peephole optimization.
Brief discussion on symbol tables, Run-time storage administration.
Learning Outcomes:

- Explain different machine dependent code optimization techniques and working of symbol table
- Explain how intermediate code can be optimized using machine dependent techniques and the use and working of symbol table in Compilation process.

Text Book:

1. Aho, D. Ullman "*Principles of Compiler Design* ",Second Edition,Pearson Education

Reference Books:

1. Santanu Chattopadhyay, "*Compiler Design*", Sixth Edition,PHI Learning Pvt. Ltd.
2. A.A.Puntambekar , *"Compiler design".* First Edition, Technical Publications .

Web resources:

1. http://nptel.ac.in/courses/106104123/.
2. https://www.youtube.com/playlist?list=PLEbnTDJUr_IcPtUXFy2b1sGRPsLFMghhS.
3. http://www.nptelvideos.in/2012/11/compiler-design.html.

DATA BASE MANAGEMENT SYSTEMS	
CSE 315	**Credits : 3**
Instruction **:** 3 Periods	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:

- Elementary knowledge about computers including some experience using UNIX or Windows. Knowledge about data structures and algorithms, corresponding to the basic course on Data Structures and Algorithms.

Course Objectives:

- Understand basic database concepts, including the structure and operation of the relational data model.
- Understand logical database design principles, including E-R diagrams and database normalization.
- To learn the basics of SQL and construct queries using SQL.
- Understand the concept of database transaction and concurrency control, backup and recovery, data object locking and protocols.

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Describe basic concepts of database systems and principles of transaction processing, concurrency techniques and recovery of database.
2.	Apply Conceptual and logical database design principles, including E-R diagrams.
3.	Compose SQL queries to perform operations on database. (Create, Retrieve, Update, Delete)
4.	Construct relational algebra expressions for queries
5.	Analyze and apply schema Refinement ,database normalization principles.

Mapping of Course Outcomes with Program Outcomes:

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

1-low,2-Medium 3-strong

SYLLABUS

UNIT-I: **12 periods**

Introduction to DBMS:
Overview of DBMS, File system versus a DBMS , Advantages of a DBMS, Three Schema architecture of DBMS, Data Models, Database Languages, Transaction Management , Structure of a DBMS ,Client/Server Architecture ,Database Administrator and Users.

Entity-Relationship Model:
Design Issues, ER Modeling concepts , Cardinality constraints, Weak-entity types, Subclasses and inheritance, Specialization and Generalization, Conceptual Database Design With the ER Model.

Learning Outcomes: At the end of this unit the Students will be able to

- To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.
- Describe the fundamental elements of relational database management systems and design ER-models to represent simple database application scenarios

UNIT-II: **10 periods**

Relational Model:
Structure of Relational Databases, Basics of Relational Model ,Integrity Constraints, Logical Database Design, Introduction to Views, Destroying/ Altering Tables and Views, Relational Algebra, Relational Calculus.

Learning Outcomes: At the end of this unit the Students will be able to

- Design entity relationship and convert entity relationship diagrams into RDBMS and formulate SQL queries on the respect data into RDBMS and formulate SQL queries on the data.
- Recall Relational Algebra concepts, and use it to translate queries to Relational Algebra statements and vice versa.

UNIT-III: **12 periods**

SQL:
Concept of DDL, DML, DCL, Set operations, Nested queries, Aggregate Functions, Null Values, Referential Integrity Constraints, assertions, views, Embedded SQL ,Cursors Stored procedures and triggers, ODBC and JDBC, Triggers and Active Database, designing active databases.

Learning Outcomes: At the end of this unit the Students will be able to

- Perform PL/SQL programming using concept of Cursor Management, Error Handling, Package and Triggers.
- Use the basics of SQL and construct queries using SQL in database creation and interaction

UNIT-IV: **12 periods**

Database Design:
Schema Refinement, Functional Dependencies, Reasoning about Functional Dependencies, Normalization using functional dependencies, Decomposition, Boyce-Codd Normal Form, 3NF, Normalization using multi-valued dependencies, 4NF, 5NF
Security:
Access Control, Discretionary Access Control - Grant and Revoke on Views and Integrity Constraints, Mandatory Access Control.
Learning Outcomes: At the end of this unit the Students will be able to

- Apply various Normalization techniques , functional Dependency and Functional Decomposition to improve the database design.
- Implement typical security techniques in real time applications.

UNIT-V: **15 periods**

Transaction Management:
The ACID Properties, Transactions & Schedules, Concurrent Execution of Transactions, Lock-Based Concurrency Control.
Concurrency Control: 2PL, Serializability and Recoverability, Introduction to Lock Management, Lock Conversions, Dealing with Deadlocks, Specialized Locking Techniques, Concurrency Control without Locking.
Crash Recovery: Introduction to ARIES, The Log, Other Recovery-Related Structures, The Write-Ahead Log Protocol, Check pointing, recovering from a System Crash, Media Recovery.
Learning Outcomes: At the end of this unit the Students will be able to

- Execute various SQL queries related to Transaction Processing & Locking using the concept of Concurrency control.
- Analyze the crash recovery techniques of database systems and apply transaction processing mechanisms in relational databases.

Text Books:

1. Raghu Ramakrishnan, Johannes *Gehrke "Database Management Systems*", 4th Edition,
   McGraw- Hill

References Books:

1. A.Silberschatz.H.Korth, *"Database System Concepts"* , 5th Edition, McGraw-Hill

Web Resources:

1. https://www.oreilly.com/library/view/web-database-applications/0596005431/ch01.html
2. http://nptel.ac.in/courses/IIT-MADRAS/Intro_to_Database_Systems_Design/pdf/1_Introduction.pdf
3. https://www.edx.org/learn/databases
4. https://www.youtube.com/watch?v=1057YmExS-I

DESIGN AND ANALYSIS OF ALGORITHMS	
CSE 316	**Credits : 3**
Instruction **:** 2 Periods & 1 Tut/Week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:

- Some programming skills and a good back ground in discrete mathematics, data structures and probability will be very helpful.

Course Objectives:

- Student will understand the basic design concepts (e.g., pseudo code, specifications, top-down design).
- Student will learn the different algorithm design strategies (procedural knowledge).
- Student can acquire the knowledge to solve the complexities of different problems.
- Student will able to choose appropriate design strategies for different problems.

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Demonstrate knowledge about basic design concepts (e.g., pseudo code, specifications, top-down design).
2.	Use and explain the algorithms for different design strategies.
3.	Apply the algorithms and design strategies to solve problems.
4.	Analyze the complexities of various problems in different domains.
5.	Categorize the notions of P and NP problems, NP complete and NP-hard problems.

Mapping of Course Outcomes with Program Outcomes:

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

SYLLABUS

UNIT-I: **12 periods**

Introduction :
Introduction, Steps for algorithmic problem solving , Important Problem Types Analysis framework (Orders of growth, Cases), Asymptotic Notations and Efficiency Classes, Mathematical Analysis for recursive Algorithms and Non-recursive Algorithms, Empirical Analysis, Algorithm Visualization.
Case Study: Pseudo code Conventions, Time and Space Complexities
Learning Outcomes:

- Argue the correctness of algorithms using inductive proofs and invariants.
- Analyze worst-case running times of algorithms using asymptotic analysis.

UNIT-II: **14 periods**

Brute Force:
Brute Force- Selection and Bubble sort, Sequential Search, String Matching, Closest- Pair, Convex Hull Problems, Exhaustive Search -Travelling Salesman problem, knapsack problem, Assignment Problem.
Decrease and Conquer:
Decrease by a constant: Insertion Sort, Algorithms for generating combinatorial problems, Decrease by constant factor algorithms, Variable size decrease.
Divide-and-Conquer :
Merge sort, Quick sort, Binary Search, Multiplication of large integers and Stassen's Matrix Multiplication, Closest- Pair, Convex Hull Problems.
Learning Outcomes:

- Describe the divide-and-conquer paradigm and explain when an algorithmic design situation calls for it.
- Recite algorithms that employ this paradigm.

UNIT-III: **12 periods**

Transform and conquer:
Presorting, Gauss Elimination, Balanced Trees –2-3 Trees, Heap sort, Horner's rule and binary exponentiation, Problem reduction.
Dynamic Programming:
Computing a Binomial Coefficient, Warshall's and Floyd's Algorithm, Optimal Binary Search Trees, The Knapsack Problem and Memory Functions.
Learning Outcomes:

- Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it.
- Recite algorithms that employ this paradigm. Synthesize dynamic-programming algorithms, and analyze them.

UNIT-IV: **12 periods**

Greedy Technique: Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm – Huffman Trees. **Space And Time Tradeoffs:** Sorting by computing, Input Enhancement in String Matching-Horspool's Algorithm, Boyer-Moore Algorithm, Hashing, B-Trees

Learning Outcomes:

- Describe the greedy paradigm and explain when an algorithmic design situation calls for it.
- Recite algorithms that employ this paradigm. Synthesize greedy algorithms, and analyze them.

UNIT-V: **14 periods**

Limitations of Algorithm Power: Lower-Bound Arguments, Decision Trees, P, NP and NP complete problems, Challenges of Numerical Algorithms

Coping with the limitations of Algorithms Power – Backtracking, Branch-and-Bound Case study for Backtracking: Graph Coloring

NP Problems - Approximation Algorithms for NP-hard Problems, Algorithms for solving Nonlinear Equations.

Learning Outcomes:

- Explain what competitive analysis is and to which situations it applies.
- Perform competitive analysis.

Text Books:

1. Anany Levitin,*"Introduction to Design & Analysis of Algorithms*", 2003, Pearson Education, New Delhi.

Reference Books :

1. Ellis Horowitz, S. Sahni et.al,"*Fundamentals of Computer Algorithms*",2001,Galgotia Pub.
2. Thomas H. Corman, Charles E. Leiserson, Ronald R. Rivest & Clifford Stein, "*Introduction to Algorithms*" Prentice Hall of India, New Delhi
3. Aho, Hopcroft & Ullman,"*The Design and Analysis of computer Algorithms*",2003 Pearson Education, New Delhi
4. Gilles Brassard & Paul Bratley,"*Fundamentals of Algorithmic*", Prentice Hall of India, New Delhi

Web Resources:

1. http://nptel.ac.in/courses/106101060/
2. https://www.edx.org/course/subject/data-analysis-statistics
3. https://www.udacity.com/courses/data-science
4. https://www.coursera.org/specializations/algorithms

Proposal for Industry collaborated

Course in Engineering Curriculum

"SOFT SKILLS"

For

THIRD YEAR- B.TECH.

Designed in collaboration with

Infosys Limited

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY & SCIENCES

(UGC AUTONOMOUS)

(Affiliated to AU, Approved by AICTE & Accredited by NBA)

SANGIVALASA-531 162, Bheemunipatnam Mandal,

Visakhapatnam District

31-07-2021

SOFT SKILLS

Contents

1. Background

ANIL NEERUKONDS INSTITUTE OF TECHNOLOGY & SCIENCES (A) has partnered with Infosys Limited to roll-out Campus Connect Program. Under this program we have conducted Soft Skills training for eligible students from all the branches from 2010 to 2013. They were later selected through the Infosys drives on Campus. Our faculty was enabled in delivering these courses on campus. Certificates were issued after training to the students.

The purpose of this proposal is to describe the Soft Skills course for B.Tech. students and seek approval to make it in par with the campus connect programme to meet the Industrial need.

2. Objectives

- To inculcate effective communication skills with appropriate body language.
- To produce potent leaders, productive team players and effective individuals with proper professional ethics.
- To enable students to make successful oral presentations using relevant content.
- To train students for Group discussions and job Interviews which improves their employability skills.
- To make the students understand the importance of setting realistic goals and achieving them using time management techniques.

3. Overview of the Course Design

3.1 Synopsis:

- The proposed course exposes the engineering students to those soft skills which are crucial to an employee's ability to work "smarter". The Core Modules of this course includes Strengthening English, Art of Communication, Working in Teams, and Interview & GD handling skills amongst other related topics. The course enhances the overall effective communication skills and soft skills in the B. Tech students which improves their employability skills.

3.2 Prerequisites:

- Basic English language skills- LSRW at Intermediate Level.

3.3 Assumptions:

1. This elective will be applicable to B.E/B.Tech/M.E/M.Tech students
2. The duration of the course will be One Semester
3. The elective design follows University Curriculum standards
4. There will be a compulsory final Examination
5. The elective will be designed in exclusive collaboration with Infosys
6. The college will leverage existing Lab & IT infrastructure

4. Learning outcomes

At the end of the course, students will be able to:	
1	Have competent knowledge of grammar with an understanding of its basic rules for Communication Etiquette
2	Communicate effectively and enhance their interpersonal and presentation skills with confidence.
3	Work together effectively in teams and accomplish objectives in a cordial atmosphere
4	Participate in group discussions, present analytical perception on various issues with confidence.
4	Attend professional interviews, Participate in group discussions and give effective presentations.
5	Display corporate etiquettes and Importance of Ethics and Values

5. Course Schedule Summary (Illustrative only)

Here it is illustrated for one semester course.

Duration of the Course	**Number of Weeks**	**Total Lecture hours**	**Total Practical hours**	**Total Credit**
One semester	15 – 16 Weeks	01 hour per week	03 hour per week	1.5

6. Course contents (SYLLABUS)

SYLLABUS

UNIT-I: Communicative English –Written and Spoken English **9 Periods CO1**

Theory:

Formal and informal expressions in business communication, Telephone etiquette, Just a Minute (JAM) procedure, Extempore – Tips, Information Transfer-(IELTS) tables, bar diagrams, and pie charts) Book Review Assignments/Contests

LAB:

1. Basic Grammar- Parts of Speech, Tenses, Subject-verb Agreement
2. Sentence Formation- Vocabulary, Synonyms, Antonyms, Idioms & Phrases
3. Dialogues, Conversations and Role Plays- Written and Spoken Communication
4. Paragraph and Content writing tasks.

Developmental Assignment (DA)

1. Just a Minute (JAM) practice of different postures and gestures and activity on giving feedback and role plays listening news and reading Newspapers,

2. Graphs Interpretation evaluation

Learning outcome: at the end of this unit student are able to

Have competent knowledge of grammar with an understanding of its basic rules for Communication Etiquette.

UNIT- II: Art of Communication **8 Periods CO2**

Theory: Elocution, analogies, YES-NO statements (sticking to a particular line of reasoning (sticking to a particular line of reasoning). Paragraph writing, supplying a suitable beginning/ending/middle sentence to make the paragraphs.

LAB:

1. Verbal Communication
2. Effective Communication, Barriers To Communication
3. Types of Communication
4. Non-verbal Communication-Postures and gestures
5. Professional grooming and Body language (of self and others)
6. Importance of feelings in communication - dealing with feelings in communication
7. Listening skills- types, analyzing videos and news.
8. Feedback and Communication Etiquette

Developmental Assignment (DA)

1. Speaking Activities followed by individual evaluation

2. Analytical Essay writing assignments.

Learning outcome: at the end of this unit student are able to

Communicate effectively and enhance their interpersonal skills with confidence.

UNIT- III : Team Building, Leadership and Attitude 8 Periods CO3

Theory: Information Transfer-(IELTS) tables, bar diagrams, and pie charts) Book Review Assignments/Contests

LAB:

1. Self Enhancement - developing self confidence
2. developing emotional intelligence
3. Different stages of team formation
4. Attributes of a successful team – Barriers involved
5. Team vs. Group,Team player and Team leader
6. Decision making and negotiating skills
7. Types of leadership
8. Working with Groups – Dealing with People

Developmental Assignment (DA)

1.Group discussions and Individual evaluation

2. Argumentative writing assignments.

Learning outcome: at the end of this unit student are able to understand the team-dynamics and work together effectively in teams and accomplish objectives in a cordial atmosphere.

UNIT- IV: Interview, GD & Presentation Skills 12 Periods CO4

Theory: Resume writing-tips, statement of purpose', 'letters of recommendation

LAB:

1. Different types of interviews, Stress management

2. Interview handling Skills – Self preparation checklist – Grooming tips: do's &don'ts – mock interview & feedback
3. Frequently asked questions FAQ's – (preparation for an interview)
4. Goal-Setting- setting- SMART goals (career related)
5. Decision making for a given situation (negotiation)
6. E mail etiquette-Email format-dos and don'ts.
7. GD skills – Understanding the objective and skills tested in a GD – General types of GDs – Roles in a GD – Do's &Don'ts – Mock GD & Feedback.
8. Presentation Skills – Stages involved in an effective presentation – selection of topic, content, aids – Engaging the audience – Mock Presentations & Feedback

Developmental Assignment (DA)

1. Group presentation by each team

2. Mock interviews evaluation,

3. Writing Resume

Learning outcome: at the end of this unit students are able to

Understand the different types of professional interviews and present themselves with confidence

Unit- V: Corporate Etiquette & Ethics **9 Periods CO4**

LAB:

1. Corporate attire and Grooming
2. Telephone & E-mail etiquette
3. Table Manners
4. Do's & Don'ts in a formal setting
5. Time-management- importance and techniques
6. Ethics – Importance of Ethics and Values
7. Choices and Dilemmas (Case studies and Situational Awareness)
8. Discussions from news headlines (Current Affairs)

Developmental Assignment (DA)

1. Time- Bound Goals - team activities

2. Debates –Teams participation.

Learning outcome: at the end of this unit student are able to understand the way of the corporate world and learn the rules for timely assimilation and adaption.

Text Books: 1. Hurlock, E.B **Personality Development**, 28th Reprint. New Delhi: Tata McGraw Hill. 2006.

Counseling / Mentoring:

The purpose of 1 hour counseling per student per month is to help understand the learning each individual student is gaining and help ensure that all the students benefit from the training. The modules on self enhancement, self esteem / confidence, communication can be supplemented by listening to the students in a non threatening one on one setting. The individual skills of the students can be analysed and the stronger students can be made to help bring up the other students thereby fostering peer learning.

7. Assignments:

1. Essay Writing

2. Resume writing

3. JAMS

4. Book Review

5. Elocution

8.Infrastructure Requirements

LAB / SOFTWARE REQUIREMENTS

- A projector to display PPts. Of modules on screen.
- Soft ware-vocabulary building, IELTS based reading and listening and telephonic skills.
- A Spacious Lab with round tables with chairs to accommodate 70 students to conduct activities in team tasks.

9.Mode of Examination:

The Institute will conduct all the assessments. The examination carries 50 Marks.

Internal assessments for Soft Skills could include the following.

		Total	50
1.	**Internal – Exams**		
	• Written Test (to be conducted along with the midterm e	20 M	25M
	• Attendance	5M	
2	**Developmental Assignment (DA)** **Continuous Evaluation on Verbal and written skills**		
	• Assignments.	5M	
	• Group Presentation (live case studies/current affairs/ social and technological issues).	10M	25M
	• Group Discussion (Factual/Controversial/abstract topics are given)	5M	
	• HR Interview(Each student will face an interview conducted by mentors)	5M	

E. Faculty enablement

The Faculty will be enabled on the course contents; Industry practices case studies etc. for duration of one week before the commencement of Course. Faculties from various colleges are required to stay in the Infosys Campus for their Enablement.

11. REFERENCE BOOKS:

1. *The Seven Habits of Highly Effective People*: Restoring the Character Ethic. 2004. Covey, Stephen R.

2. Allan Pease, **Body Language,** Sheldon Press, 1997
3. John A. Kline and Bhavna Bhalla, ***Speaking Effectively; Achieving Excellence in Presentations***, Pearson publication, 2013.
4. *Effective presentation skills* (a Fifty Minute Series Book) by Steve Mandel
5. *Developing Communication Skills* by Krishna Mohan & Meera Banerji, Mac Millan India Ltd., Delhi
6. Excel in English, Sundra Samuel, Samuel publications.
7. Effective Group Discussion:Theory and Practice by Gloria J.Galanes, Catherind Adams,John K.Brillhart.
8. Marc Mancini, ***Time Management***, Tata McGraw Hill publishing Comp.Ltd.2003.
9. Peter Veruki, ***The 250 Job Interview Questions,*** Adams Media Corporation Avon, Massachusetts, 1999.
10. Text Books: 1. Hurlock, E.B **Personality Development**, 28th Reprint. New Delhi: Tata McGraw Hill. 2006.

11. Stephen P. Robbins and Timothy A. Judge **Organizational Behavior** 16th Edition: Prentice Hall. 2014**.**
12. Monipally M. "***Business Communication Strategies"*** publisher : 5) McGraw Hill Education (2001 October
13. Sanjay Kumar and Pushpalata, ***Communication Skills***, Oxford University Press, 2011.

12. Actions:

1. The college needs to send the Board of Studies Approval letter on college letter head to Infosys.
2. Identify one department to own the responsibility of course content, assignments, projects, software tools etc. (Preferable CS/IS/EC/MCA Department).
3. Identify faculty from English and Verbal Training department for rollout and faculty training.
4. Identify and allocate resources like classrooms, labs, necessary furniture and software for rollout.
5. Complete readiness check before the rollout

13.Contact Details:

The Infosys point of contact can be reached for more info. In addition, the Institute SpoC can also be reached for additional info.

Department owning the responsibility of Course Content:

The HOD'S / Faculty Names and their Email Id, owning the course content of Soft skills are to be mentioned.

S. No.	Name	E-Mail	Phone Number
1	Dr. G.Serwani Swamy (I/CHOD-English & Humanities)	hod_english@anits.edu.in	9866094152
2	Mr. S. Ratan Kumar (SPOC)	Sratankumar.cse@anits.edu.in	9052492777

14. Faculties handling the Elective rollout:

The faculty names and their Email Id, handling the Soft Skills Course are as mentioned.

S. No.	Name	E-Mail	Phone Number
1	Dr. G.Serwani Swamy (I/CHOD-English & Humanities)	hod_english@anits.edu.in	9866094152
2	Ms. Md. Sabirunnisa Gouse	sabirunnisa.english@anits.edu.in	9866297372
4	Dr. Abhibunnisha Begum	abegum.english@anits.edu.in	8179828399
5	Dr. P. Suresh Kumar	sureshkumar.english@anits.edu.in	8639367609
6	Dr.P Prasanna	prasanna.english@anits.edu.in	7386807085
7	Ms. T. Sunandha Tulasi	sunandhatulasi.english@anits.edu.in	9618231323
8	Ms. Sudha Singh	sudhasingh.english@anits.edu.in	9000509920
9	MS. Andrea Mallya- T&P	andreamallya.tp@anits.edu.in	8886385007
10	Mr.P.Shyam Kishore - T&P	pshyamkishore.tnp@anits.edu.in	6300493953
11	Mr. B Yogesh - T&P	yogeshbavana.tnp@anits.edu.in	9652974160

15. Conclusion:

The course enhances effective communication skills and soft skills in the B. Tech students to be industry aligned and better leverage their technical as a competitive edge in their career while working in their own discipline or specialization.

****

COMPETITIVE PROGRAMMING LAB	
CSE 318	**Credits : 1.5**
Instruction **:** 3 Periods /Week	Sessional Marks : 50
End Exam : 3 Hours	End Exam Marks : 50

Prerequisites:

- C and data Structures

Course Objectives:

- Learn to select appropriate algorithm for a given problem, integrate multiple algorithms
- Solve a complex problem, design new algorithms, and implement them.

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Apply the basic, sorting and searching techniques to solve the problem components etc.
2.	Apply the concepts of path finding algorithms for flows and cuts, strings and greedy algorithms
3.	Develop solutions for the back tracking algorithms and bit manipulations
4.	Apply the number theory and knowledge of dynamic programming to the real time scenario.

Mapping of Course Outcomes with Program Outcomes:

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

List of Experiments

1. **Maximum subarray sum** **CO1**

   Solve the Maximum subarray sum with different time complexities.

2. **The Median of Two Sorted Arrays** **CO1**

   Program to find the median of two sorted arrays of same size and different size are discussed here. Firstly, let us see what is median of the array? Median is an element which divides the array into two parts - left and right. So the number of elements on the left side of the array will be equal or less than the number of elements on the right side. Now, let us consider the case of an array with odd number of elements. Array = [9,11,16,7,2] Sorted array = [2,7,9,11,16]. In this case, the median of this array is 9, since it divides the array into two parts: [2,7] and [11,16]. Further, let us consider the case of an array with even elements. Array = [1,2,3,4,5,6]. In such a case, we will take the average between the last element of the

left part and the first element of the right part. In this case, the median equals = (3 + 4) / 2 = 3.5.

Input Format

The input should contain 3 lines.

I. First line of the input should contain two integer values which specify the number of elements in array1 and array2.

II. Second line of the input should contain the elements of the first array.

III. Third line of the input should contain the elements of the second array.

Constraints

All elements must be Integers

Output Format

The output should print only the median value.

Sample Input 1

5 6
-5 3 6 12 15
-12 -10 -6 -3 4 10

Sample Output 1

3

Sample Input 2

4 6
2 3 5 8
10 12 14 16 18 20

Sample Output 2

11

3. Division with Binary Search **CO1**

We can modify binary search algorithm to perform division of two numbers, by defining range [0, infinity] which serves as initial low and high for the binary search algorithm. Now we need to find a mid that satisfies x/y = mid or x*mid for given two numbers x and y. Based on the comparison result based on x and y*mid, we either update low, update high or return mid. 1. If y*mid almost equal to x, we return mid. 2. If y*mid is less than x, we update low to mid 3. If y*mid is more than x, we update high to mid We need to care about division by zero and sign of the result etc. Input: one line of input should contain two numbers separated by space. Output: should print division of the numbers as a result.

Input Format

Input: one line of input should contain two numbers separated by space. Input1: 22 7

Constraints

x,y < infinity

Output Format

Output: should print division of the numbers as a result.

Sample Input 1

22 7

Sample Output 1

3.14286

4. Find the Triplet **CO1**

Given an array of integers, find a triplet having maximum product in the array.

Input: First line of input should specify the number of elements in the array.
Second line of input should specify each element separated by space.

Output: should print Triplets.

Test Cases:

Sample Input 1
5
-4 1 -8 9 6

Sample Output 1
-4 -8 9

Sample Input 2
5
1 7 2 -2 5
Sample Output 2
7 2 5

5. **3n+1** **CO1**

Consider the following algorithm to generate a sequence of numbers. Start with an integer n. If n is even, divide by 2. If n is odd, multiply by 3 and add 1. Repeat this process with the new value of n, terminating when n = 1. For example, the following sequence of numbers will be generated for n = 22: 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1 It is conjectured (but not yet proven) that this algorithm will terminate at n = 1 for every integer n. Still, the conjecture holds for all integers up to at least 1, 000, 000. For an input n, the cycle-length of n is the number of numbers generated up to and including the 1. In the example above, the cycle length of 22 is 16. Given any two numbers i and j, you are to determine the maximum cycle length over all numbers between i and j, including both endpoints. The input will consist of a series of pairs of integers i and j, one pair of integers per line. All integers will be less than 1,000,000 and greater than 0. Output For each pair of input integers i and j, output i, j in the same order in which they appeared in the input and then the maximum cycle length for integers between and including i and j. These three numbers should be separated by one space, with all three numbers on one line and with one line of output for each line of input.

Sample Input	**Sample Output**
1 10	1 10 20
100 200	100 200 125
201 210	201 210 89
900 1000	900 1000 174

6. String hacker rank **CO2**

Given a string A, find the common **palindromic sub sequence** (A sequence which does not need to be contiguous and is a palindrome), which is common in itself. You need to return the length of longest palindromic subsequence in A.

Input Format
First and only argument in an string A

Output Format

Return a integer denoting the length of longest palindromic subsequence in A

Sample Input 1

A = "bebeeed"

Sample Output 1

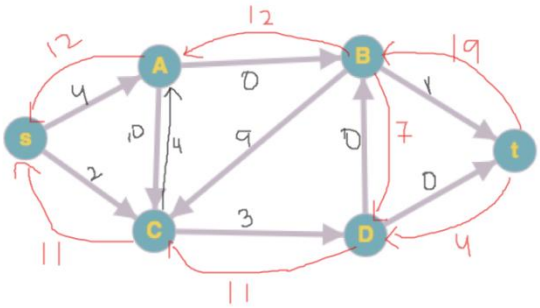
4

Explanation

The longest common pallindromic subsequence is "eeee", which has a length of 4

7. PATH OF RESIDUAL GRAPH CO2

Given a residual graphs find the visited and unvisited node from source to destination, Display as 1 for visited , 0 for unvisited node



The order of output is s,A,B,C,D,t. where *s* is the source and *t* as destination

Output:

1 1 0 1 1 0

Explanation

Indication :s as visited, A is visited ,B as unvisited, C as visited, D as visited and t as unvisited.

INPUT:

Number of vertices, V

Contain a adjacency matrix of size V*V, with weights between the u-v

CONSTRAINTS

Number of vertices <=15

Vertex values is 0 to 14

OUTPUT:

Print the value “1” for visited and “0” for not visited

Sample Input 1

6

0 4 2 0 0 0

12 0 10 0 0 0

11 4 0 0 3 0

0 12 9 0 7 1

0 0 11 0 0 0

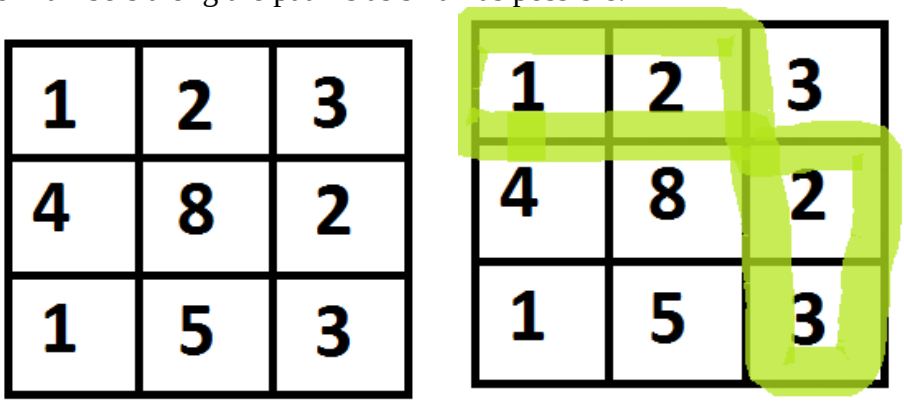
0 0 0 19 4 0

Sample Output 1

1 1 1 0 1 0

8. Minimum-Cost –Path **CO2**

Find a path in an n×n grid from the upper-left corner to the lower-right corner such that we only move down and right and diagonally lower cells from a given cell, i.e., from a given cell (i, j), cells (i+1, j), (i, j+1) and (i+1, j+1) can be traversed. Assume that all costs are positive integers. Each square contains a number, and the path should be constructed so that the sum of numbers along the path is as small as possible.



The path is (0, 0) –> (0, 1) –> (1, 2) –> (2, 2). The cost of the path is 8 (1 + 2 + 2 + 3)

Sample Input 1

1 2 3
4 8 2
1 5 3

Sample Output 1

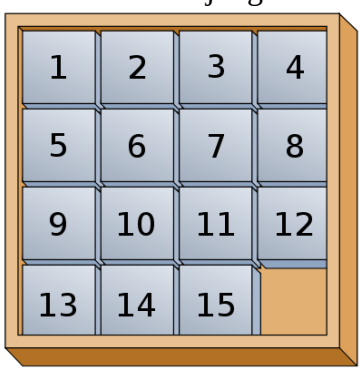
8

9. 15-Puzzle Problem **CO3**

The 15-puzzle is a very popular game: you have certainly seen it even if you don't know it by that name. It is constructed with 15 sliding tiles, each with a different number from 1 to 15, with all tiles packed into a 4 by 4 frame with one tile missing. The object of the puzzle is to arrange the e tiles so that they are ordered as below:

The only legal operation is to exchange the missing tile with one of the 2, 3, or 4 tiles it shares an edge with. Consider the following sequence of moves:

We denote moves by the neighbor of the missing tile is swapped with it. Legal values are "R," "L," "U," and "D" for right, left, up, and down, based on the movements of the hole. Given an initial configuration of a 15-puzzle you must determine a sequence of steps that take you to the final state. Each solvable 15-puzzle input requires at most 45 steps to be solved with our judge solution; you are limited to using at most 50 steps to solve the puzzle.



Input The first line of the input contains an integer n indicating the number of puzzle set inputs. The next 4n lines contain n puzzles at four lines per puzzle. Zero denotes the missing tile.

Output For each input set you must produce one line of output. If the given initial configuration is not solvable, print the line "This puzzle is not solvable." If the puzzle is solvable, then print the move sequence as described above to solve the puzzle.

Sample Input 1

2
2 3 4 0
1 5 7 8
9 6 10 12
13 14 11 15
13 1 2 4
5 0 3 7
9 6 10 12
15 8 11 14

Sample Output 1

LLLDRDRDR
This puzzle is not solvable

10. Tug of War **CO3**

Tug of war is a contest of brute strength, where two teams of people pull in opposite directions on a rope. The team that succeeds in pulling the rope in their direction is declared the winner. A tug of war is being arranged for the office picnic. The picnickers must be fairly divided into two teams. Every person must be on one team or the other, the number of people on the two teams must not differ by more than one, and the total weight of the people on each team should be as nearly equal as possible.

Input

The input begins with a single positive integer on a line by itself indicating the number of test cases following, each described below and followed by a blank line.

The first line of each case contains n, the number of people at the picnic. Each of the next n lines gives the weight of a person at the picnic, where each weight is an integer between 1 and 450. There are at most 100 people at the picnic. Finally, there is a blank line between each two consecutive inputs.

Output For each test case, your output will consist of a single line containing two numbers: the total weight of the people on one team, and the total weight of the people on the other team. If these numbers differ, give the smaller number first. The output of each two consecutive cases will be separated by a blank line.

Sample Input 1

1
3
100
90
200

Sample Output 1

190 200

11. Bit manipulation **CO3**

There are N<= 5000 worker. Each is available during some days of this month. Find two workers with maximum intersection of their schedules.

W1=={ 2,3,5,6,8} = 0110110100
W2== {2,4,5,8} = 0101100100
& = 0100100100. =count=3

12. Euclid Problem **CO4**

From Euclid, it is known that for any positive integers A and B there exist such integers X and Y that AX + BY = D, where D is the greatest common divisor of A and B. The problem is to find the corresponding X, Y , and D for a given A and B.

Input The input will consist of a set of lines with the integer numbers A and B, separated with space (A, B < 1, 000, 000, 001).

Output For each input line the output line should consist of three integers X, Y , and D, separated with space. If there are several such X and Y , you should output that pair for which $X \leq Y$ and $|X| + |Y|$ is minimal.

Sample Input 1

4 6
17 17

Sample Output 1

-1 1 2
0 1 17

13. Coin Problem **CO4**

Given a value V. You have to make change for V cents, given that you have infinite supply of each of C{ C1, C2, .. , Cm} valued coins. Find the minimum number of coins to make the change and print the coins that appear in an optimal solution.

Input:

The first line of input contains an integer T denoting the number of test cases.
The first line of each test case is V and N, V is the value of cents and N is the number of coins.
The second line of each test case contains N input C[i], value of available coins.

Output:

Print the coins appear in an optimal solution and in a newline print the minimum number of coins to make the change and, if not possible print "-1".

Constraints:
$1 \leq T \leq 100$
$1 \leq V \leq 10^6$
$1 \leq N \leq 10^6$
$1 \leq C[i] \leq 10^6$

Sample Input 1

1
7 2
2 1

Sample Output 1

2 2 2 1
4

Explanation :
Testcase 1: We can use coin with value 2 three times, and coin with value 1 one times to change a total of 7.

DATABASE MANAGEMENT SYSTEMS LAB	
CSE 319	**Credits : 1.5**
Instruction **:** 3 Periods/Week	Sessional Marks : 50
End Exam : 3 Hours	End Exam Marks : 50

Prerequisites:

- Elementary knowledge about computers including some experience using UNIX or Windows. Knowledge about data structures and algorithms, corresponding to the basic course on Data Structures and Algorithms.

Course Objectives

- To understand the basics of SQL and construct queries using SQL.
- To learn connectivity between web pages, OLAP, OLTP.

Course Outcomes:

By the end of the course, the student will be able to:

CO1	Make use of basic SQL queries to solve simple problems.
CO2	Solve complex queries using nested queries and joins.
CO3	Construct triggers, views and stored procedures for different scenarios
CO4	Apply the principles of ER model and normalization for schema refinement in logical database design.

.

Articulation Matrix for CO-PO Mapping:

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

SYLLABUS

List of Experiments:

EXPERIMENT NAME	MAPPED TO
1. SQL DDL ,DML Statements	CO1
2. SQL Constraints.	CO1
3. Inbuilt functions in RDBMS.	CO2
4. Aggregate functions	CO2
5. Nested Queries & Join Queries.	CO2
6. Creation and dropping of Views.	CO3
7. Creating Triggers.	CO3
8. Stored Procedures.	CO3
9. Developing a sample application which includes all database design steps like requirements analysis, logical database design, normalization, developing user interface to access database from the application.	CO4

Sample Applications:

1. Development of an Online Course Portal for a campus
2. Book Bank Management System
3. Car Rental Management System
4. Exam/academic system for College Management
5. Real estate Management system
6. University Management System
7. Database manager for a Magazine agency or a newspaper agency
8. Ticket booking for performances
9. Inventory Control System
10. Students management System

REFERENCE BOOKS:

1. Raghu Ramakrishnan, Johannes *Gehrke "Database Management Systems*", 3rd
2. Edition, McGraw- Hill
3. A.Silberschatz.H.Korth, *"Database System Concepts"* , 5th Edition, McGraw-Hill

WEB REFERENCES:

1. https://dev.mysql.com/doc/refman/5.5/en/sql-syntax-data-definition.html
2. https://cse.iitkgp.ac.in/~pabitra/course/dbms/dbms_lab.html

Constitution of India & - Intellectual Property Rights	
CSE 320	**Credits : 0**
Instruction **:** 2 Periods/Week	Sessional Marks : 50

Course Objectives

- To impart knowledge in basic concepts of Constitution of India
- To understand the fundamental principles of Intellectual Property Rights and its importance

At the end of this course,students will be able to:

	COURSE OUTCOMES	Bloom's Level
CO-1	Recognise basic knowledge about the Constitution of India	L1
CO-2	Comprehend the Fundamental Rights and Fundamental Duties of the Indian Citizen to implant morality, social values and their social responsibilities.	L2
CO-3	Familiarize with distribution of powers and functions of Local Self Government, state and central policies and amendment procedure	L2
CO-4	Recognise the fundamental principles of IPR	L1
CO-5	Appraise of IP rights like patents, industrial design, trademark, copyrights for effective protection and utilization of their innovations.	L3

SYLLABUS

Unit 1 - Introduction and Basic Information about Indian Constitution: 10hrs
Meaning of the constitution law and constitutionalism, Historical Background of the Constituent Assembly, Government of India Act of 1935 and Indian Independence Act of 1947, Enforcement of the Constitution, Indian Constitution and its Salient Features, Preamble of the Constitution.

Unit 2 - Fundamental Rights and Directive Principles 10hrs
Fundamental Rights, Fundamental Duties, Directive Principles of State Policy – Its importance and implementation, Scheme of the Fundamental Right to certain Freedom under Article 19, Scope of the Right to Life and Personal Liberty under Article 21

Unit 3 - Administrative organisation & **Amendments 9hrs**
Indian Federal System , Centre and State Relations , President's Rule , Constitutional Amendments , Parliamentary System in India

Unit 4 - Intellectual Property Rights 10hrs
Introduction to IPRs, Basic concepts and need for Intellectual Property – Patents, Copyrights, Geographical Indications, IPR in India and Abroad , Nature of Intellectual Property, Inventions and Innovations – Important examples of IPR

Unit 5 - Registration of IPR's **9hrs**

Meaning and practical aspects of registration of Copy Rights, Trademarks, Patents, Trade Secrets , Industrial Design registration in India and Abroad

TEXTBOOKS:

1. V. Scople Vinod, Managing Intellectual Property, Prentice Hall of India pvt Ltd, 2012
2. S. V. Satakar, ―Intellectual Property Rights and Copy Rights, Ess Publications, New Delhi,2002
3. Brij Kishore Sharma: Introduction to the Indian Constitution, 8th Edition, PHI Learning Pvt. Ltd.
4. Granville Austin: The Indian Constitution: Cornerstone of a Nation (Classic Reissue),Oxford University Press.

REFERENCES:

1. Deborah E. Bouchoux, ―Intellectual Property: The Law of Trademarks, Copyrights,
2. Patents and Trade Secrets, Cengage Learning, Third Edition, 2012.
   PrabuddhaGanguli,Intellectual Property Rights: Unleashing the Knowledge Economy,
3. McGraw Hill Education, 2011.
   Edited by Derek Bosworth and Elizabeth Webster, The Management of Intellectual
4. Property,Edward Elgar Publishing Ltd., 2013.
5. Subhash C. Kashyap: Our Constitution: An Introduction to India's Constitution and
6. constitutional Law, NBT, 2018.
7. Madhav Khosla: The Indian Constitution, Oxford University Press.
8. PM Bakshi: The Constitution of India, Latest Edition, Universal Law Publishing

SEM-II

HUMAN COMPUTER INTERACTION	
CSE 322(A)	**Credits : 3**
Instruction **:** 3 Periods /Week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:

- Student should have an idea about User Interface Design and Programming

Course Objectives:

- The main objective is to get student to think constructively and analytically about how to design and evaluate interactive technologies.

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Make use of four pillars of design, participatory design, scenario development of design processes that supports social, ethical and legal issues.
2.	Apply an interactive design process and universal design principles to design HCI systems.
3.	Analyze Importance of response time, attitudes and user productivity related to quality of service on Display Design, Web Page Design, Window Design HCI Systems.
4.	Distinguish the online user documentation from paper documentation along with online communities' assistance.
5.	Compare searching and visualization methodologies in Textual Documents, Database Querying, and Multimedia Documents.

Mapping of Course Outcomes with Program Outcomes:

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

SYLLABUS

UNIT I: **12 -14 Hours**

Introduction: Usability of Interactive Systems- introduction, usability goals and measures, usability motivations, universal usability, goals for our profession.
Managing Design Processes: Introduction, Organizational design to support usability, Four pillars of design, development methodologies, Ethnographic observation, Participatory design, Scenario Development, Social impact statement for early design review, legal issues, Usability Testing and Laboratories.
Learning Outcomes: At the end of this unit, the student will be able to

- Make use of four pillars of design; know what usability is and what are its goals and measures.
- Analyze the design process.

UNIT II: **16-18 Hours**

Menu Selection, Form Fill-In and Dialog Boxes: Introduction, Task- Related Menu Organization, Single menus, Combinations of Multiple Menus, Content Organization, Fast Movement Through Menus, Data entry with Menus: Form Fill-in, dialog Boxes, and alternatives, Audio Menus and menus for Small Displays.
Command and Natural Languages: Introduction, Command organization Functionality, Strategies and Structure, Naming and Abbreviations, Natural Language in Computing
Interaction Devices: Introduction, Keyboards and Keypads, Pointing Devices, Speech and Auditory Interfaces, Displays- Small and large.
Learning Outcomes: At the end of this unit, the student will be able to

- Compare Different menu selections, and form filling techniques.
- Understand Natural Language Processing and its structure, along with interaction devices.

UNIT III: **14-16 Hours**

Quality of Service: Introduction, Models of Response-Time impacts, Expectations and attitudes, User Productivity, Variability in Response Time, Frustrating Experiences
Balancing Function and Fashion: Introduction, Error Messages, Non anthropomorphic Design, Display Design, Web Page Design, Window Design, Color.
Learning Outcomes: At the end of this unit, the student will be able to

- Analyze Importance of response time, Different issues related to response times.
- Apply function and fashion in designing web pages.

UNIT IV: **7-9 Hours**

User Documentation and Online Help: Introduction, Online Vs Paper Documentation, Reading from paper Vs from Displays, Shaping the content of the Documentation, Accessing the Documentation, Online tutorials and animated documentation, Online communities for User Assistance, The Development Process.
Learning Outcomes: At the end of this unit, the student will be able to

- Compare user documentation and online help.
- Identify the need for online communities for user assistance.

UNIT V: **7-9 Hours**

Information Search: Introduction, Searching in Textual Documents and Database Querying, Multimedia Document Searches, Advanced Filtering and Searching Interfaces Information Visualization: Introduction, Data Type by Task Taxonomy, Challenges for Information Visualization

Learning Outcomes: At the end of this unit, the student will be able to

- Compare different searching methodologies in Textual Documents.
- Apply visualization techniques in designing applications.

Text Books:

1. Designing the User Interface, Strategies for Effective Human Computer Interaction, 5ed, Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven M Jacobs, Pearson
2. The Essential guide to user interface design,2/e, Wilbert O Galitz, Wiley DreamaTech.

Reference Books :

1. Human Computer, Interaction Dan R.Olsan, Cengage ,2010.
2. Designing the user interface. 4/e, Ben Shneidermann , PEA.
3. User Interface Design, Soren Lauesen , PEA.
4. Interaction Design PRECE, ROGERS, SHARPS, Wiley.

Web Resources:

1. https://onlinecourses.nptel.ac.in/noc18_cs23/preview

MOBILE COMPUTING	
CSE 322(B)	**Credits : 3**
Instruction **:** 3 Periods /Week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Pre-requisite: Data Communication, Computer Network

Course Objectives:

- To make the student understand the concept of mobile computing terminology and basic services.
- To interpret the knowledge on the working principle of wireless technology and applications of wireless protocols.
- To make the student aware of various architectures and technologies in mobile networking.
- To gain sufficient knowledge on various routing mechanisms.

COURSE OUTCOMES:

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

1.	Interpret the GSM architecture and its services.
2.	Analyze the various wireless applications and study technical feasibility of various mobile applications.
3.	Develop the mobile network layer protocols and its functionalities.
4.	Analyze & develop any existing or new models of mobile environments for 3G networks.
5.	Evaluate and create the platform, protocols and related concepts of Ad hoc and Enterprise wireless networks

CO-PO Mapping:

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

SYLLABUS

UNIT-1 **12 Hours**

Introduction: Mobile Communications, Mobile Computing – Paradigm, Promises/Novel Applications and Impediments and Architecture; Mobile and Handheld Devices, Limitations of Mobile and Handheld Devices.

Global System for Mobile communication(GSM): Services, System Architecture, Radio Interfaces, Protocols, Localization, Calling, Handover, New Data Services, GPRS Architecture, GPRS Network Nodes.

Learning Outcomes: The student will be able to

- Explain the basics of mobile communications and services.
- Interpret the mobile communication using GSM architecture.

UNIT-2 **12 Hours**

Medium Access Control (MAC) : Motivation for a specialized MAC (Hidden and exposed terminals, Near and far terminals), Wireless LAN/(IEEE 802.11) architecture, key IEEE802.11 a/b/c/d/e/g/i/n/T/ac/ standards.

Wireless Application Protocol (WAP): The Mobile Internet standard, WAP Gateway and Protocols, wireless mark up Languages (WML). Wireless Local Loop(WLL): Introduction to WLL Architecture, wireless Local Loop Technologies.

Learning Outcomes: The student will be able to

- Analyse the various IEEE Standards and their architecture.
- Explain the technical feasibility of various protocols Using WAP and WLL archtiecture.

UNIT-3 **12 Hours**

Mobile Network Layer : IP and Mobile IP Network Layers, Packet Delivery and Handover Management, Location Management, Registration, Tunneling and Encapsulation, Route Optimization using Soft computing techniques – ANT Bee colony, Support Vector Machine, Particle Swarm Optimization and Genetic Algorithm.

Learning Outcomes: The student will be able to

- Interpret the properties of Mobile Network layer
- Analyse various optimization techniques using soft computing.

UNIT-4 **12 Hours**

Mobile Transport Layer : Conventional TCP/IP Protocols, Indirect TCP, Snooping TCP, Mobile TCP.

Third Generation (3G) Mobile Services: Introduction to International Mobile Telecommunications 2000 (IMT 2000) vision, Wide-band Code Division Multiple Access (W-CDMA) and CDMA 2000, Quality of services in 3G.

Learning Outcomes: The student will be able to

- Develop various mobile Transport layer protocols.
- Analyse 3G mobile services.

UNIT-5 **12 Hours**

Mobile Ad hoc Networks (MANETs) : Introduction, Applications & Challenges of a MANET, Routing, Classification of Routing Algorithms, Algorithms such as DSR, AODV, DSDV, etc. , Mobile Agents, Service Discovery ,case study using NS2 –traffic analysis using CBR and VBR

Wireless Enterprise Networks: Introduction to Virtual Networks, Blue tooth technology, Blue tooth Protocols.

Learning Outcomes: The student will be able to

- Evaluate parameters on MANETs and its associated routing algorithms.
- Create a Local Area ne network using NS2 simulator.

TEXT BOOKS:

1. Jochen Schiller, "Mobile Communications", Addison-Wesley, Second Edition, 2009.
2. Raj Kamal, "Mobile Computing", Oxford University Press, 2007, ISBN: 0195686772

REFERENCE BOOKS:

1. Asoke K Talukder, Hasan Ahmed, Roopa R Yavagal, "Mobile Computing, Technology Applications and Service Creation" Second Edition, Mc Graw Hill.
2. Martin Sauter, "From GSM to LTE-Advanced: An Introduction to Mobile Networks and Mobile Broadband," Second Edition, Wiley.

Web Resource:

1. https://onlinecourses.nptel.ac.in/noc18_cs09/preview

NOSQL DATABASES	
CSE 322(C)	**Credits:3**
Instruction : 3 Hours /Week	Sessional Marks : 40
End Exam : 3 Hours	Ena Exam Marks : 60

Prerequisites:

- Knowledge on Relational Database management systems.

Course Objectives:

- Distinguish and describing how NoSQL databases differ from relational databases from a theoretical perspective.
- Explore the origins of NoSQL databases and the characteristics .
- Demonstrate competency in selecting a particular NoSQL database for specific use cases.
- Demonstrate Document databases with MongoDB.

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Compare and contrast the uses of relational RDBMSs and NoSQL systems for different types of data and applications.
2.	Differentiate various data models.
3.	Recognize Key value Databases and document databases.
4.	Create a sample database using NoSql.
5.	Apply the Query concepts in MongoDB database

Mapping of course outcomes with program outcomes:

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

SYLLABUS

UNIT-I : **10 Hours**

Why NoSQL?
The value of relational databases – Impedance mismatch – Application and integration databases – Attack of the cluster.
Learning Outcomes: The student will be able to

- Recall Relational databases and security aspects in Realtime
- Identify working with multiple databases.

UNIT-II : **18 Hours**

Aggregate Data Models :
Aggregates - Example of Relations and Aggregates – Consequences of Aggregate Orientation - Key-Value and Document Data Models - Column-Family Stores
More Details on Data Models :
Relationships - Graph Databases – Schemaless Databases - Materialized Views - Modeling for Data Access.
Learning Outcomes: The student will be able to

- Explain internal operations on Database.
- Analyze how to view data from database in different ways.

UNIT –III : **16 Hours**

Distribution Models :
Single Server – Sharding - Master-Slave Replication - Peer-to-Peer Replication – Combining Sharding and Replication
Consistency:
Update Consistency - Read Consistency - Relaxing Consistency - The CAP Theorem - Relaxing Durability.
Learning Outcomes: The student will be able to

- Analyze how multiple clients can interact with database server.
- Apply updating values dynamically in database.

UNIT-IV : **16 Hours**

Key-Value Databases:
What Is a Key-Value Store - Key-Value Store Features – Consistency – Transactions - Query Features - Structure of Data – Scaling - Suitable Use Cases - Storing Session Information - User Profiles, Preferences - Shopping Cart Data - When Not to Use - Relationships among Data - Multioperation Transactions - Query by Data - Operations by Sets.
Learning Outcomes: The student will be able to

- Use the way how Operations used in Real-time applications.
- Solve working with the database transactions.

UNIT-V : **20 Hours**

Document Databases:
What Is a Document Database? – Features – Consistency – Transactions – Availability - Query Features – Scaling - Suitable Use Cases - Event Logging - Content Management Systems - Blogging Platforms - Web Analytics or Real-Time Analytics - E-Commerce Applications - When Not to Use - Complex Transactions Spanning Different Operations - Queries against Varying Aggregate Structure.
Introduction to MongoDB:
Introduction to MongoDB - The Data Model - Working with Data – GridFS.
Learning Outcomes: The student will be able to

- Integrate database server like MongoDB to cloud apps.
- Test Cloud data storage.

Text Book:

1. Pramod J.Sadalag and Martin Fowler," *NoSQL Distilled, A Brief Guide to the Emerging World of Polyglot Persistence*" ,1st Edition, Addison Wesley
2. David Hows, Eelco Plugge, Peter Membrey , and Tim Hawkins, "*The definitive guide to MongoDB"*, "*A complete guide to dealing with big data using MongoDB".* 1st Edition, Apress

Reference Books:

1. Shashank Tiwari, Professional NoSQL, Wrox Press, Wiley, 2011, ISBN: 978-0-470-94224-6
2. Gaurav Vaish, Getting Started with NoSQL, Packt Publishing, 2013.

Web Resources:

1. http://allvideolectures.com/courses/course/96uv57kBOZ.
2. https://university.mongodb.com/

DATA WAREHOUSING AND DATA MINING	
CSE 322(D)	Credits : 3
Instruction : 3 Periods /Week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:

- Database management systems concepts, Probability and statistics

Course Objectives:

- To know the data storage in data warehousing.
- To handle real world data to pre-processing
- The importance of data analysis through data mining.

Course Outcomes:

By the end of the course, the student will be able to:

1.	Extend the basics, challenging issues in Data Mining data warehousing.
2.	Focus on data pre-processing approaches and used in the data mining.
3.	Analyse association rule mining in various dimensional databases.
4.	Apply classification by using decision tree induction, Bayesian, back propagation and prediction methods for data analysis.
5.	Interpret categorization of major clustering methods.

CO-PO Mapping:

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

SYLLABUS

UNIT-1 **14 periods**

Data warehouse: Introduction to Data warehouse, Difference between operational database systems and Data warehouses. Differences between operational databases systems and data warehouses. **Multidimensional data model**: From Tables and spreadsheets to Data Cubes, stars, Snowflakes, and Fact Constellations schemas for Multidimensional databases. Examples for defining star, snowflake and fact constellation schemas. **Data Warehouse Architecture:** Steps for the design and construction of data warehouses. A three-tier data warehouse architecture. **From Data warehousing to data mining**: Data warehouse usage, from on-line analytical processing to online analytical mining.

Learning Outcomes: at the end of this unit student are able to

- Understand the data storage in the data warehouse and multidimensional model
- To know the architecture of data warehouse

UNIT-2 **14 periods**

Data Mining Introduction: Data mining-on what kind of data, Relational databases, data warehouses, transactional databases, advanced database systems and advanced database applications. Data mining functionalities, classification of data mining systems, Major issues in data mining.

Data Pre-processing: Data cleaning: Missing values, Noisy data, inconsistent Data, Data Integration and Transformation: Data Integration, Data transformation Data Reduction: Data cube aggregation, dimensionality reduction, data compression, Numerosity reduction.

Learning Outcomes: at the end of this unit student are able to

- Motivate the importance of data mining, its functionalities and issues
- Find the importance of data pre-processing methodologies

UNIT-3 **12 periods**

Association Rule mining in Large Databases: Association rule mining , mining single-dimensional Boolean association rules from transaction databases, Mining multilevel association rules from transaction databases. Mining multidimensional association rules from relational databases. From association mining to correlation analysis. Constraint based association mining.

Learning Outcomes: at the end of this unit student are able to

- Generate association rules to given data and analyse market basket analysis
- Analyse association mining to correlation analysis.
- Understand the constraint based association analysis.

UNIT-4 **12 Periods**

Classification and Prediction: Issues regarding classification and prediction, Classification by decision tree induction, Bayesian classification, Classification by back propagation, Prediction, classification accuracy.

Learning Outcomes: at the end of this unit student are able to

- Understand the importance of classification and prediction in data analysis.

- Apply classification and regression techniques in real time data for analysis
- Able to find the accuracy of the classification and regression methods

UNIT-5 **10 Periods**

Cluster Analysis: Types of data in cluster analysis, a categorization of major clustering methods, Partition based methods: K-means, K-mediods. Hierarchical methods: BIRCH, CURE Density-based methods: DBSCAN.

Learning Outcomes: at the end of this unit student are able to

- Understand the various types of clustering methods
- Understand the difference between classification and clustering.
- Apply and analyse various types of clustering methods to real time data set.

Text Book:

1. Data Mining Concepts and Techniques, Jiawei Han and Micheline Kamber, Morgan Kaufman Publications, 2nd edition

Reference Books:

1. Introduction to Data Mining, Adriaan, Addison Wesley Publication , 2nd edition
2. Data Mining Techniques, A.K.Pujari, University Press
3. Introduction to data mining and soft computing- Dr.M.Ramarkrishna Murty , University science press. 1st edition.

E-Resources :

1. https://www.coursera.org/specializations/data-mining
2. https://www.udemy.com/course/data-mining/

DISTRIBUTED OPERATING SYSTEMS	
CSE 323(A)	**Credits : 3**
Instruction **:** 3 Periods /Week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:

- Fundamentals of Operating Systems and Computer networks and protocols.

Course Objectives:

- To provide an overview of the concepts of distributed operating systems and challenges that includes Architecture and Fundamental Models.
- To explore about various types of communication procedures and protocols in a distributed operating systems environment.
- To interpret the concept of communication between distributed objects and remote procedural calls.
- To analyse and understand the concepts of Distributed File system.
- To demonstrate the idea of Transactions and Replications in distributed operating system.

Course Outcomes:

By the end of the course student will be able to:

1.	Analyze the system model, software layers of distributed operating systems and its challenges.
2.	Examine the inter-process communication, TCP stream communication procedures and protocols.
3.	Evaluate the concepts of Remote procedural calls and communication among objects in distributed operating system.
4.	Apply the knowledge of peer-to-peer system, distributed mutual exclusion of distributed file system in real world scenario.
5.	Apply concurrency control, deadlock management techniques in distributed operating system for group communication.

CO-PO Mapping:

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

SYLLABUS

UNIT-I: **12 periods**

Characterization of Distributed Systems: Introduction, Examples of Distributed Systems, Resource Sharing and the Web, Challenges. System Models: Introduction, Architectural Models- Software Layers, System Architecture, Variations, Interface and Objects, Design Requirements for Distributed Architectures, Fundamental Models- Interaction Model, Failure Model, Security Model.

Learning Outcomes: At the end of this unit, Students are able to

- Explore the concepts of Distributed Systems.
- Analyse the Various System models in distributed systems.

UNIT-II: **12 periods**

Inter process Communication: Introduction, The API for the Internet Protocols- The Characteristics of Inter process communication, Sockets, UDP Datagram Communication, TCP Stream Communication; External Data Representation and Marshalling; Client Server Communication; Group Communication- IP Multicast- an implementation of group communication, Reliability and Ordering of Multicast.

Learning Outcomes: At the end of this unit, Students are able to

- Analyse the concept of Inter process communication.
- Compare the UDP datagram communication and TCP stream communication.

UNIT-III: **10 periods**

Distributed Objects and Remote Invocation: Introduction, Communication between Distributed Objects- Object Model, Distributed Object Modal, Design Issues for RMI, Implementation of RMI, Distributed Garbage Collection; Remote Procedure Call, Events and Notifications, Case Study: JAVA RMI

Learning Outcomes: At the end of this unit, Students are able to

- Determine the design issues and implementation of RMI.
- Summarize the concept of distributed garbage collection and remote procedure call.

UNIT-IV: **12 periods**

Distributed File Systems: Introduction, File Service Architecture; Peer-to-Peer Systems: Introduction, Napster and its Legacy, Peer-to-Peer Middleware, Routing Overlays. Coordination and Agreement: Introduction, Distributed Mutual Exclusion, Elections, Multicast Communication.

Learning Outcomes: At the end of this unit, Students are able to

- Demonstrate the architecture of file systems and build knowledge on peer-to-peer systems.
- Identify the various algorithms of Distributed mutual exclusion

UNIT-V: **12 periods**

Transactions & Replications: Introduction, System Model and Group Communication, Concurrency Control in Distributed Transactions, Distributed Dead Locks, Transaction Recovery; Replication-Introduction, Passive (Primary) Replication, Active Replication.

Learning Outcomes: At the end of this unit, Students are able to

- Determine the concurrency control in distributed transactions.
- Explore the concept of active and passive replication

Text Books:

1. Ajay D Kshemkalyani, Mukesh Sighal, "Distributed Computing, Principles, Algorithms and Systems", Cambridge.
2. George Coulouris, Jean Dollimore, Tim Kindberg, "Distributed Systems- Concepts and Design", Fourth Edition, Pearson Publication.

Reference Books :

1. Advanced Concepts in Operating Systems, Makes Singhal and Niranjan G.Shivaratna, Tata McGraw Hill Edition.

Web Resources:

1. https://www.coursera.org/learn/distributed-programming-in-java
2. https://www.edx.org/course/javacheng-xu-she-ji-java-programming-pekingx-04830340x
3. https://www.coursera.org/courses?languages=en&query=java

SMART SYSTEM DESIGN & PROGRAMMING	
CSE 323(B)	**CREDITS: 3**
Instruction: 3 Theory / Week	**Sessional Marks: 40**
End Exam: 3 Hours	**End Exam Marks: 60**

Prerequisites:

- Basic knowledge of Microprocessor & Interfacing, Computer Organization, Digital logic circuits
- Students must have knowledge of the C programming language.

Course Objectives:

- To learn the design and programming of microcontrollers.
- To learn the basics of ARM processors.
- To learn to program using ARM assembly language.
- To familiarize the students with Arduino kit and Raspberry Pi to implement small scale embedded system applications.

Course Outcomes:

By the end of the course, student will be able to	
CO-1:	To describe the Embedded system fundamentals, design and memory management.
CO-2:	To write programs in ARM based assembly level language.
CO-3:	To design Embedded system applications.
CO-4:	To test and debug embedded system applications.
CO-5:	To develop applications on Arduino and Raspberry Pi kits.

Mapping of Course Outcomes with Program Outcomes:

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

SYLLABUS

UNIT-I **10-12 hours**

Introduction to Embedded Systems - Application domain of embedded systems, Desirable features and general features, Figures of merit, classification of MCUs.
Hardware Point of View - Microcontroller Unit, Memory for embedded systems.
Examples – Mobile phone, Automotive electronics, RFID, WISENET, Robotics, Biomedical applications, Brain machine interface.
Learning Outcomes – At the end of this unit, students are able to

- Describe about the Embedded systems and their classifications.
- identify the different micro controller units used in verity of embedded systems.

UNIT-II **10-12 hours**

Hardware Software Co-design and Embedded Product Development Lifecycle Management – Hardware Software Co-design, Modeling of systems, Embedded product development lifecycle management, Lifecycle models.
Embedded Design: A Systems Perspective – A typical example, Product design, The design process, Testing, Bulk manufacturing.
Learning Outcomes – At the end of this unit, students are able to

- Extract information about the embedded product development lifecycle through different models.
- Classifies the design process embedded products and manufacturing.

UNIT –III **15-20 hours**

ARM Architecture and Assembly Language Programming – History, Architecture, Interrupt vector table, Programming, ARM Assembly language, ARM instruction set, Conditional execution, Arithmetic, logical & compare instructions, Multiplication, Division, Starting ALP, General structure of an Assembly Language Line, Writing ALP, Branch instructions, Loading Constants, Load and Store instructions.
Learning Outcomes – At the end of this unit, students are able to

- Develop the simple ARM assembly language programs using instruction set.
- Analyze the programs developed by using branching and comparing.

UNIT-IV **10-12 hours**

Introduction to Arduino : What Is Physical Computing?. The Arduino Way, The Arduino Platform, Really Getting Started with Arduino. Advanced Input and Output. Troubleshooting. **Case study:** Automatic Garden-Irrigation System. .
Learning Outcomes – At the end of this unit, students are able to

- Describes how to compute with Arduino platform and troubleshooting.
- Demonstrate the Arduino systems with some real time applications.

UNIT-V **15-20 hours**

Introducing the Raspberry Pi: The History of Raspberry Pi, Exploring the Pi Board, Hardware Requirements of the Pi, The Pi Operating System, Connecting the Peripherals, Configuring the Pi, Getting Started with Python, Accessing the GPIO Pins, Using the GPIO Library in python, Connecting the Temperature/Humidity Sensor, Setting Up the Motion Sensor. **Case Study**: Weather Station.

Learning Outcomes – At the end of this unit, students are able to

- Recognize and explores Raspberry Pi boards and its functionality when connecting with peripherals.
- Summarize accessing the GPIO pins and its library in python and can analyze different sensors.

Text Books:

1. Das, Lyla B, *Embedded Systems: An Integrated Approach,*Pearson Education India, 2013.
2. Donat, Wolfram, *Learn Raspberry Pi Programming with Python*, Apress, 2014.
3. Banzi, Massimo, and Michael Shiloh, *Getting Started with Arduino: The Open Source Electronics Prototyping Platform*, Maker Media, Inc., 2014.

Reference Books:

1. Hohl, William, and Christopher Hinds. *ARM Assembly Language: Fundamentals and Techniques*, Crc Press, 2016.
2. Monk, Simon, *Raspberry Pi cookbook: Software and hardware problems and solutions,* O'Reilly Media, Inc., 2016.
3. Simon Monk, *30 Arduino™ Projects for the Evil Genius*, The McGraw-Hill Companies.

Web Resources:

1. http://nptel.ac.in/syllabus/117106111/
2. Muhammad Ali Mazidi, *ARM Assembly Language Programming & Architecture,* Kindle edition

Proposal for Introduction of New Industry Elective in Engineering Curriculum

"Machine learning using R"

Designed in collaboration with
Infosys Limited

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY & SCIENCES
(UGC AUTONOMOUS)
(Affiliated to AU, Approved by AICTE & Accredited by NBA)
SANGIVALASA-531 162, Bheemunipatnam Mandal, Visakhapatnam District
31-07-2021

Contents

7. Background

ANIL NEERUKONDS INSTITUTE OF TECHNOLOGY & SCIENCES(A)has partnered with Infosys Limited to roll-out Campus Connect Program. Under this program we have been conducting training leveraging IT Industry-Ready program for CS/IT/MCA students. Our faculty was enabled in delivering these courses.

Infosys is willing to extend the relationship with our college by collaboratively designing a new industry elective **Machine Learning using R.** The purpose of this proposal is to describe the contents of the new elective, its benefits and seek approval to start the elective offering effective **Aug 2021**.

8. Objectives

- Understand and Appreciate why data science is gaining importance in today's business world.
- Comprehend where data science can be applied in different scenarios across industry domains.
- Understand the fundamentals of R language and its usage for statistical computing
- Identify various data importing, manipulation, visualization techniques in R and perform exploratory data analysis
- Recognize various machine learning techniques such as classification, regression etc. across several use cases

9. Overview of the Course Design

3.1 Synopsis:

The proposed elective course exposes the B.E/B.Tech/M.E/M.Tech students to "Machine Learning using R". The core modules of this elective include introduction to R Programming, Exploratory Data Analysis, Machine Learning and Deep Learning Techniques.

3.2 Prerequisites:

- **Probability, Statistics, Introduction to computer programming, Linear Algebra**

3.3 Assumptions:

7. This elective will be applicable to B.E/B.Tech/M.E/M.Tech students
8. The duration of the course will be One Semester
9. The elective design follows University Curriculum standards
10. There will be a compulsory final Examination
11. The elective will be designed in exclusive collaboration with Infosys
12. The college will leverage existing Lab & IT infrastructure

10. Learning outcomes

By the end of the course, the student will be able to:	
1.	Realize the fundamentals of machine learning.
2.	Recognize the strengths and weaknesses of many popular machine learning techniques.
3.	Apply various machine learning algorithm on real time problem.
4.	Formulate machine learning problems corresponding to different applications
5	Explore advanced Machine Learning techniques like support Vector Machines and Deep Learning based CNN.

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

11. Course Schedule Summary (Illustrative only)

Here it is illustrated for one semester course.

Duration of the Course	Number of Weeks	Total Lecture hours	Total Tutorial hours	Total Practical hours	Total Credit
One semester	15 – 16 Weeks	3 hours per week	2 hour(s) per month	2.5 hour per week	3

12. Course contents (Draft only)

Syllabus

UNIT I: 10 Hour

Introduction to Machine Learning:
Understanding Data with Statistics, What is Machine Learning? Why Machine Learning? Concept of Learning, Types of Machine Learning: Supervised Machine Learning, Unsupervised Machine Learning, Semi-supervised Machine Learning, Reinforcement Machine Learning, Industrial applications of Machine Learning across domains such as Healthcare, Finance, Retail etc.

Learning outcome: at the end of this unit student are able to

1. Understand different types of machine learning techniques and application of AI.
2. To know the issues in machine learning.

UNIT II: 10 Hours

R Objects, Data Handling:
Introduction to R, Why to learn R, Object, Vector, List, Factor, Matrix, Array, Data Frame, Manipulating Objects, Input/output, R constructs, Useful R Packages.
Learning outcomes: at the end of this unit student are able to
1. Analyze the fundamentals of statistical analysis in R environment.
2. Implement Probability and Probability Distributions to solve a wide variety of problems.

UNIT III: 12 Hours

Understanding Data with Visualization
Data Visualization techniques, variance, standard deviation, shape – skewness, kurtosis, percentiles, five point summary, boxplots, histograms, barplot, pie chart, scatter plot, two way tables, covariance, correlation, Chi-Square test for two way tables.
Learning outcomes: at the end of this unit student are able to
1. Recognize the concepts and techniques employed in Statistical Analysis.
2. Visualize the data using various data visualization techniques.

UNIT IV: 14 Hours

Clustering, Classification, Regression.
What is Clustering? Applications of Clustering, K means clustering, k-medoid, Hierarchical clustering, Density based clustering, What is Regression, Simple Linear Regression, Multiple Linear Regression, What is Classification? Logistic Regression, Decision Tree, k-Nearest Neighbors, Support Vector Machine.
Learning outcomes: at the end of this unit student are able to
1. Analyze clustering, classification and regression algorithm.
2. Examine various algorithms related to problem solving

UNIT V: 12 Hours

Neural Networks
Introduction to Neural Networks, Activation functions, Learning rate, Stochastic Gradient Descent, Feed forward, Back propagation, Basics of Deep Learning Networks, Convolutional Neural Networks (CNN).
Learning outcomes: at the end of this unit student are able to
1. Understand the importance of deep learning in neural network.
2. Apply various deep learning techniques in real time data for analysis.
3. Compare the accuracy of traditional ML with deep learning methods.

TEXT BOOK:

1. Machine Learning ,Tom M. Mitchell, Mc Graw Hill, 1997.
2. R for Data Science, By Hadley Wickham, Garrett Grolemund, Shroff/O'Reilly, First edition (4 February 2017).
3. Deep Learning with R, By Francois Chollet, J. J. Allaire, Manning Publications; 1st edition (9 February 2018).

REFERENCE BOOK:

1. Machine Learning, An Algorithmic Perspective, Stephen Marsland, Taylor &Francis (CRC)
2. Advanced R By Hadley Wickham, Chapman and Hall/CRC, 1st edition (28 October 2014)

3. Introduction to Machine Learning, Ethem Alpaydin, PHI, 2004

13. Tutorial/Optional Assignments

The purpose of 2 hour tutorial per month is to help the students to explore points outside the prescribed material and to enhance their learning. The assignments for elective could include the following.

- Seminars from the topics related to Machine Learning applications
- Relevant lab exercises to get exposure to various tools such as R.

Sl No	Tool Name	Usage Area / Scope
1	R, RStudio	R is the most popular tools used by data scientists. R is open-source and freely available and was designed for statistical analysis. R becomes the right choice who wants to gain a better understanding of the underlying details and build innovative. R is suitable for data analysis and visualization. Similar to Python, R comprises of multiple packages which help to improve the performance of the machine learning projects. And also R is Suitable for exploratory work.
2	Python	If we need an adaptable and multi-reason programming language with a supporting enormous network of engineers alongside the extendable AI bundles then Python is a top pick.

Optional Assignments:

Case study on IRIS Data sets

Implement well-known machine learning algorithm called “KNN” or *k*-nearest neighbors using R tools.

1. Get the Data
2. Know the Data
3. Prepare the Workspace
4. Prepare Data
5. The Actual KNN Model
6. Evaluation of the Model

8. Infrastructure Requirements

HARDWARE / SOFTWARE REQUIREMENTS

Machine:

Pentium P5, 3 GHz or higher
8 GB (or higher) RAM, 100 GB (or higher) HD
Windows 8 or 10 (or higher)

Software required for Tutorials and Practical:

Sl. No	Course	S/W on Students Machine	Remarks
1.	Machine Learning using R	R	

9. Mode of Examination:

- The Institute will conduct all the assessments. The examination carries 100 Marks.
- Internal assessments carry 40 Marks which includes two descriptive tests, assignments and quiz and Semester-End Exam Carries 60 Marks.

10. Faculty enablement

The Faculty will be enabled on the course contents; Industry practices case studies etc. for duration of one week before the commencement of elective. Faculties from various colleges are required to stay in the Infosys Campus for their Enablement.

11. Courseware & reference books:

The courseware including PowerPoint is available for the Elective. In addition, following reference book can also be used:

- Text Books
  - Practical Data Science with R. Author(s): Nina Zumel, John Mount, Manning Shelter Island
  - Data Mining Concepts and Techniques, 3rd Edition. Author(s): J.Han, M Kamber, J Pei
  - Introduction to Data Mining. Author(s): Pang-Ning Tan, Steinberg, Vipin Kumar
  - Introduction to Statistical Learning using R. Author(s): Trevor Hastie, Tibshirani
  - Applied Predictive Modeling. Author(s): by Max Kuhn, Kjell Johnson

12. Actions:

6. The college needs to send the Board of Studies Approval letter on college letter head to Infosys.
7. Identify one department to own the responsibility of course content, assignments, projects, software tools etc. (Preferable CS/IS/EC/MCA Department)
8. Identify faculty from CS/IS/EC/MCA department for rollout and faculty training
9. Identify and allocate resources like classrooms, labs, necessary hardware and software for rollout.

10. Complete readiness check before the rollout

13. Contact Details:

The Infosys point of contact can be reached for more info. In addition, the Institute SPoC can also be reached for additional info.

Department owning the responsibility of Course Content:

The HOD'S / Faculty Names and their Email Id, owning the course content of Elective are to be mentioned.

S. No.	Name	E-Mail	Phone Number
1	Dr. Sivaranjani Reddy (HOD-CSE)	hod_cse@anits.edu.in	9642022170
2	Mr. S. Ratan Kumar (SPOC)	Sratankumar.cse@anits.edu.in	9052492777
3			
4			
5			

Faculties handling the Elective rollout:

The faculty names and their Email Id, handling the Elective rollout are to be mentioned.

S.No.	Name / Dept.	E-Mail	Phone Number
1	Mr. S.R.Mishra	Srmishra.cse@anits.edu.in	8096177170
2			
3			
4			
5			

14. Conclusion:

Introduction of the collaboratively designed elective will significantly help the students to be industry aligned and leverage IT as a competitive edge in their career while working in their own discipline or specialization.

Hence, we request for approval the introduction of this elective

COMPUTER VISION	
CSE 323(D)	**Credits : 3**
Instruction **:** 3 Periods /Week	Sessional Marks : 40
End Exam : 3Hours	End Exam Marks : 60

Prerequisites:

- Basic Knowledge of computer graphics and image processing.

Course Objectives:

- To understand light and shading effects
- To understand filtering and texture techniques
- To understand the use of clustering techniques & models for segmentation
- To understand fitting techniques

Course Outcomes:

Upon completion of the course, the students will be able to

By the end of the course, the student will be able to:	
1.	Study about measuring light, shadows and shading effects
2.	Filtering techniques, edge detection methods, texture techniques
3.	Perform segmentation using clustering techniques
4.	Perform segmentation using models
5.	Perform fitting using classifiers

Mapping of Course Outcomes with Program Outcomes:

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

SYLLABUS

UNIT-1 **9 Hours**

RADIOMETRY-MEASURING LIGHT: Light in Space, Light at Surfaces, Important SpecialCases.
SOURCES, SHADOWS, AND SHADING: Qualitative Radiometry, Sources and Their Effects, Local Shading Models, Application: Photometric Stereo, Inter reflections: Global Shading Models.
Learning Outcomes: At the end of this unit, Students are able to

- Interpret behaviour of light and basic definitions of light
- Describe the effects that result when surfaces reflect light onto one another.

UNIT-2 **13 Hours**

LINEAR FILTERS: Linear Filters and Convolution, Shift Invariant Linear Systems, Spatial Frequency and Fourier Transforms, Sampling and Aliasing, Filters as Templates, Technique: Normalized Correlation and Finding Patterns, Technique: Scale and Image Pyramids.
EDGE DETECTION: Noise, Estimating Derivatives, Detecting Edges.
TEXTURE: Representing Texture, Analysis using Oriented Pyramids, Application: Synthesizing Textures for Rendering, Shape for Texture for Planes.
Learning Outcomes: At the end of this unit, Students are able to

- Describe the Extraction of useful information from the images
- Explain the edge detection techniques, representation of texture, analysis of pyramids and application of texture based manipulations.

UNIT-3 **13 Hours**

SEGMENTATION BY CLUSTERING: What is Segmentation, Human Vision: Grouping and Gestalt, Applications: Shot Boundary Detection and Background Subtraction, Image Segmentation by Clustering Pixels, Segmentation by Graph-Theoretic Clustering.
Learning Outcomes: At the end of this unit, Students are able to

- Describe the Image interpretation through segmentation techniques
- Explain the Portioning and grouping based image clustering.

UNIT-4 **13 Hours**

SEGMENTATION BY FITTING A MODEL: The Hough Transform, Fitting Lines, Fitting Curves, Fitting as Probabilistic Inference Problem, Robustness, Example: Using RANSAC to Fit Fundamental Matrices, Missing Data Problems, the EM Algorithm.
Learning Outcomes: At the end of this unit, Students are able to

- Differentiate the different image segmentations
- Describe the need of segmentation in terms of missing data problems and outliers.

UNIT-5 **12 Hours**

FINDING TEMPLATES USING CLASSIFIERS: Method for Building Classifiers, Building Classifiers from Class Histograms, Feature Selection, Neural Networks, the Support Vector Machine.
Learning Outcomes: At the end of this unit, Students are able to

- Explain different types of image classification
- Describe the proper selection of classification techniques.

TEXT BOOK:

1. David A.Forsyth, Jean Ponce, Computer Vision-A Modern Approach, PHI, 2003.

REFERENCES:

1. Geometric Computing With Clifford Algebras: Theoretical Foundations and Applicationsin Computer Vision and Robotics , Springer; 1/ e,2001 by Sommer.
2. Digital Image Processing and Computer Vision, 1/e, by Sonka.
3. Computer Vision and Applications: Concise Edition(With CD) by Jack, Academy Press, 2000.

OBJECT ORIENTED SOFTWARE ENGINEERING	
Course Code: CSE324	**Credits : 3**
Instruction: 3 Periods & 1 Tut/Week	Sessional Marks: 40
End Exam: 3 Hours	End Exam Marks: 60

Prerequisites:

- Basic Mathematical Knowledge.
- Basic knowledge on procedural and object oriented programming.
- Basic knowledge on problem solving.

Course Objectives:

The course should enable the students:

- To Explain the importance of OOSE in Software development.
- To Provide knowledge on software Life Cycle and Development Models with object oriented concepts.
- To Explain the role of UML and Testing in Software Development.
- To Plan and manage the development of software projects.

Course Outcomes:

By the end of the course, the student will be able to:	
1	Analyze the different software process models and their significance.
2	Interpret the functional, non–functional requirements and requirement Engineering Process.
3	Choose the Architecture for a given software application & Design UML diagrams.
4	Demonstrate skills in Object-oriented Modelling and Plan software project management activities.
5	Identify the Testing Strategies and design test suits for the given Test Scenarios.

Mapping of Course Outcomes with Program Outcomes:

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

SYLLABUS

UNIT-I **(10 Hours)**

Importance of Software Engineering: Problem Solving Activity, Modelling Activity, Knowledge Acquisition activity, Rationale Driven Activity, Umbrella Activities , Software Engineering Challenges ,Software Development Life Cycle.
Process Models: Waterfall Model, Incremental Model, Prototype Model, V Model, Spiral Model, Rapid Model, Agile Model.
Learning Outcomes: At the end of this unit, Students are able to

- List Various Software Development activities.
- Classify the Working of Software Process Models.

UNIT-II **(14 hours)**

Software Requirement Analysis & Specification: Need for SRS, Characteristics of Requirements, Functional Requirements, Non Functional Requirements, Requirement Specification Document -IEEE Format.
Function Oriented Design: Structured Design Methodology –DFD Diagram, Design Principles, Module-Level Concepts- Cohesion & Coupling.
Learning Outcomes: At the end of this unit, Students are able to

- Find Functional and Non-functional Requirements and prepare SRS Document..
- Design Data flow in Functional Approach (DFD).

UNIT-III **(18 Hours)**

Architectural Design: Role of Software Architecture, Architecture Views, Architectural Styles for C & C View- Pipe and Filter, Client and Server, Shared Data.
Object Oriented Design: OO Concepts, Importance of Modelling.
Overview of UML: Building Blocks of UML (Things, Relationships, Diagrams), Class Diagram, Class Relationships, Design Axioms, Corollaries, Use Case- Use Case Scenario, Use Case Diagram, Relationships among Use Cases.
Interaction Diagrams: Sequence Diagram, Collaboration Diagram.
Learning Outcomes: At the end of this unit, Students are able to

- Make use of Various Software Architectures.
- Design UML Diagrams with Object Oriented Approach.

UNIT – IV **(15 Hours)**

UML Diagrams: Activity Diagram, State Chart Diagram, Component Diagram & Deployment Diagram.
Object Oriented Methodologies: Unified Methodology, Rambaugh Methodology, Booch Methodology.
Software Project Management: Definition of Software Project, Need of Software Project Management, Software Project management Plan, Effort Estimation ,Case Study on Effort Estimation with COCOMO, Schedule - Overall and detailed scheduling, Risk Management Plan, Project Tracking & Control.
Learning Outcomes: At the end of this unit, Students are able to

- Illustrate Various object oriented Methodologies.
- Adapt the approaches in Software Project Management to Develop the Software

Project.

UNIT-V **(10 Hours)**

Testing: Testing Fundamentals, Testing Process, Levels of Testing ,Test Plan, Test Case Specification, Test Case Execution & Analysis, Defect Logging & Tracking, Black Box Testing, White Box Testing.

Learning Outcomes: At the end of this unit, Students are able to

1. Summarize various Software Testing Techniques.
2. Design Test Suites for Test Scenarios.

Text Books:

1. Timothy C. Lethbridge, "Object Oriented Software Engineering" (Practical Software Development using UML and Java" Tata McGraw-Hill, 2nd Edition, 2019.
2. Booch, Maksimchuk, Engle, Young, Conallen and Houston, "Object Oriented Analysis and Design with Applications ", Pearson Education, 3rd Edition, 2009.
3. Pankaj Jalote, An integrated Approach to Software Engineering, Springer, 3rd edition, 2005.

Reference Books:

1. Ivar Jacobson, "Object Oriented Software Engineering", Pearson, 2009.
2. Rumbaugh et. al, "Object Oriented Modeling and Design", Pearson.
3. Bertrand Meyer, *Object-Oriented Software Construction*, Prentice Hall, 2nd edition, 1998.
4. Edwards Yourdon, Carl Argila,"Case Studies in object oriented analysis and design" Prentice Hall.

WEB TECHNOLOGIES	
CSE 325	**CREDITS : 3**
INSTRUCTION : 3 Periods & 1 Tut/Week	**SESSIONAL MARKS : 40**
END EXAM : 3 Hours	**END EXAM MARKS : 60**

Prerequisites:

- Basic Knowledge of Programming Fundamentals
- Knowledge of Programming Languages (C,JAVA,SQL)

Course Objectives:

The course should enable the students:

- To learn designing of dynamic and interactive web pages by embedding Java Script code in HTML.
- To know how to design and to develop simple database driven web applications using a server-side scripting language –PHP.
- Better understanding of developing web applications using advanced java.
- Understanding with database technology and middleware technologies.
- Making the web applications using Node JS and Express JS.

Course Outcomes:

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

1.	Design the Webpage and templates.
2.	Develop web applications using PHP and Mysql database Technology.
3.	Implement web applications using JSP.
4.	Create data Interchangeable techniques between Client to server
5.	Design web applications using Node JS and Express JS.

CO-PO Mapping:

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

SYLLABUS

UNIT 1: **16 Periods**

HTML Common tags: Basics of HTML5, formatting and fonts, Comments, color, hyperlink, lists, tables, images, forms, frames, CSS-syntax, Types, Examples **Java Script**: Introduction to Java Scripts, Objects in Java Script, Dynamic HTML with Java Script.

Learning Outcomes: At the end of this unit the Students will be able

- To design webpages and validation of webpages.
- To Identify software that can be used to create, maintain, or modify HTML and CSS

UNIT 2: **12 Periods**

PHP: Introduction and basic syntax of PHP, decision and looping with examples, PHP and HTML, Arrays, Functions, Browser control and detection, string, Form processing, Files, Advance Features: Cookies and Sessions, Object Oriented Programming with PHP.

Learning Outcomes: At the end of this unit the Students will be able

- To develop the PHP scripts to handle HTML forms.
- To Create PHP programs that use various PHP library functions, and that manipulate files and directories.

UNIT 3: **12 Periods**

JSP Application Development: Introduction to JSP, Difference between Servlets and JSP, Types of Scripting Elements, Implicit JSP Objects, Error Handling and Debugging, Sharing Data Between JSP pages, Session Tracking and Application Data- Working with Java Beans, Servlets and JSPs.

Learning Outcomes: At the end of this unit the Students will be able

- To create dynamic web pages, using Servlets and JSP
- To develop proficiency in creating web based applications MVC architecture.

UNIT 4: **12 Periods**

Database Access: Database Programming using JDBC, Studying Javax.sql.* package, Accessing a Database from JSP Page , Application – Specific Database Actions.

MongoDB: Installation, Connection, Database, Collection, Operations, Modelling, Administration, Security, Aggregation, Indexes.

Learning Outcomes: At the end of this unit the Students will be able

- To learn how to store and access data from database and dynamic database activities.
- To learn how to run queries against a MongoDB instance in order to store, manipulate, and retrieve data on it.

UNIT 5: **12 Periods**

NodeJS :Basics and Setup, REPL terminal, NPM and Command Utilities, Global Objects, Modules – (OS, Path, DNS, Net, Domain),File System.

ExpressJS :Environment, Routing, Http Methods, URL Binding, Middleware, Templating, Static files, Form data.

Learning Outcomes: At the end of this unit the Students will be abl

- To Develop interactive Web applications or single page applications using Node JS and Express .
- To apply the Node.js MongoDB driver in order to manipulate our data directly from Node.js.

TextBooks:

1. Thomas A. Powell, - HTML & CSS Complete Reference ,Mc Graw Hill, Fifth Edition,2017
2. Dietel and Nieto PHI/Pearson Education Asia., "Internet and World Wide Web – How to program ",4thedition,PearsonEducation.
3. Bruce W. Perry,Java Servlet & JSP Cookbook, O'Reilly Media, Inc Released January 2004,ISBN: 9780596005726.
4. Kristina Chodorow, Michael Dirolf MongoDB: The Definitive Guide,Released Sep 2010,O'Reilly
5. Ethan Brown,Web Development with Node and Express: Leveraging the JavaScript Stack by 2nd Edition, O'Reilly Media, Inc.

Reference Books:

1. Kogent Learning Solutions Inc. Web Technologies: HTML, JAVASCRIPT, PHP, JAVA, JSP, ASP.NET, XML and Ajax, Black Book, Dreamtech Press; 1st edition (1 January 2009)
2. Kyle Banker, MongoDB in Action, Manning, Second Edition, March 2016.

Web Resources:

1. https://nptel.ac.in/courses/106/105/106105084/
2. https://www.coursera.org/learn/html-css-javascript-for-web-developers
3. https://www.coursera.org/learn/server-side-nodejs

CRYPTOGRAPHY AND NETWORK SECURITY	
CSE 326	**Credits : 3**
Instruction **:** 3 Hours & 1 Tut/Week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:

- Computer networks
- Number theory
- Probability theory

Course Objectives:

The course should enable students to

- Acquire the basics of Cryptographic security.
- Learn the working of the cryptographic algorithms for confidentiality, authentication and integrity.
- Learn the different techniques for distributing public and private keys.
- Understand how the different algorithms are used in different protocols to provide security in network

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Memorize the foundational concepts of Cryptographic systems
2.	Develop the applications of cryptographic algorithms in Network Security
3.	Demonstrate the algorithms to achieve the security goals of Confidentiality, Authentication and Integrity to a given application
4.	Synthesize the different algorithms to determine protocol implementation and achieve Authentication.
5.	Examine the techniques of Intrusion Detection systems and classify the types of Firewalls.

Mapping of Course Outcomes with Program Outcomes:

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

SYLLABUS

UNIT 1: INTRODUCTION TO CRYPTOLOGY 14 Hours

Cryptography, Need for Security, Security Goals, Security Methodology, OSI Security Architecture: Threats-Attacks & Attack Types-Services-Mechanisms, Network Security Model: Plain Text-Cipher Text-Encryption-Decryption-Key, Key Range and Key Size, Classic Cryptography: Substitution-Transposition, Steganography.
Basic Concept of Symmetric Cryptography, Algorithm Types and Modes, Principles of Public-Key Cryptography.
Learning Outcomes: At the end of this unit ,student will be able to

- Explore the concepts of Security.
- Analyse the importance of Cryptography algorithms.

UNIT 2: CONFIDENTIALITY 12 Hours

Symmetric Cryptography Techniques: Feistel Structure, DES-AES-RC4
Asymmetric Cryptography Techniques: Encryption/Decryption using RSA, Encryption/Decryption using Elliptic Curve Cryptography, Digital Envelope.
Learning Outcomes: At the end of this unit ,student will be able to

- Implement symmetric Cryptographic algorithms.
- Analyse the Asymmetric Cryptographic algorithms.

UNIT 3: KEY MANAGEMENT AND INTEGRITY: 12 Hours

Key Distribution And Management: RSA Key Exchange, Diffie-Hellman Key Exchange, Digital Certificates (public key), Private Key Management.
Hashing: Cryptographic Hash Function Definition, Applications of Cryptographic Hash Functions, Message Authentication Functions, SHA-256
Learning Outcomes: At the end of this unit ,student will be able to

- Interpret key distribution and management.
- Apply Hash Functions to achieve security goals.

UNIT 4: AUTHENTICATION 10 Hours

Authentication Using Asymmetric Cryptography (Digital Signatures): Basic Idea of Digital Signatures, RSA Digital Signature Scheme, Digital Signature Standard, Kerberos.
Learning Outcomes: At the end of this unit ,student will be able to

- Authenticate using Digital Signature implementation.
- Interpret RSA Digital signatures for achieving Authentication.

UNIT 5: NETWORK SECURITY: 12 Hours

Application Layer: PGP, S/MIME, Transport Layer: TLS, SSL, Network Layer: IP Security Intrusion Detection Systems (IDS): Types of IDS Technologies, False Positives and

Negatives, Intrusion Detection Techniques, Firewalls: Definition, Packet Filters, Circuit Level filters, Application Layer Filters.

Learning Outcomes: At the end of this unit ,student will be able to

- Analyse various Intrusion Detection Techniques.
- Formulate rules for providing security.

TEXTBOOK:

1. Cryptography and Network Security, Forouzan and Mukhopadhyay, 2nd edition, TMH.
2. Cryptography and Network Security: Principles and Practice, William Stallings, 5th edition, Pearson.

REFERENCES:

1. Cryptography and Network security, Atul Kahate, Tata McGraw-Hill Pub company Ltd., New Delhi
2. Network Security Private Communication in a public world, Charlie Kaufman, Radia Perlman & Mike Speciner, Prentice Hall of India Private Ltd., New Delhi.
3. Network Security: The Complete Reference, Robert Bregga, Mark Rhodes-Ousley, Keith Strassberg, TMH.

QA-II & VA	
CSE 327	**Credits : 1.5**
Instruction **:** 3 Periods /Week	Sessional Marks : 100

Quantitative Aptitude II
3/4 B. Tech, Semester II

Course Objectives:

• To categorize, apply and use thought process to distinguish between concepts of reasoning
• To prepare and explain the fundamentals related to various possibilities and probabilities related to quantitative aptitude.
• To critically evaluate numerous possibilities related to puzzles.

Course Outcomes:

The student will be able to

• Use their logical thinking and analytical abilities to solve reasoning questions from company specific and other competitive tests.
• Solve questions related to permutation & combinations and probabilities from company specific and other competitive tests.
• Understand and solve puzzle related questions from specific and other competitive tests.

UNIT I **8 Periods**

Numerical Reasoning:

Problems related to Number series, Analogy of numbers,), Letter series, Seating arrangements, Group Reasoning ,Directions, blood relations and puzzle test(figures).

UNIT II **4 Periods**

Modern Aptitude

Permutations, Combinations and Probability

UNIT III **4 Periods**

Venn diagrams, Syllogisms & high level logical deductions based on syllogisms and data sufficiency

UNIT IV **4 Periods**

Modern Reasoning :

Clocks (Base 24), Calendars (Base7), Counting (figure)techniques, Ranking Test ,Alphabet test

UNIT V **4 Periods**

Puzzle Solving & Time Management using various problems solving tools and techniques:

Selective puzzles from previous year placement papers Selective puzzles from book Puzzles to puzzle you by Shakunatala devi Selective puzzles from book more puzzles by Shakunatala devi Selective puzzles from book puzzles by George Summers Pseudo code

techniques for campus placement Inserting of numbers (high level)

Books for practice

1. Quantitative aptitude by R S Agarwal, S Chand Publications
2. Verbal and nonverbal Reasoning by R S Agarwal, S Chand publications
3. Puzzles to puzzle you by Shakunatala devi orient paperback publication
4. More puzzles by Shakunatala devi orient paperback publication
5. Puzzles by George summers orient paperback publication

References:

1. Barron'sbySharonWelnerGreenandIraKWolf(GalgotiaPublicationspvt.Ltd.)
2. Materialfrom'IMS,CareerLauncherandTime'institutesforcompetitiveexams.
3. ReasoningbyBSSijwaliArihantpublications
4. ReasoningArunSharmaMcGrawhillpublications

Websites:

1. www.m4maths.com
2. www.Indiabix.com
3. 800score
4. Official GREsite
5. Official GMATsite

VERBAL ABILITY		
Common for all branches	**Credits : 1.5**	**Sessional Marks: 50**

III/IV B.TECH
(W.e.f. 2019-20 admitted batch)

2 Period-Theory,

Course objectives:

- To explain principles of grammar in order to minimize errors in English.
- To list and quote high frequency words by giving relevant examples.
- To categorize, apply and use data as per the requirement.
- To make sentences using idioms, phrasal verbs and other expressions in professional contexts.
- To critically evaluate reading material for better comprehension

Course Outcomes:

At the end of the course, students will be able to:	
1	Detect grammatical errors in the text/sentences and rectify them while answering their competitive/ company specific tests and frame grammatically correct sentences while writing.
2	Answer questions on synonyms, antonyms, hyponyms, hypernyms and other vocabulary based exercises while attempting company specific and other competitive tests.
3	Use their logical thinking ability and solve questions related to reasoning based exercises.
4	Choose the appropriate word/s/phrases suitable to the given context in order to make the sentence/paragraph coherent.
5	Analyze the given data/text and find out the correct responses to the questions asked based on the reading exercises; identify relationships or patterns within groups of words or sentences.

Section –B

VERBAL ABILITY

UNIT I **6 Hours**

Parts of speech (appropriate application and usage of prepositions, co-relative conjunctions, pronouns- number and person, relative pronouns and degrees of comparison, articles (nuances while using definite and indefinite articles), tenses (appropriate application considering mood and aspect according to the situation), subject-verb agreement (to differentiate between number and person),clauses (use of the appropriate clause, conditional clauses)

UNIT II **6 Hours**

Voice, direct & indirect speech, question tags, modifiers (misplaced and dangling modifiers, absence of modifiers), determiners, parallelism in structure (symmetry in two part sentences), word order, subjunctive mood, redundancy, signpost words, miscellaneous types, identifying errors in a given sentence, correcting errors in sentences, homonyms, hyponyms, hypernyms, frequently confused words, words often mis-used, words often mis-spelt

UNIT III **6 Hours**

Synonyms and synonym variants (with emphasis on high frequency words), antonyms and antonym variants (with emphasis on high frequency words), phrases (use of the phrases, phrasal verbs), idiomatic language (with emphasis on business communication), foreign phrases, punctuation depending on the meaning of the sentence, run on errors, sentence fragments, coma splices

UNIT IV **6 Hours**

Critical reasoning (understanding the terminology used in CR- premise, assumption, inference, conclusion), , fact- inference-judgment (FIJ), sequencing of sentences (to form a coherent paragraph, to construct a meaningful and grammatically correct sentence using the jumbled text), to use logical reasoning and eliminate the unrelated word from a group, contextual meanings (multiple meanings of the same word (differentiating between meanings with the help of the given context), one word substitutes

UNIT V **6 Hours**

Reading comprehension, types of passages (to understand the nature of the passage), types of questions (with emphasis on inferential and analytical questions), style and tone (to comprehend the author's intention of writing a passage), strategies for quick and active reading(importance given to skimming, scanning), summarizing , reading between the lines, reading beyond the lines, techniques for answering questions related to vocabulary (with emphasis on the context), supplying suitable titles to the passage, identifying the theme and central idea of the given passages

Books for Practice:

1. Intermediate English Grammar Raymond Murphy, Cambridge University Press,
2. Practical English Grammar A. J. Thomson, A. V. Martinet by Oxford University Press.
3. Remedial English Grammar for Foreign Students by FT wood published by Macmillan Publishers.
4. Objective English-Edgar Torpe, Showick Thorpe-Pearson Education.
5. Word Power Made Easy by Norman Lewis-Goyal Publishers.
6. RS Agarwal's books on Objective English and Verbal Reasoning from S Chand Publications.

Reference Books and websites:

1. Barron's by Sharon Welner Green and Ira K Wolf (Galgotia Publications Pvt.Ltd.)
2. Longman, Cambridge and Oxford Dictionaries
3. Collins Co-build English Grammar by Goyal Publishers
4. Websites: Bankers' Adda , India bix, 800 score, official CAT, GRE and GMAT sites
5. Practical English usage, Michael Swan, OUP 1995
6. Remedial English Grammar, F.T.Wood.Macmillan, 2007

Applications:

1. Longman Dictionary
2. British Council for Testing
3. The Hindu for Reading)

******

OBJECT ORIENTED SOFTWARE ENGINEERING LAB	
CSE 328	**Credits : 1.5**
Instruction : 3 Periods /Week	Sessional Marks : 50
End Exam : 3 Hours	End Exam Marks : 50

Prerequisites:

- Basic Mathematical Knowledge
- Basic knowledge on procedural and object oriented programming
- Basic knowledge on problem solving.

Course Objectives:

The course should enable the students:

- To provide working knowledge of UML.
- To provide working knowledge of the technologies essentially for incorporating in the project.
- To expertise for testing and document software.
- To present the project in a professional manner.

Course Outcomes:

By the end of the course, the student will be able to:	
1	Design DFD, UML Diagrams for the specified software project
2	Write the Software Requirements Document for a specified project
3	Design Test Suites
4	Discuss about the project implementation among the team members and improve their professional skills

Mapping of Course Outcomes with Program Outcomes & PSOs:

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

List of Experiments

Experiment No.	Name of the Experiment	CO Number
1	Introduction and Project Definition	2 ,4
2	System Modelling –DFD Diagram	1,4
3	Introduction to UML and Use Case Scenario, Use Case Diagram	1,4
4	Object Oriented Analysis : Discovering Classes , Class Diagram	1,4
5	Interaction Diagrams: Sequence and Collaboration Diagrams	1,4
6	State Chart Diagram	1,4
7	Flow of Events and Activity Diagrams	1,4
8	Component and Deployment Diagrams	1,4
9	Software Requirements Specification Document	2,4
10	Design Test Cases	3,4
11	Test Report & Error Report	2,3,4
12	PPT Presentation of their project	4

Text Books:

1. Roger S Pressman,Software Engineering: A Practitioner's approach,Tata McGraw Hill Education,8th edition,2015
2. Pankaj Jalote, An integrated Approach to Software Engineering, Springer, 3rd edition,2005
3. Timothy C. Lethbridge, "Object Oriented Software Engineering" (Practical Software Development using UML and Java" Tata McGraw-Hill,2nd Edition,2019

Reference :

1. Ali Bahrami,Object Oriented Systems Development,Tata Mcgraw-Hill Education ,1st Edition,2008.

WEB TECHNOLOGIES LAB	
CSE 327	**CREDITS:2**
INSTRUCTION: 3 Hr Lab/ week	**SESSIONAL MARKS: 50**
END EXAM: 3 Periods	**END EXAM MARKS: 50**

Prerequisites:

- Basic knowledge of Computer Networks
- Exposure to Problem solving techniques and programming skills
- Basic knowledge of JAVA and Python Programming.

Course objectives:

- Introducing new web application development languages and tools to students.
- Introducing Open Source Technologies- HTML, CSS,JAVASCRIPT,PHP,MYSQL,APACHE, FLASK.

Course outcomes:

By the end of the course, Student will be able to:

1.	Develop Static Web pages and Dynamic Web pages using HTML and JavaScript respectively.
2.	Develop a dynamic website using client side scripting language- Javascript and Server side scripting language- PHP.
3.	Implement Server side programming using Servlets and JSP.
4.	Develop Engaging Web applications using Node JS and Express JS.

Mapping of Course Outcomes with Program Outcomes:

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

SYLLABUS

LIST OF EXPERIMENTS:

1. HTML & Java Script(Technologies for Client Side Programming) **CO1-3 weeks**

HTML basic tags, HTML forms. CSS, Java script Basics, JavaScript syntax, Statements,Comments, Variables, Operators, If...Else, Popup Boxes, For Loop, While Loops, Events, Functions,Objects.

2. PHP(Technology for Server Side Programming) **CO2- 4 weeks**

Creating simple webpage using PHP, Use of conditional statements in PHP, Use of looping statements in PHP,Creating different types of arrays,Usageofarrayfunctions,Creatinguser defined functions, Creation of sessions, cookies, Creation of cookies, Database connectivity in PHP with MySQL, Validating Input, Formatting the Output.

Project Development **3 weeks**

Developing a Project using PHP, MySQL, Java Script and uploading in GitHub

3. Servlets & JSP **CO3-3 weeks**

Servlet programming: Simple servlet Programming, Understanding Life cycle Methods Exercise

JSP: JSP basic tags, Working with Implicit objects, Working with Action tags

4. Web Development using Node JS & Express JS **CO4-3 weeks**

Installation, Create basic Web Applications using Node JS & Express JS, Implement user authentication on your websites.

Text Books :

1. Dietel and Nieto, *" Internet and World Wide Web – How to program* ", 4th Edition
2. PHI/Pearson EducationAsia.
3. Web Development with Node and Express: Leveraging the JavaScript Stack, by Ethan Brown, 2 nd Edition, O'Reilly Media, Inc.

Reference Books:

1. Steven Holzner, " *PHP : Complete reference*", 1st Edition, McGraw Hill Education

Web Resources:

1. https://www.coursera.org/learn/html-cssjavascript
2. https://www.coursera.org/learn/duke-programmingwebhttps://
3. www.w3schools.com/php/