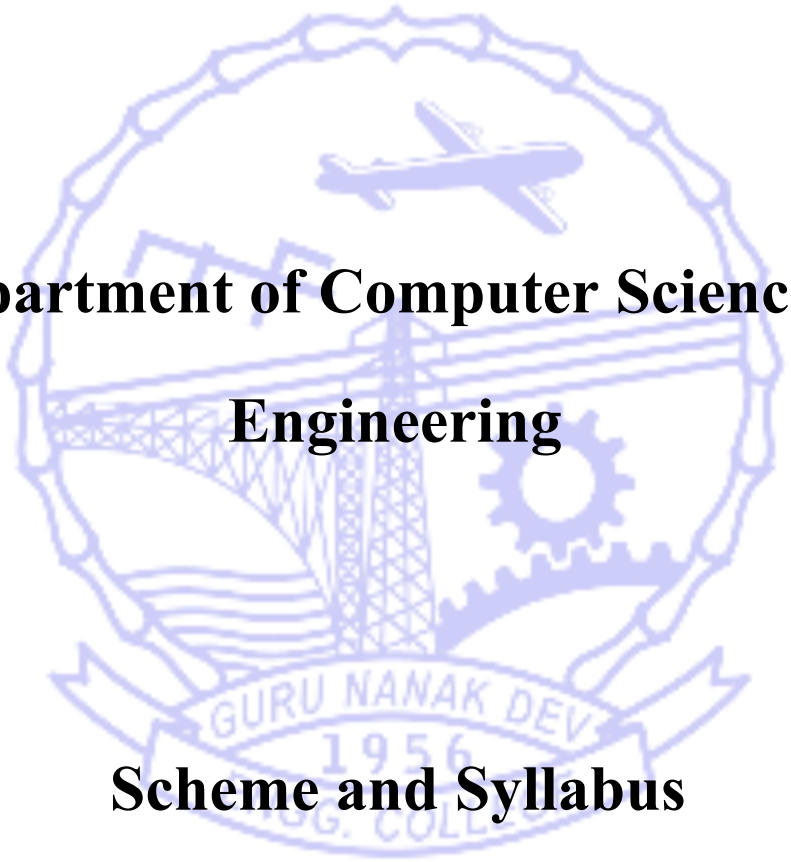
Guru Nanak Dev Engineering College, Ludhiana



Department of Computer Science & Engineering

Scheme and Syllabus

B. Tech. Computer Science & Engineering (2018 Batch Onwards)

Third Semester

S. No.	Course Category	Course Code	Course Title	Theory/ Practical	Hours per week			Internal Marks	External Marks	Total Marks	Credits
					L	T	P				
1.	Professional Core Courses	PCCS-101	Object Oriented Programming	Theory	3	0	0	40	60	100	3
2.	Professional Core Courses	PCCS-102	Computer Networks	Theory	3	0	0	40	60	100	3
3.	Engineering Science Courses	ESCS-101	Digital Electronics	Theory	3	0	0	40	60	100	3
4.	Basic Science Course	BSCS-101	Mathematics-III	Theory	3	1	0	40	60	100	4
5.	Humanities and Social Sciences including Management Courses	HSMCS-101	Human values and Professional Ethics	Theory	3	0	0	40	60	100	3
6.	Professional Core Courses	LPCCS-101	Object Oriented Programming Laboratory	Practical	0	0	4	30	20	50	2
7.	Professional Core Courses	LPCCS-102	Computer Networks Laboratory	Practical	0	0	2	30	20	50	1
8.	Engineering Science Courses	LESCS-101	Digital Electronics Laboratory	Practical	0	0	2	30	20	50	1
9.	Training	TR-101	Training-I *	Practical	0	0	0	60	40	100	1
10.	Seminar/Project	PRCS-101	Seminar and Technical Report Writing For Engineers	Practical	0	0	2	50	0	50	1
11.	Mentoring#		Mentoring and Professional Development	Practical	0	0	1	-	-	-	0
				Total	**15**	**1**	**10 + 1#**	**400**	**400**	**800**	**22**

*Evaluation of 4 weeks institutional/industrial training held after 2nd semester in the institute.

There will be one period per week for Mentoring and Professional Development; final evaluation of this course will be done based on the combined assessment of odd and even semester of respective year of study.

Fourth Semester

S. No.	Course Category	Course Code	Course Title	Theory/ Practical	Hours per week			Internal Marks	External Marks	Total Marks	Credits
					L	T	P				
1.	Professional Core Courses	PCCS-103	Discrete Mathematics	Theory	3	1	0	40	60	100	4
2.	Professional Core Courses	PCCS-104	Computer Architecture and Microprocessor	Theory	3	0	0	40	60	100	3
3.	Professional Core Courses	PCCS-105	Operating Systems	Theory	3	1	0	40	60	100	4
4.	Professional Core Courses	PCCS-106	Data Structures	Theory	3	0	0	40	60	100	3
5.	Professional Core Courses	PCCS-107	Software Engineering	Theory	3	1	0	40	60	100	4
6.	Professional Core Courses	LPCCS-103	Computer Architecture and Microprocessor Laboratory	Practical	0	0	2	30	20	50	1
7.	Professional Core Courses	LPCCS-104	Operating Systems Laboratory	Practical	0	0	2	30	20	50	1
8.	Professional Core Courses	LPCCS-105	Data Structures Laboratory	Practical	0	0	4	30	20	50	2
9.	Mandatory Courses	MCCS-101	Environmental Sciences	Theory	2	0	0	50	0	50	0
10.	Mentoring #	MPD-102	Mentoring and Professional Development	Practical	0	0	1	100	0	100	1
				Total	**17**	**3**	**8 + 1#**	**440**	**360**	**800**	**23**

There will be one period per week for Mentoring and Professional Development; final evaluation of this course will be done based on the combined assessment of odd and even semester of respective year of study.

Fifth Semester

S. No.	Course Category	Course Code	Course Title	Theory/ Practical	Hours per week			Internal Marks	External Marks	Total Marks	Credits
					L	T	P				
1.	Professional Core Courses	PCCS-108	Artificial Intelligence	Theory	3	0	0	40	60	100	3
2.	Professional Core Courses	PCCS-109	Database Management Systems	Theory	3	0	0	40	60	100	3
3.	Professional Core Courses	PCCS-110	Formal Language &Automata Theory	Theory	3	1	0	40	60	100	4
4.	Professional Core Courses	PCCS-111	Design and Analysis of Algorithms	Theory	3	1	0	40	60	100	4
5.	Professional Elective Courses	PECS-XXX	Elective-I	Theory	3	0	0	40	60	100	3
6.	Professional Core Courses	LPCCS-106	Artificial Intelligence Laboratory	Practical	0	0	2	30	20	50	1
7.	Professional Core Courses	LPCCS-107	Database Management Systems Laboratory	Practical	0	0	2	30	20	50	1
8.	Professional Core Courses	LPCCS-108	Design and Analysis of Algorithms Laboratory	Practical	0	0	2	30	20	50	1
9.	Mandatory Courses	MCCS-102	Constitution of India	Theory	2	0	0	50	0	50	0
10.	Industrial/ Institutional Training	TR-102	Training-II**	Practical	-	-	-	60	40	100	1
11.	Mentoring#		Mentoring and Professional Development	Practical	0	0	1	-	-	-	0
				Total	**17**	**2**	**6 +**	**400**	**400**	**800**	**21**

**Evaluation of 4 weeks industrial/institutional training held after 4th semester.

There will be one period per week for Mentoring and Professional Development; final evaluation of this course will be done based on the combined assessment of odd and even semester of respective year of study.

Sixth Semester

S. No.	Course Category	Course Code	Course Title	Theory/ Practical	Hours per week			Internal Marks	External Marks	Total Marks	Credits
					L	T	P				
1.	Professional Core Courses	PCCS-112	Compiler Design	Theory	3	1	0	40	60	100	4
2.	Professional Core Courses	PCCS-113	Computer Graphics	Theory	3	1	0	40	60	100	4
3.	Professional Core Courses	PCCS-114	Machine Learning	Theory	3	0	0	40	60	100	3
4.	Professional Core Courses	PCCS-115	Cyber Security	Theory	3	0	0	40	60	100	3
5.	Professional Elective Courses	PECS-XXX	Elective-II	Theory	3	0	0	40	60	100	3
6.	Open Elective Courses	OECS-XXX	Open Elective-I	Theory	3	0	0	40	60	100	3
7.	Professional Core Courses	LPCCS-109	Computer Graphics Laboratory	Practical	0	0	2	30	20	50	1
8.	Professional Core Courses	LPCCS-110	Machine Learning Laboratory	Practical	0	0	2	30	20	50	1
9.	Professional Elective Courses	PECS-XXX	Elective-II Laboratory	Practical	0	0	2	30	20	50	1
10.	Project	PRCS-102	Minor Project	Practical	0	0	2	60	40	100	1
11.	Mentoring#	MPD-103	Mentoring and Professional Development	Practical	0	0	1	100	-	100	1
Total					**18**	**2**	**8+1[#]**	**490**	**460**	**950**	**25**

[#]There will be one period per week for Mentoring and Professional Development; final evaluation of this course will be done based on the combined assessment of odd and even semester of respective year of study.

Note: Industrial Training (TR-104) of 7th/8th semester will be choice based as under:

Choice-I: Industrial Training (TR-104) will be offered in 7th semester.
Choice-II: Industrial Training (TR-104) will be offered in 8th semester.
Choice-III: Industrial Training (TR-104) will not be offered as a course.

CHOICE-I: Applicable to students opting for one semester Industrial Training (TR-104) in 7th Semester.

Seventh Semester								
S. No.	**Course Category**	**Course Code**	**Course Title**	**Theory/ Practical**	**Internal Marks**	**External Marks**	**Total Marks**	**Credits**
1.	Industrial/Institutional Training	TR-103	Training-III*	---	60	40	100	1
2.	Industrial/Institutional Training	TR-104	Industrial Training	---	350	150	500	15
Total					**410**	**190**	**600**	**16**

*Evaluation of 4 weeks industrial/institutional training held after the 6th semester.

Eighth Semester											
S. No.	**Course Category**	**Course Code**	**Course Title**	**Theory/ Practical**	**Hours per week**			**Internal Marks**	**External Marks**	**Total Marks**	**Credits**
					L	**T**	**P**				
1.	Professional Elective Courses	PECS-XXX	Elective-III	Theory	3	1	0	60	40	100	4
2.	Professional Elective Courses	PECS-XXX	Elective-IV	Theory	3	0	0	60	40	100	3
3.	Professional Elective Courses	LPECS-XXX	Elective-IV Laboratory	Practical	0	0	2	30	20	50	1
4.	Open Elective Courses	OECS-XXX	Open Elective-II	Theory	3	0	0	60	40	100	3
5.	Seminar/Project	PRCS-103	Major Project	Practical	0	0	6	120	80	200	3
6.	Seminar/Project	PRCS-107	Software Management Tools	Practical	0	0	2	50	0	50	1
7.	Mentoring#	MPD-104	Mentoring and Professional Development	Practical	0	0	1	100	-	100	1
Total					**9**	**1**	**10+1#**	**420**	**280**	**700**	**16**

#There will be one period per week for Mentoring and Professional Development; final evaluation of this course will be done based on the combined assessment of odd and even semester of respective year of study.

List of Professional Elective Courses for CHOICE-I

TRACK 1: Software Engineering

1. Elective-I: PECS-101 Software Project Management
2. Elective-II: PECS-102 Software Testing and Quality Assurance

3. Elective-III: PECS-105 Software Metrics
4. Elective-IV: PECS-106 Component Based Development
5. Elective-II Lab: LPECS-101 Software Testing and Quality Assurance Laboratory
6. Elective-IV Lab: LPECS-103 Component Based Development Laboratory

TRACK 2: Network Technologies

1. Elective-I: PECS-107 Advanced Computer Networks
2. Elective-II: PECS-108 Network Security and Cryptography
3. Elective-III: PECS-111 Blockchain Technology
4. Elective-IV: PECS-112 Internet of Things
5. Elective-II Lab: LPECS-104 Network Security and Cryptography Laboratory
6. Elective-IV Lab: LPECS-106 Internet of Things Laboratory

TRACK 3: Data Management

1. Elective-I: PECS-113 Statistics for Data Science
2. Elective-II: PECS-114 Advanced Database Management Systems
3. Elective-III: PECS-117 Big Data
4. Elective-IV: PECS-118 Data Science
5. Elective-II Lab: LPECS-107 Advanced Database Management Systems Laboratory
6. Elective-IV Lab: LPECS-109 Data Science Laboratory

TRACK 4: Machine Intelligence

1. Elective-I: PECS-119 Information Retrieval
2. Elective-II: PECS-120 Natural Language Processing
3. Elective-III: PECS-123 Human Computer Interaction
4. Elective-IV: PECS-124 Deep Learning
5. Elective-II Lab: LPECS-110 Natural Language Processing Laboratory
6. Elective-IV Lab: LPECS-112 Deep Learning Laboratory

TRACK 5: Algorithm Design and Programming

1. Elective-I: PECS-125 System Programming

2. Elective-II: PECS-126 Java Programming
3. Elective-III: PECS-129 Parallel and Distributed Algorithms
4. Elective-IV: PECS-130 Mobile Application Development
5. Elective-II Lab: LPECS-113 Java Programming Laboratory
6. Elective-IV Lab: LPECS-115 Mobile Application Development Laboratory

CHOICE-II: Applicable to students opting for one semester Industrial Training (TR-104) in 8th Semester.

Seventh Semester											
S. No.	**Course Category**	**Course Code**	**Course Title**	**Theory/ Practical**	**Hours per week**			**Internal Marks**	**External Marks**	**Total Marks**	**Credits**
					L	**T**	**P**				
1.	Professional Elective Courses	PECS-XXX	Elective-III	Theory	3	1	0	60	40	100	4
2.	Professional Elective Courses	PECS-XXX	Elective-IV	Theory	3	0	0	60	40	100	3
3.	Professional Elective Courses	LPECS-XXX	Elective-IV Laboratory	Practical	0	0	2	30	20	50	1
4.	Open Elective Courses	OECS-XXX	Open Elective-II	Theory	3	0	0	60	40	100	3
5.	Seminar/Project	PRCS-103	Major Project	Practical	0	0	6	120	80	200	3
6.	Seminar/Project	PRCS-106	Technical Aptitude	Practical	0	0	2	50	0	50	1
7.	Training	TR-103	Training-III*	Practical	-	-	-	60	40	100	1
8.	Mentoring #	MPD-104	Mentoring and Professional Development	Practical	0	0	1	-	-	-	0
Total					**9**	**1**	**10+1#**	**380**	**320**	**700**	**16**

*Evaluation of 4 weeks industrial/institutional training held after 6th semester.

#There will be one period per week for Mentoring and Professional Development; final evaluation of this course will be done based on the combined assessment of odd and even semester of respective year of study.

Eighth Semester								
S. No.	**Course Category**	**Course Code**	**Course Title**	**Theory/ Practical**	**Internal Marks**	External Marks	Total Marks	**Credits**
1.	Industrial/ Institutional Training	TR-104	Industrial Training	---	350	150	500	15
2.	Mentoring #	MPD-104	Mentoring and Professional Development	Practical	100	0	100	1
			Total		**450**	**150**	**600**	**16**

List of Professional Elective Courses for CHOICE-II

TRACK 1: Software Engineering

1. Elective-I: PECS-101 Software Project Management
2. Elective-II: PECS-102 Software Testing and Quality Assurance
3. Elective-III: PECS-103 Agile Software Development
4. Elective-IV: PECS-104 Object Oriented Design using UML
5. Elective-II Lab: LPECS-101 Software Testing and Quality Assurance Laboratory
6. Elective-IV Lab: LPECS-102 Object Oriented Design using UML Laboratory

TRACK 2: Network Technologies

1. Elective-I: PECS-107 Advanced Computer Networks
2. Elective-II: PECS-108 Network Security and Cryptography
3. Elective-III: PECS-109 Software Defined Networks
4. Elective-IV: PECS-110 Wireless Sensor Networks
5. Elective-II Lab: LPECS-104 Network Security and Cryptography Laboratory
6. Elective-IV Lab: LPECS-105 Wireless Sensor Networks Laboratory

TRACK 3: Data Management

1. Elective-I: PECS-113 Statistics for Data Science
2. Elective-II: PECS-114 Advanced Database Management Systems
3. Elective-III: PECS-115 Data Warehouse and Data Mining
4. Elective-IV: PECS-116 Cloud Computing
5. Elective-II Lab: LPECS-107 Advanced Database Management Systems Laboratory
6. Elective-IV Lab: LPECS-108 Cloud Computing Laboratory

TRACK 4: Machine Intelligence

1. Elective-I: PECS-119 Information Retrieval
2. Elective-II: PECS-120 Natural Language Processing
3. Elective-III: PECS-121 Computer Vision
4. Elective-IV: PECS-122 Soft Computing
5. Elective-II Lab: LPECS-110 Natural Language Processing Laboratory
6. Elective-IV Lab: LPECS-111 Soft Computing Laboratory

TRACK 5: Algorithm Design and Programming

1. Elective-I: PECS-125 System Programming
2. Elective-II: PECS-126 Java Programming
3. Elective-III: PECS-127 Advanced Algorithm Design and Analysis
4. Elective-IV: PECS-128 Web Technologies
5. Elective-II Lab: LPECS-113 Java Programming Laboratory
6. Elective-IV Lab: LPECS-114 Web Technologies Laboratory

CHOICE-III: Applicable to students not opting for one semester Industrial Training (TR-104).

Seventh Semester											
S. No.	Course Category	Course Code	Course Title	Theory/ Practical	Hours per week			Internal Marks	External Marks	Total Marks	Credits
					L	T	P				
1	Professional Elective Courses	PECS-XXX	Elective-III	Theory	3	1	0	60	40	100	4
2	Professional Elective Courses	PECS-XXX	Elective-IV	Theory	3	0	0	60	40	100	3
3	Professional Elective Courses	LPECS-XXX	Elective-IV Laboratory	Practical	0	0	2	30	20	50	1
4	Open Elective Courses	OECS-XXX	Open Elective-II	Theory	3	0	0	60	40	100	3
5	Seminar/Project	PRCS-103	Major Project	Practical	0	0	6	120	80	200	3
6	Seminar/Project	PRCS-106	Technical Aptitude	Practical	0	0	2	50	0	50	1
7	Training	TR-103	Training-III*	Practical	-	-	-	60	40	100	1
8	Mentoring #	MPD-104	Mentoring and Professional Development	Practical	0	0	1	-	-	-	0
	Total				**9**	**1**	**10+1#**	**380**	**320**	**700**	**16**

*Evaluation of 4 weeks industrial/institutional training held after 6th semester.

#There will be one period per week for Mentoring and Professional Development; final evaluation of this course will be done based on the combined

assessment of odd and even semester of respective year of study.

Eighth Semester											
S. No.	**Course Category**	**Course Code**	**Course Title**	**Theory/ Practical**	**Hours per week**			**Internal Marks**	**External Marks**	**Total Marks**	**Credits**
					L	**T**	**P**				
1.	Professional Elective Courses	PECS-XXX	Elective-III	Theory	3	1	0	60	40	100	4
2.	Professional Elective Courses	PECS-XXX	Elective-IV	Theory	3	0	0	60	40	100	3
3.	Professional Elective Courses	LPECS-XXX	Elective-IV Laboratory	Practical	0	0	2	30	20	50	1
4.	Open Elective Courses	OECS-XXX	Open Elective-II	Theory	3	0	0	60	40	100	3
5.	Seminar/Project	PRCS-103	Major Project	Practical	0	0	6	120	80	200	3
6.	Seminar/Project	PRCS-107	Software Management Tools	Practical	0	0	2	50	0	50	1
7.	Mentoring #	MPD-104	Mentoring and Professional Development	Practical	0	0	1	100	-	100	1
Total					**9**	**1**	**10+1[#]**	**420**	**280**	**700**	**16**

[#]There will be one period per week for Mentoring and Professional Development; final evaluation of this course will be done based on the combined assessment of odd and even semester of respective year of study.

Professional Elective Courses for CHOICE-III

TRACK 1: Software Engineering

1. Elective-I: PECS-101 Software Project Management
2. Elective-II: PECS-102 Software Testing and Quality Assurance
3. Elective-III: PECS-103 Agile Software Development
4. Elective-IV: PECS-104 Object Oriented Design using UML
5. Elective-V: PECS-105 Software Metrics
6. Elective-VI: PECS-106 Component Based Development
7. Elective-II Lab: LPECS-101 Software Testing and Quality Assurance Laboratory
8. Elective-IV Lab: LPECS-102 Object Oriented Design using UML Laboratory
9. Elective-VI Lab: LPECS-103 Component Based Development Laboratory

TRACK 2: Network Technologies

1. Elective-I: PECS-107 Advanced Computer Networks
2. Elective-II: PECS-108 Network Security and Cryptography
3. Elective-III: PECS-109 Software Defined Networks
4. Elective-IV: PECS-110 Wireless Sensor Networks
5. Elective-V: PECS-111 Blockchain Technology
6. Elective-VI: PECS-112 Internet of Things
7. Elective-II Lab: LPECS-104 Network Security and Cryptography Laboratory
8. Elective-IV Lab: LPECS-105 Wireless Sensor Networks Laboratory
9. Elective-VI Lab: LPECS-106 Internet of Things Laboratory

TRACK 3: Data Management

1. Elective-I: PECS-113 Statistics for Data Science
2. Elective-II: PECS-114 Advanced Database Management Systems
3. Elective-III: PECS-115 Data Warehouse and Data Mining
4. Elective-IV: PECS-116 Cloud Computing
5. Elective-V: PECS-117 Big Data

6. Elective-VI: PECS-118 Data Science
7. Elective-II Lab: LPECS-107 Advanced Database Management Systems Laboratory
8. Elective-IV Lab: LPECS-108 Cloud Computing Laboratory
9. Elective-VI Lab: LPECS-109 Data Science Laboratory

TRACK 4: Machine Intelligence

1. Elective-I: PECS-119 Information Retrieval
2. Elective-II: PECS-120 Natural Language Processing
3. Elective-III: PECS-121 Computer Vision
4. Elective-IV: PECS-122 Soft Computing
5. Elective-V: PECS-123 Human Computer Interaction
6. Elective-VI : PECS-124 Deep Learning
7. Elective-II Lab: LPECS-110 Natural Language Processing Laboratory
8. Elective-IV Lab: LPECS-111 Soft Computing Laboratory
9. Elective-VI Lab: LPECS-112 Deep Learning Laboratory

TRACK 5: Algorithm Design and Programming

1. Elective-I: PECS-125 System Programming
2. Elective-II: PECS-126 Java Programming
3. Elective-III: PECS-127 Advanced Algorithm Design and Analysis
4. Elective-IV: PECS-128 Web Technologies
5. Elective-V: PECS-129 Parallel and Distributed Algorithms
6. Elective-VI: PECS-130 Mobile Application Development
7. Elective-II Lab: LPECS-113 Java Programming Laboratory
8. Elective-IV Lab: LPECS-114 Web Technologies Laboratory
9. Elective-VI Lab: LPECS-115 Mobile Application Development

List of Open Electives courses to be offered to other departments:

Open Elective – I:

1. OECS-101: Software Project Management
2. OECS-102: Object Oriented Programming using Java
3. OECS-103: Cyber Laws and Ethics
4. OECS-104: Data Structures

Open Elective – II:

1. OECS-105: Python Programming
2. OECS-106: Business Information System
3. OECS-107: Artificial Intelligence
4. OECS-108: Computer Networks

Open Elective – III:

1. OECS-109: Cyber Security
2. OECS-110: Real Time Systems
3. OECS-111: Multimedia Systems
4. OECS-112: Database Management Systems

Subject Code: PCCS-101

Subject Name: Object Oriented Programming

Programme: B.Tech. (CSE)	**L:** 3 **T:** 0 **P:** 0
Semester: 3rd	**Teaching Hours:** 40
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 20%
External Marks: 60	**Duration of End Semester Exam (ESE):** 3 hrs
Total Marks: 100	**Elective Status:** Compulsory

Additional Material Allowed in ESE: [NIL]

On completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1	Develop an understanding of object oriented programming principles and object oriented design.
2	Use of operators, control structures, and data types with their methods.
3	Make use of arrays and string handling methods.
4	Design user defined functions, modules, and packages.
5	Investigate and implement polymorphism, inheritance, dynamic memory management and exception handling techniques to solve problems.
6	Create and handle files in object oriented programming.

Detailed Contents:

Part-A

Object-Oriented Programming Concepts: Introduction, Comparison between procedural programming paradigm and object-oriented programming paradigm, Features of object-oriented programming: Encapsulation, Class, Object, Abstraction, Data hiding, polymorphism, and Inheritance. Introduction of object oriented design. **[3 Hours]**

Data Types, Operators, and Control Structures: Basic data types, Derived data types, Keywords, Identifiers, Constants and variables, Type casting, Operators, and Operator precedence. Control Structures: if statement, switch-case, for, while and do-while loops, break

and continue statement. **[6 Hours]**

Classes and Objects: Implementation of a class, Creating class objects, Operations on objects, Relationship among objects, Accessing class members, Access specifiers, Constructor and destructor, Types of constructor, Static members, Empty classes, Nested classes, Local classes, Abstract classes, Container classes. **[5 Hours]**

Functions, Arrays, and String Handling: Function components, Default arguments, Passing parameters, Function prototyping, Call by value, Call by reference, Return by reference, Inline functions, Friend functions, Static functions, Recursion, Array declaration, Types of arrays, Array of objects, String handling. **[6 Hours]**

Part-B

Polymorphism and Type Conversion: Introduction, Concept of binding – Early binding and late binding, Virtual functions, Pure virtual functions, Operator Overloading, Rules for overloading operators, Overloading of various operators, Function overloading, Constructor overloading, Type conversion – Basic type to class type, Class type to basic type, Class type to another class type. **[6 Hours]**

Inheritance: Introduction, defining derived classes, Types of inheritance, Ambiguity in multiple and multipath inheritance, Virtual base class, Objects slicing, Overriding member functions, Object composition and delegation. **[5 Hours]**

Dynamic Memory Management using Pointers: Declaring and initializing pointers, Accessing data through pointers, Pointer arithmetic, Memory allocation –Static and Dynamic, Dynamic memory management using new and delete operators, Pointer to an object, this pointer, Pointer related problems – Dangling/wild pointers, Null pointer assignment, Memory leak and Allocation failures. **[5 Hours]**

Exceptions Handling: Review of traditional error handling, Basics of exception handling, Exception handling mechanism, Throwing mechanism, Catching mechanism, Rethrowing an exception, Specifying exceptions. **[2 Hours]**

Files Handling: File streams, Hierarchy of file stream classes, Error handling during file operations, Reading/writing of files, Accessing records randomly, Updating files. **[2 Hours]**

Text Books

1. R. Lafore, "Object Oriented Programming in C++", Waite Group.
2. E. Balagurusamy, "Object Oriented Programming with C++", Tata McGraw Hill.
3. P YashavantKanetkar, "Let Us C++", BPB Publications.
4. Bjarne Stroustrup, "The C++ Programming Language", Addison Wesley.

Reference Books

1. Herbert Schildt, "The Complete Reference to C++ Language", McGraw Hill-Osborne.
2. B.F.Lippman, "C++ Primer", Addison Wesley.
3. Farrell, "Object Oriented using C++", Cengage Learning.
4. Barbara Liskov, Program Development in Java, Addison-Wesley.

Books and online learning material

1. E. Balagurusamy, "Object Oriented Programming with C++", Tata McGraw Hill.
   http://www.mldcollege.com/panel/programs/Object%20Oriented%20Programming%20with%20C -Bal%20-%20E.Balagurusamy.pdf

Online Courses and Video Lectures

1. https://nptel.ac.in/courses/106101208/1 Accessed on August, 20, 2019
2. https://nptel.ac.in/courses/106101208/18 Accessed on August, 20, 2019
3. https://nptel.ac.in/courses/106101208/20 Accessed on August, 20, 2019
4. https://nptel.ac.in/courses/106101208/21 Accessed on August, 20, 2019
5. https://nptel.ac.in/courses/106101208/23 Accessed on August, 20, 2019
6. https://nptel.ac.in/courses/106101208/25 Accessed on August, 20, 2019

Subject Code: PCCS-102

Subject Name: Computer Networks

Programme: B.Tech. CSE	**L:** 3 **T:** 0 **P:** 0
Semester: 3	**Teaching Hours:** 38
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 40%
External Marks: 60	**Duration of End Semester Exam (ESE):** 3 hrs
Total Marks: 100	**Elective Status:** Compulsory

Prerequisites: Knowledge of Computer System fundamentals.

Additional Material Allowed in ESE: [NIL]

On completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1	Develop an understanding of modern network architectures from a design and performance.
2	Understand the major concepts involved in wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs).
3	Analyze various protocols to develop network related applications for future needs.
4	Apply the knowledge of different network designs and various logical models of networking to solve problems of communication over different medium.
5	Utilize knowledge of routing and congestion control algorithms to overcome various issues over different complex networking structures.
6	Discuss algorithms for medium access sub layer to avoid collision and error problems over different types of networks.

Detailed Contents:

Part-A

Data Communication Components: Representation of data and data flow, Various Network Topologies, Protocols and Standards, LAN: Wired LAN, Wireless LANs, Connecting LAN and Virtual LAN, Techniques for Bandwidth utilization: Multiplexing- Frequency division, Time

division and Wave division, Concepts on spread spectrum, OSI model, TCP/IP reference model and their comparison. **[6 Hours]**

Physical Layer: Concept of analog and digital systems, Transmission Media, Transmission impairments and Data rate limits- Nyquist formula, Shannon formula, Switching- Circuit, Message and Packet switching. **[7 Hours]**

Data Link Layer and Medium Access Sub Layer: Error Detection and Error Correction- Fundamentals, Block coding, Hamming Distance, CRC; Flow Control and Error control protocols- Stop and Wait, Go back–N ARQ, Selective Repeat ARQ, Sliding Window, Piggybacking, Random Access, Multiple access protocols- Pure ALOHA, Slotted ALOHA, CSMA/CD, CDMA/CA. **[7 Hours]**

Part-B

Network Layer: Logical addressing- IPV4, IPV6; Address mapping- ARP, RARP, BOOTP and DHCP–Delivery, Routing algorithms, Congestion control policies, Leaky bucket and token bucket algorithms. **[6 Hours]**

Transport Layer: Design issues, Elements of transport Protocols- Connection establishment and release, Process to Process Communication, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), flow control. **[6 Hours]**

Session, Presentation and Application Layer: Session Layer- Design issue, remote procedure call. Presentation Layer- Design issue, Data compression techniques. Application Layer- Domain Name Space (DNS), DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP. **[6 Hours]**

Text Books

1. Andrew S. Tanenbaum, “Computer Networks”, Pearson Education.
2. Behrouz A. Forouzan, “Data Communication &Networking”, Tata McGraw Hill.

Reference Books

1. Douglas E. Comer, “Internetworking with TCP/IP”, Volume-I, Prentice Hall, India.

2. W. Stallings, "Data and Computer Communication", Prentice Hall of India.
3. James F. Kurose and Keith W. Ross, "Computer Networking", Pearson Education.

Books and online learning material

1. An Introduction to Computer Networks by Peter L Dordal, Department of Computer Science, Loyola University Chicago.

   http://intronetworks.cs.luc.edu/current/ComputerNetworks.pdf

Online Courses and Video Lectures

1. https://nptel.ac.in/courses/106105081/ Accessed on May. 15, 2019
2. https://nptel.ac.in/courses/106105081/2 Accessed on May. 15, 2019
3. https://nptel.ac.in/courses/106105081/5 Accessed on May. 15, 2019
4. https://nptel.ac.in/courses/106105081/16 Accessed on May. 15, 2019
5. https://nptel.ac.in/courses/106105081/31 Accessed on May. 15, 2019

Subject Code: ESCS-101

Subject Name: Digital Electronics

Programme: B.Tech. (CSE)	**L:** 3 **T:** 0 **P:** 0
Semester: 3	**Teaching Hours: 3**6
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 50%
External Marks: 60	**Duration of End Semester Exam (ESE):** 3 hrs
Total Marks: 100	**Elective Status:** Compulsory

Prerequisites: NIL

Additional Material Allowed in ESE: [Scientific Calculator or NIL]

On completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1	Understand the relationships between Boolean algebra, combinational logic, and sequential logic.
2	Solve combinational logic problem formulation and logic optimization.
3	Construct digital logic circuits using gates and state-of-the art MUX, ROM, PLA and PAL units
4	Create profound analysis and design of synchronous and asynchronous sequential Circuits
5	Design and inspect digital circuits to meet desired needs within realistic constraints.
6	Develop skills to build and troubleshoot digital circuits.

Detailed Contents:

Part-A

Number Systems: Binary, Octal, Decimal, Hexadecimal. Number base conversions, 1's, 2's, rth's complements, signed Binary numbers. Binary Arithmetic, Binary codes: Weighted BCD, Gray code, Excess 3 code, ASCII – conversion from one code to another. **[5 Hours]**

Boolean Algebra: Boolean postulates and laws – De-Morgan's Theorem, Principle of Duality, Boolean expression – Boolean function, Minimization of Boolean expressions – Sum of Products (SOP), Product of Sums (POS), Minterm, Maxterm, Canonical forms, Conversion between canonical forms, Karnaugh map Minimization. **[5 Hours]**

Logic GATES: AND, OR, NOT, NAND, NOR, Exclusive-OR and Exclusive-NOR. Implementations of Logic Functions using gates, NAND-NOR implementations. Study of logic families like RTL, DTL, DCTL, TTL, MOS, CMOS, ECL and their characteristics. **[5 Hours]**

Combinational Circuits: Design procedure – Adders, Subtractors, Serial adder/Subtractor, Parallel adder/ Subtractor Carry look ahead adder, BCD adder, Magnitude Comparator, Multiplexer/Demultiplexer, encoder/decoder, parity checker, code converters. Implementation of combinational logic using MUX. **[6 Hours]**

Part-B

Sequential Circuits: Flip flops SR, JK, T, D and Master slave, Excitation table, Edge triggering, Level Triggering, Realization of one flip flop using other flip flops. Asynchronous/Ripple counters, Synchronous counters, Modulo-n counter, Ring Counters. Classification of sequential circuits-Moore and Mealy, Design of Synchronous machines: state diagram, Circuit implementation. Shift registers. **[6 Hours]**

Signal Conversions: Analog & Digital signals. A/D and D/A conversion techniques (Weighted type, R-2R Ladder type, Counter Type, Dual Slope type, Successive Approximation type). **[4 Hours]**

Introduction to Design with PLDs: Introduction to programmable logic devices- Programmable Logic Array (PLA), Programmable Array Logic (PAL), Field Programmable Gate Arrays (FPGA) **[5 Hours]**

Text Books

1. M. Morris Mano, "Digital Design", Prentice Hall of India Pvt. Ltd./Pearson Education (Singapore) Pvt. Ltd., New Delhi.
2. John F.Wakerly, "Digital Design", Pearson/PHI.
3. John M. Yarbrough, "Digital Logic Applications and Design", Thomson Learning.
4. Charles H.Roth., "Fundamentals of Logic Design", Thomson Learning.
5. A. Anand Kumar, "Fundamentals of Digital Circuits", PHI.

Reference Books

1. Donald P. Leach and Albert Paul Malvino, "Digital Principles and Applications", TMH.
2. William H. Gothmann, "Digital Electronics", PHI.
3. A. K. Maini, "Digital Electronics: Principles, Devices and Applications, Wiley, 2007.

E-Books and online learning material

1. Web packages for HDL, GHDL, FreeHDL
2. PSpices and NGSpice
3. Xcircuit and Scilab
4. NPTEL website and IITs virtual laboratory

Online Courses and Video Lectures

1. https://nptel.ac.in/courses/117106086/32 Accessed on May 19, 2019
2. https://nptel.ac.in/courses/108105113/ Accessed on May 19, 2019
3. https://www.coursera.org/learn/digital-systems Accessed on May 19, 2019

Subject Code: BSCS-101

Subject Name: Mathematics III

Programme: B.Tech.(CSE)	**L:** 3 **T:** 1 **P:** 0
Semester: 3	**Teaching Hours:** 40
Theory/Practical: Theory	**Credits:** 4
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 90%
External Marks: 60	**Duration of End Semester Exam (ESE):** 3 hrs
Total Marks: 100	**Elective Status:** Compulsory

Prerequisites: Knowledge of partial differentiation, probability and statistics.

Additional material allowed in ESE: Scientific calculator, log tables, probability distribution tables, statistical tables or NIL

On completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1	Understand Partial Differential Equations and their solutions techniques.
2	Understand Analytic functions and evaluation of derivative of functions of complex variable.
3	Evaluate integration of functions of complex variables.
4	Analyze probability spaces, random variables and different probability distribution.
5	Fit the given data into best fit curve.
6	Apply statistical methods for analyzing experimental data.

Detailed Contents:

Part-A

Differential calculus of complex variables: Separation of elementary functions of complex variables, Cauchy-Riemann equations, analytic functions, elementary analytic functions (exponential, trigonometric, logarithm) and their properties, harmonic functions, finding harmonic conjugate. **[8 Hours]**

Integral Calculus of functions of complex variables: Complex integrals, Cauchy-Goursat theorem (without proof), Cauchy Integral formula (without proof) Taylor's series, zeros of analytic functions, singularities, Laurent's series; Residues, Cauchy Residue theorem (without proof), Evaluation of definite integral involving sine and cosine, Mobius transformations and their properties. **[8 Hours]**

Linear Systems: Gauss's elimination method and Gauss's Jordan method **[3 Hours]**

Part-B

Applied Statistics: Curve fitting by the method of least squares- fitting of straight lines and second degree parabolas **[5 Hours]**

Probability Distributions: Probbility spaces, Discrete random variables, Poisson and binomial distribution. Continuous random variables and their properties, distribution functions and densities, normal, exponential and gamma densities. **[8 Hours]**

Statistics: Test of significance: Large sample test for single proportion, difference of proportions, single mean, difference of means, and difference of standard deviations. Small sample test for single mean and difference of means, test for ratio of variances - Chi-square test for goodness of fit and independence of attributes. **[8 Hours]**

Textbooks:

1. Erwin Kreyszig, Advanced Engineering Mathematics, John Wiley & Sons.
2. N.P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications.
3. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers.

Reference books:

1. G.B. Thomas and R.L. Finney, Calculus and Analytic geometry, Pearson.
2. P. G. Hoel, S. C. Port and C. J. Stone, Introduction to Probability Theory, Universal Book Stall.
3. S. Ross, A First Course in Probability, Pearson Education India.
4. J. W. Brown and R. V. Churchill, Complex Variables and Applications, Mc-Graw Hill

Subject Code: HSMCS-101

Subject Name: Human values and Professional Ethics

Programme: B.Tech. (CSE)	**L:** 3 **T:** 0 **P:** 0
Semester: 3	**Teaching Hours:** 38
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** Nil
External Marks: 60	**Duration of End Semester Exam (ESE):** 3 hrs
Total Marks: 100	**Elective Status:** Compulsory

Additional Material Allowed in ESE: NIL

On completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1	Discriminate between valuable and superficial in the life.
2	Encourages students to discover what they consider valuable.
3	Understand the value required to be a good human being and apply these values in real life.
4	Evaluate and modify the behavior.
5	Understand fundamental and organizational duties and protect individual and social rights.
6	Know about professional behavior, values and guiding principles.

Detailed Contents:

Part-A

Ethics and values: Importance of Ethics and values, Difference between moral, ethics and values , Nature of Values, The Structure of Value Relations , Integrity, Work ethics, Service learning, Virtues, Respect for others, Living peacefully, caring , Sharing , Honesty, Courage, Valuing time , Cooperation, Commitment, Empathy, Self confidence, Challenges in the work place, spirituality.

[8 Hours]

Value education: Need for value education, Basic guidelines, Self Exploration, Values in family and Harmony in existence, Values across cultures. **[6 Hours]**

Personality and behavior development: God and scientific attitude, positive thinking, Integrity and discipline, punctuality, Aware of self destructive habits, Association and cooperation, Doing best, motivation Theories and Case study, Johari Window, Leadership Styles and Theories, Win-Win policy, SWOT Analysis **[6 Hours]**

Part-B

Human rights: Definition, Fundamental rights and duties, Regional, national and Universal protection of human rights, human rights and vulnerable groups. **[5 Hours]**

Professional ethics: Introduction, Objectives and types of professional ethics, Personal vs. Professional Ethics. Ethics in Profession, Ethics for employees, Rights of an Employee, Whistle-blower Policy, Code of Conduct, code of ethics, global issues (Technology revolution, international trade, globalization, environmental ethics, war ethics and intellectual property rights). **[8 Hours]**

Engineering Ethics: Scope and approach, Steps to Deal with Issues, Types of Inquiries, Moral Dilemma, Steps to Solve Dilemma, Engineering as Social Experimentation, Engineers as responsible experimenters. **[5 Hours]**

Text Books

1. John Berry, Janek, Pandey; Poortinga, Ype 'Handbook of Cross-cultural Psychology', Boston.
2. R.S Naagarazan , "Professional Ethics and Human Values", New Age Publisher.
3. P.L. Dhar, R.R. Gaur, 'Science and Humanism', Common Wealth Publishers.
4. Tanu Shukla, Anupam Yadav , Gajendra Singh Chauhan, "Human Values and Professional Ethics", First edition, 2017.
5. M. Govindrajran, S. Natrajan & V.S. Senthil Kumar, "Engineering Ethics (including Human Values)", Prentice Hall of India Ltd.
6. Premvir Kapoor, "Professional Ethics and Human Values", Khanna Publishing.

Reference Books

1. E.G. Seebauer& Robert L. Berry, 'Fundamentals of Ethics for Scientists & Engineers', Oxford University Press.
2. R.R. Gaur , R Sangal, G P Bagria, "A Foundation Course in Human Values and Professionals Ethics", Excel Publishers.
3. Vaishali R Khosla, Kavita Bhagat, "Human Values And Professional Ethics", Technical Publications.

E-Books and online learning material

1. Human values and Professional Ethics

https://crescent.education/wp-content/uploads/2018/12/Crescent-human-values-professional-ethics.pdf

2. Professional Ethics and Human Values Noteshttps://www.academia.edu/8844628/Professional_Ethics_and_Human_Values_Notes
3. Engineering Ethics Tutorial https://www.tutorialspoint.com/engineering_ethics/index.htm
4. Professional Ethics and Values in Engineering

https://www.srecwarangal.ac.in/cse-downloads/peve-unit-1.pdf

Online Courses and Video Lectures

1. https://nptel.ac.in/courses/109104068/30 Accessed on June 27, 2019
2. https://www.youtube.com/watch?v=3-UEi_djb7w Accessed on June 27, 2019
3. https://www.youtube.com/watch?v=VVsrsOaEcFQ Accessed on June 27, 2019
4. https://www.youtube.com/watch?v=-iJIrYreix0 Accessed on June 27, 2019

Subject Code: LPCCS-101

Subject Name: Object Oriented Programming Laboratory

Programme: B.Tech. (CSE)	**L:** 0 **T:** 0 **P:** 4
Semester: 3	**Teaching Hours:** 42
Theory/Practical: Practical	**Credits:** 2
Internal Marks: 30	**Percentage of Numerical/Design/Programming Problems:**100%
External Marks: 20	**Duration of End Semester Exam (ESE):** 2 hrs
Total Marks: 50	**Elective Status:** Compulsory

Prerequisites: NIL

On Completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1	Compare and contrast object oriented programming paradigm with procedure oriented programming paradigm.
2	Design and implement efficient programs to solve computing problems in a high level programming language.
3	Utilize knowledge of different object oriented principles to identify and apply the appropriate techniques in problem solving.
4	Apply the knowledge acquired to troubleshoot programming related problems.
5	Utilize the knowledge and principles of object oriented programming while working in multidisciplinary teams.
6	Design and develop projects using object oriented tools and techniques.

[Control statements]

1. Demonstrate the use of conditional control statements like if, if-else, if-else ladder, nested if-else, and switch-case statement.
2. Illustrate the use of loop control statements like for, while, and do-while.
3. Write a program to demonstrate the use of break and continue statement.

[Arrays and Strings]

4. Demonstrate the use of one dimensional and two dimensional arrays by using suitable programs.
5. Illustrate the use of various string handling functions.

[Classes and Objects]

6. Program to illustrate the concept of classes and object.
7. Program to illustrate the concept of nesting of member functions.
8. Program to show the working of static members (static functions and static variables) in a class.
9. Program to demonstrate the use of friend functions.

[Constructors and Destructors]

10. Program to illustrate the concept of default constructor, parameterized constructor, and copy constructor.
11. Program to illustrate the concept of destructors.

[Polymorphism]

12. Program to demonstrate the concept of operator overloading
13. Program to illustrate the concept of function overloading and constructor overloading.
14. Program to illustrate the concept of virtual functions and pure virtual functions.

[Inheritance]

15. Program to illustrate the concept of inheritance.
16. Program to illustrate the concept of ambiguity in multiple inheritance.
17. Program to illustrate the order of execution of constructors and destructors in inheritance.
18. Program to demonstrate the concept of function overriding.

[Exception handling]

19. Program to illustrate the exception handling mechanism.

[File handling]

20. Program to illustrate the concept of file pointers.

21. Program to perform read and write operations on a file.

Any one project

Banking System Project

Description: The BANKING SYSTEM project has account class with data members like account number, name, deposit, withdraw amount and type of account. Customer data is stored in a binary file. A customer can deposit and withdraw amount in his account. User can create, modify and delete account.

Library Management System Project

The LIBRARY MANAGEMENT SYSTEM project has book and student class with data members like book no, bookname, authorname. Books record is stored in a binary file. A student can issue book and deposit it within 15 days. Student is allowed to issue only one book. Student Records are stored in binary file. Administrator can add, modify or delete record.

Reference Material

Manuals available in Lab

Subject Code: LPCCS-102

Subject Name: Computer Networks Laboratory

Programme: B.Tech. CSE	**L:** 0 **T:** 0 **P:** 2
Semester: 3	**Teaching Hours:** 24
Theory/Practical: Practical	**Credits:** 01
Internal Marks: 30	**Percentage of Numerical/Design/Programming Problems:**100%
External Marks: 20	**Duration of End Semester Exam (ESE):** 2 hrs
Total Marks: 50	**Elective Status:** Compulsory

Prerequisites: Fundamentals of Computer System.

On Completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1	Analyze and configure protocols concerning various network technologies over different mediums and layers.
2	Apply the knowledge of different network components, transmission mediums and tools to solve various problems of communication.
3	Design and develop different network design and logical models of networking to solve network related problems.
4	Utilize knowledge of modern network simulation tools to propose solution for efficient working of networks for real world problems.
5	Make use of various troubleshooting methods to overcome networking problems.
6	Function in multidisciplinary teams through groups while working in different network environments with the help of resource sharing.

Special Instruction related to resources requirement: Except practical number 10.

Sr.No.	Name of Practical
1.	Familiarization with networking components, transmission media, tools and devices: LAN Adapters, Hubs, Switches, Routers etc.
2.	Study of various LAN topologies and their creation using network devices, cables and

	computers, Preparing straight and cross cables.
3.	Configuration of TCP/IP Protocols in Windows and Linux.
4.	Implementation of resource (file, printer, etc.) sharing.
5.	Designing and implementing class A, B and C networks.
6.	Subnet planning and its implementation.
7.	To configure dynamic IP address for a computer connected to a LAN.
8.	Use of commands like ping, ipconfig for trouble shooting network related problems.
9.	Develop a program to compute the Hamming Distance between any two code words.
10.	To configure proxy server, Familiarization with network simulation tools.

Reference Material

Manuals available in

Subject Code: LESCS-101

Subject Name: Digital Electronics Laboratory

Programme: B.Tech. (CSE)	**L:** 0 **T:** 0 **P:**2
Semester: 3	**Teaching Hours: 24**
Theory/Practical: Practical	**Credits:** 1
Internal Marks: 30	**Percentage of Numerical/Design/Programming Problems:**100%
External Marks: 20	**Duration of End Semester Exam (ESE):** 2 hrs
Total Marks: 50	**Elective Status:** Compulsory

Prerequisites: NIL

On Completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1	Identify and apply the knowledge of logic gates and integrated circuits to solve related problems.
2	Design and implement combinational & sequential circuits for engineering problems.
3	Choose and compare the usage of appropriate techniques and tools to solve digital circuits problem.
4	Apply the knowledge acquired to demonstrate the usage of digital circuits in computers at large.
5	Utilize the knowledge and principles of digital electronics while working in multidisciplinary team formation.
6	Design simple digital systems based on these digital abstractions, using the "digital paradigm".

Resources required: Digital Trainer Board, ICs, connecting wires.

S. No.	Name of Practical
1.	Truth-table verification of OR, AND, NOT, XOR, NAND and NOR gates using various IC's.
2.	Realization of OR, AND, NOT and XOR functions using universal gates IC's 7400 and 7402.

3.	Half adder / Full adder: Realization using basic and XOR gates IC's.
4.	Half subtractor / Full subtractor: Realization using IC's 7400 and 7402.
5.	Realization of IC7483 as Parallel adder/subtractor.
6.	4-Bit Binary-to-Gray and Gray-to-Binary Code Converter: Realization using basic, XOR gates and universal gates.
7.	4-Bit and 8-Bit Comparator: Implementation using IC7485 magnitude comparator chips.
8.	Multiplexer: Truth-table verification and realization of half adder and full adder using IC74153 chip.
9.	Demultiplexer: Truth-table verification and realization of half subtractor and full subtractor using IC74139 chip.
10.	Flip Flops: Truth-table verification of JK master slave FF, T-type and D-type FF using IC7476 chip.
11.	Asynchronous Counter: Realization of 4-bit up counter and Mod-N counter using IC7490 & IC7493 chip.
12.	Synchronous Counter: Realization of 4-bit up/down counter and Mod-N counter using IC74192 and IC74193 chip.

A mini project such as burglar alarm, fire alarm, traffic alert system etc.

Reference Material

Manuals available in Lab.

Subject Code: PRCS-101

Subject Name: Seminar and Technical Report Writing for Engineers

Programme: B.Tech. (CSE)	**L: 0 T: 0 P:** 2
Semester: 3	**Teaching Hours:** 24
Theory/Practical: Practical	**Credits:** 1
Internal Marks: 50	**Percentage of Numerical/Design/Programming Problems:** Nil
External Marks: Nil	**Duration of End Semester Exam (ESE):**
Total Marks: 50	**Elective Status:** Compulsory

Additional Material Allowed in ESE: NA

On completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1.	Understand the basic components of definitions, descriptions, process explanations, and other common forms of technical writing.
2.	Analyze and critique various speech techniques, content, purpose, strengths and weaknesses.
3.	Reference and quote correctly, and not infringe copyright.
4.	Practice the unique qualities of professional rhetoric, writing and presentation style.
5.	Use a technical report to communicate information.
6.	Follow the stages of the writing process (prewriting/writing/rewriting) and apply them to technical and workplace writing tasks.

Part-A

Technical report: Importance of technical report, Structure of technical report, Planning the report, Diagrams, graphs, tables and mathematics, Citing and referencing, References to diagrams, graphs, tables and equations, Originality and plagiarism, Finalising the report and proofreading. [**2 Hours**]

Presentation Skills: Plan, structure and prepare presentation, Voice and body language, Conversational style, Rules for effective speaking, Enhancing improvising skills, Managing a challenging audience, Using visuals and metaphors to make presentation more inspiring. **[2 Hours]**

Latex: Installation of the software LaTeX, Latex compilation, Creating a document- preamble of a document, Basic formatting- abstract, paragraphs and newlines, Headers and footers, spacing, hyphenation, Bold, italics and underlining text, Paragraph alignment and indentation, Lists- Unordered, ordered, nested **[3 Hours]**

Part-B

Tables: The tabular environment, Tables with fixed length, Combining rows and columns, Multi-page tables, Positioning tables, Captions, labels and references, Line width and cell padding, colors, Sideways tables, Table with legend. **[3 Hours]**

Figures: Insertion, Changing the image size and rotating the picture, captions, lists of figures and tables, Captions, labels and references, Wrapping text around figures, Subfigures, Wide figures in two-column documents. **[3 Hours]**

References: Bibliography management in LaTeX, Creating a .bib file, citation styles, citation of references in the text. **[2 Hours]**

Text Books

7. John Seely. The Oxford Guide to Effective Writing and Speaking. Oxford University Press.
8. Frank Mittelbach , Michel Goossens, Johannes Braams, David Carlisle, Chris Rowley, "The LaTeX Companion (Tools and Techniques for Computer Typesetting)", Addison-Wesley.
9. Stefan Kottwitz, "LaTeX Beginner's Guide", PACKT.

Reference Books

1. Davies J.W., "Communication for Engineering Students", Longman.
2. Van Emden J., "Effective communication for Science and Technology", Palgrave.
3. Van Emden J., "A Handbook of Writing for Engineers", Macmillan.
4. Van Emden J. and Easteal J., "Technical Writing and Speaking, an Introduction", McGraw-Hill.
5. Pfeiffer W.S., "Pocket Guide to Technical Writing", Prentice Hall.
6. Eisenberg A., "Effective Technical Communication", McGraw-Hill.

E-Books and online learning material

1. AH Basson & TW von Backström, "Guide for Writing Technical Reports",3rd Edition, Stellenbosch University",2007 Accessed on Feb 23, 2019

2. "Introduction to LaTex", http://home.iitk.ac.in/~kalpant/docs/intro_latex.pdf Accessed on Feb 23,2019

Online Courses and Video Lectures

1. "Technical Report Writing for engineers", https://www.futurelearn.com/courses/technicalreport-writing- for-engineers. Accessed on April 26th 2019.
2. "Technical Writing", https://www.coursera.org/learn/technical-writing. Accessed on April 26th 2019.

Subject Code: LPCCS-101

Subject Name: Object Oriented Programming Laboratory

Programme: B.Tech. (CSE)	**L:** 0 **T:** 0 **P:** 4
Semester: 3	**Teaching Hours:** 42
Theory/Practical: Practical	**Credits:** 2
Internal Marks: 30	**Percentage of Numerical/Design/Programming Problems:**100%
External Marks: 20	**Duration of End Semester Exam (ESE):** 2 hrs
Total Marks: 50	**Elective Status:** Compulsory

Prerequisites: NIL

On Completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1	Compare and contrast object oriented programming paradigm with procedure oriented programming paradigm.
2	Design and implement efficient programs to solve computing problems in a high level programming language.
3	Utilize knowledge of different object oriented principles to identify and apply the appropriate techniques in problem solving.
4	Apply the knowledge acquired to troubleshoot programming related problems.
5	Utilize the knowledge and principles of object oriented programming while working in multidisciplinary teams.
6	Design and develop projects using object oriented tools and techniques.

[Control statements]

1. Demonstrate the use of conditional control statements like if, if-else, if-else ladder, nested if-else, and switch-case statement.
2. Illustrate the use of loop control statements like for, while, and do-while.
3. Write a program to demonstrate the use of break and continue statement.

[Arrays and Strings]

4. Demonstrate the use of one dimensional and two dimensional arrays by using suitable programs.
5. Illustrate the use of various string handling functions.

[Classes and Objects]

6. Program to illustrate the concept of classes and object.
7. Program to illustrate the concept of nesting of member functions.
8. Program to show the working of static members (static functions and static variables) in a class.
9. Program to demonstrate the use of friend functions.

[Constructors and Destructors]

10. Program to illustrate the concept of default constructor, parameterized constructor, and copy constructor.
11. Program to illustrate the concept of destructors.

[Polymorphism]

12. Program to demonstrate the concept of operator overloading
13. Program to illustrate the concept of function overloading and constructor overloading.
14. Program to illustrate the concept of virtual functions and pure virtual functions.

[Inheritance]

15. Program to illustrate the concept of inheritance.
16. Program to illustrate the concept of ambiguity in multiple inheritance.
17. Program to illustrate the order of execution of constructors and destructors in inheritance.
18. Program to demonstrate the concept of function overriding.

[Exception handling]

19. Program to illustrate the exception handling mechanism.

[File handling]

20. Program to illustrate the concept of file pointers.

21. Program to perform read and write operations on a file.

Any one project

Banking System Project

Description: The BANKING SYSTEM project has account class with data members like account number, name, deposit, withdraw amount and type of account. Customer data is stored in a binary file. A customer can deposit and withdraw amount in his account. User can create, modify and delete account.

Library Management System Project

The LIBRARY MANAGEMENT SYSTEM project has book and student class with data members like book no, bookname, authorname. Books record is stored in a binary file. A student can issue book and deposit it within 15 days. Student is allowed to issue only one book. Student Records are stored in binary file. Administrator can add, modify or delete record.

Reference Material

Manuals available in Lab

Subject Code: LPCCS-102

Subject Name: Computer Networks Laboratory

Programme: B.Tech. CSE	**L:** 0 **T:** 0 **P:** 2
Semester: 3	**Teaching Hours:** 24
Theory/Practical: Practical	**Credits:** 01
Internal Marks: 30	**Percentage of Numerical/Design/Programming Problems:**100%
External Marks: 20	**Duration of End Semester Exam (ESE):** 2 hrs
Total Marks: 50	**Elective Status:** Compulsory

Prerequisites: Fundamentals of Computer System.

On Completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1	Analyze and configure protocols concerning various network technologies over different mediums and layers.
2	Apply the knowledge of different network components, transmission mediums and tools to solve various problems of communication.
3	Design and develop different network design and logical models of networking to solve network related problems.
4	Utilize knowledge of modern network simulation tools to propose solution for efficient working of networks for real world problems.
5	Make use of various troubleshooting methods to overcome networking problems.
6	Function in multidisciplinary teams through groups while working in different network environments with the help of resource sharing.

Special Instruction related to resources requirement: Except practical number 10.

Sr.No.	Name of Practical
1.	Familiarization with networking components, transmission media, tools and devices: LAN Adapters, Hubs, Switches, Routers etc.
2.	Study of various LAN topologies and their creation using network devices, cables and

	computers, Preparing straight and cross cables.
3.	Configuration of TCP/IP Protocols in Windows and Linux.
4.	Implementation of resource (file, printer, etc.) sharing.
5.	Designing and implementing class A, B and C networks.
6.	Subnet planning and its implementation.
7.	To configure dynamic IP address for a computer connected to a LAN.
8.	Use of commands like ping, ipconfig for trouble shooting network related problems.
9.	Develop a program to compute the Hamming Distance between any two code words.
10.	To configure proxy server, Familiarization with network simulation tools.

Reference Material

Manuals available in

Subject Code: LESCS-101

Subject Name: Digital Electronics Laboratory

Programme: B.Tech. (CSE)	**L:** 0 **T:** 0 **P:**2
Semester: 3	**Teaching Hours: 24**
Theory/Practical: Practical	**Credits:** 1
Internal Marks: 30	**Percentage of Numerical/Design/Programming Problems:**100%
External Marks: 20	**Duration of End Semester Exam (ESE):** 2 hrs
Total Marks: 50	**Elective Status:** Compulsory

Prerequisites: NIL

On Completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1	Identify and apply the knowledge of logic gates and integrated circuits to solve related problems.
2	Design and implement combinational & sequential circuits for engineering problems.
3	Choose and compare the usage of appropriate techniques and tools to solve digital circuits problem.
4	Apply the knowledge acquired to demonstrate the usage of digital circuits in computers at large.
5	Utilize the knowledge and principles of digital electronics while working in multidisciplinary team formation.
6	Design simple digital systems based on these digital abstractions, using the "digital paradigm".

Resources required: Digital Trainer Board, ICs, connecting wires.

S. No.	Name of Practical
1.	Truth-table verification of OR, AND, NOT, XOR, NAND and NOR gates using various IC's.
2.	Realization of OR, AND, NOT and XOR functions using universal gates IC's 7400 and 7402.

3.	Half adder / Full adder: Realization using basic and XOR gates IC's.
4.	Half subtractor / Full subtractor: Realization using IC's 7400 and 7402.
5.	Realization of IC7483 as Parallel adder/subtractor.
6.	4-Bit Binary-to-Gray and Gray-to-Binary Code Converter: Realization using basic, XOR gates and universal gates.
7.	4-Bit and 8-Bit Comparator: Implementation using IC7485 magnitude comparator chips.
8.	Multiplexer: Truth-table verification and realization of half adder and full adder using IC74153 chip.
9.	Demultiplexer: Truth-table verification and realization of half subtractor and full subtractor using IC74139 chip.
10.	Flip Flops: Truth-table verification of JK master slave FF, T-type and D-type FF using IC7476 chip.
11.	Asynchronous Counter: Realization of 4-bit up counter and Mod-N counter using IC7490 & IC7493 chip.
12.	Synchronous Counter: Realization of 4-bit up/down counter and Mod-N counter using IC74192 and IC74193 chip.

A mini project such as burglar alarm, fire alarm, traffic alert system etc.

Reference Material

Manuals available in Lab.

Subject Code: PRCS-101

Subject Name: Seminar and Technical Report Writing for Engineers

Programme: B.Tech. (CSE)	**L: 0 T: 0 P:** 2
Semester: 3	**Teaching Hours:** 24
Theory/Practical: Practical	**Credits:** 1
Internal Marks: 50	**Percentage of Numerical/Design/Programming Problems:** Nil
External Marks: Nil	**Duration of End Semester Exam (ESE):**
Total Marks: 50	**Elective Status:** Compulsory

Additional Material Allowed in ESE: NA

On completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1.	Understand the basic components of definitions, descriptions, process explanations, and other common forms of technical writing.
2.	Analyze and critique various speech techniques, content, purpose, strengths and weaknesses.
3.	Reference and quote correctly, and not infringe copyright.
4.	Practice the unique qualities of professional rhetoric, writing and presentation style.
5.	Use a technical report to communicate information.
6.	Follow the stages of the writing process (prewriting/writing/rewriting) and apply them to technical and workplace writing tasks.

Part-A

Technical report: Importance of technical report, Structure of technical report, Planning the report, Diagrams, graphs, tables and mathematics, Citing and referencing, References to diagrams, graphs, tables and equations, Originality and plagiarism, Finalising the report and proofreading. [**2 Hours**]

Presentation Skills: Plan, structure and prepare presentation, Voice and body language, Conversational style, Rules for effective speaking, Enhancing improvising skills, Managing a challenging audience, Using visuals and metaphors to make presentation more inspiring. **[2 Hours]**

Latex: Installation of the software LaTeX, Latex compilation, Creating a document- preamble of a document, Basic formatting- abstract, paragraphs and newlines, Headers and footers, spacing, hyphenation, Bold, italics and underlining text, Paragraph alignment and indentation, Lists- Unordered, ordered, nested **[3 Hours]**

Part-B

Tables: The tabular environment, Tables with fixed length, Combining rows and columns, Multi-page tables, Positioning tables, Captions, labels and references, Line width and cell padding, colors, Sideways tables, Table with legend. **[3 Hours]**

Figures: Insertion, Changing the image size and rotating the picture, captions, lists of figures and tables, Captions, labels and references, Wrapping text around figures, Subfigures, Wide figures in two-column documents. **[3 Hours]**

References: Bibliography management in LaTeX, Creating a .bib file, citation styles, citation of references in the text. **[2 Hours]**

Text Books

1. John Seely. The Oxford Guide to Effective Writing and Speaking. Oxford University Press.
2. Frank Mittelbach , Michel Goossens, Johannes Braams, David Carlisle, Chris Rowley, "The LaTeX Companion (Tools and Techniques for Computer Typesetting)", Addison-Wesley.
3. Stefan Kottwitz, "LaTeX Beginner's Guide", PACKT.

Reference Books

1. Davies J.W., "Communication for Engineering Students", Longman.
2. Van Emden J., "Effective communication for Science and Technology", Palgrave.
3. Van Emden J., "A Handbook of Writing for Engineers", Macmillan.
4. Van Emden J. and Easteal J., "Technical Writing and Speaking, an Introduction", McGraw-Hill.
5. Pfeiffer W.S., "Pocket Guide to Technical Writing", Prentice Hall.
6. Eisenberg A., "Effective Technical Communication", McGraw-Hill.

E-Books and online learning material

1. AH Basson & TW von Backström, "Guide for Writing Technical Reports",3rd Edition, Stellenbosch University",2007 Accessed on Feb 23, 2019

2. "Introduction to LaTex", http://home.iitk.ac.in/~kalpant/docs/intro_latex.pdf Accessed on Feb 23,2019

Online Courses and Video Lectures

1. "Technical Report Writing for engineers", https://www.futurelearn.com/courses/technicalreport-writing- for-engineers. Accessed on April 26th 2019.
2. "Technical Writing", https://www.coursera.org/learn/technical-writing. Accessed on April 26th 2019.

Subject Code: PCCS-103

Subject Name: Discrete Mathematics

Programme: B.Tech. (CSE)	**L:** 3 **T:** 1 **P: 0**
Semester: 4	**Teaching Hours:** 35
Theory/Practical: Theory	**Credits:** 4
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 90%
External Marks: 60	**Duration of End Semester Exam (ESE):** 3 hrs
Total Marks: 100	**Elective Status:** Compulsory

Prerequisites: Familiarization with sequences and series, sets, functions and derivatives.

Additional Material Allowed in ESE: [Scientific Calculator]

On completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1	Apply knowledge of mathematical proofs, techniques and algorithms to solve complex engineering problem.
2	Prove elementary properties of modular arithmetic and explain their application in analysis and interpretation of data and synthesis of information to provide valid conclusions.
3	Create, select and apply appropriate techniques to model real world problems using graphs.
4	Identify and formulate solutions of engineering problems related to counting and probability theory**.**
5	Utilize the importance of discrete structures towards simulation of problems in multidisciplinary environments.
6	Formulate a logical statement in terms of a symbolic expression and evaluate the truth value of compound statement.

Detailed Contents:

Part-A

Fundamentals of Sets, Relations and Functions: Sets – Operations on sets, Subsets, Types of sets, Ordered pairs, Proofs of general identities of sets, Classes of sets and partitions, Countable and uncountable

sets. Relations –Representations of relations, Types of relations, Composition of relations, Closure properties of relations, Equivalence relations, Compatibility relations, Partial order relations. Functions – Introduction and types of functions, Sum and product of functions, Hashing functions, Recursively defined functions. [**7 Hours**]

Prepositional and Predicate Logic: Prepositions and compound prepositions, Logical connectives, Truth tables, Logical implication and logical equivalence, Normal forms– Conjunctive and Disjunctive, Validity of well-formed formula, Propositional inference rules–Modus ponens and modus tollens. Predicate logic, Universal and existential quantification, Limitations of propositional and predicate logic. **[3 Hours]**

Combinatorial Mathematics: Basic counting principles, Permutations and combinations, Pigeonhole principle, Inclusion and exclusion principle, Recurrence relations – Solving homogeneous and non-homogeneous recurrence relations, Sequences, Generating function. **[7 Hours]**

Part-B

Algebraic Structures and Morphism : Algebraic structures with one binary operation, Properties of an operation, Congruence relation , Semi groups, Monoids, Groups, Substructures, Cyclic groups, Cosets, Normal subgroups, Dihedral groups, Permutation Groups. Homomorphism and isomorphism of groups, Applications of groups. Algebraic structures with two binary operation, Rings – Introduction, Abelian ring, Ring with unity, Multiplicative inverse, Subrings, Homomorphism of rings. Integral Domain, Ideals. **[8 Hours]**

Boolean Algebra: Boolean algebra, Boolean sub-algebra, Boolean rings, Application of Boolean algebra (Logic implications, Logic gates, Karnaugh-map). **[2 Hours]**

Graphs and Trees: Graphs – Definition, degree, Connectivity, path, cycle, Directed and undirected, Sub Graph, Bi-connected component and Articulation points. Eulerian chains and cycles, Hamiltonian chains and cycles, Shortest paths algorithms – Dijkstra'salgorithm, Warshall's algorithm. Rooted trees, Spanning tree algorithms – Kruskal's algorithm, Prim's algorithm. Graph coloring, Map Coloring, Chromatic number, Planar graphs, Euler's formula, Isomorphism and homomorphism of graphs, Applications of graph theory. **[8 Hours]**

Text Books

1. S. Lipschutz and M.Lipson, "Schaum's Outline of Discrete Mathematics", Tata McGraw Hill.

2. A. Doerr and K. Levarseur, "Applied Discrete Structures for Computer Science", Pearson Education, Inc.
3. K.H. Rosen, "Discrete Mathematics and its applications", Tata McGraw Hill.

Reference Books

1. C.L. Liu , " Elements of Discrete Mathematics" , Tata McGraw Hill.
2. Susanna S. Epp, Discrete Mathematics with Applications, Wadsworth Publishing Co. Inc.

E-Books and online learning material

1. Discrete Mathematics and its applications by Kenneth H. Rosen
https://mathcs.clarku.edu/~djoyce/ma114/Rosen6E.pdf 6th Edition Accessed on Feb. 27, 2019

2. Discrete Mathematics: An Open Introduction by Oscar Levin
https://open.umn.edu/opentextbooks/textbooks/discrete-mathematics-an-open-introduction Accessed on Feb. 27, 2019

3. A Course in Discrete Structures by Rafael Pass Wei-Lung Dustin Tseng
https://www.cs.cornell.edu/~rafael/discmath.pdf Accessed on Feb. 27, 2019 **Online Courses and Video Lectures**

1 https://nptel.ac.in/courses/106106094 Accessed on Feb. 27, 2019
2. https://nptel.ac.in/courses/111107058 Accessed on Feb. 27, 2019
3. https://nptel.ac.in/courses/111104026 Accessed on Feb. 27, 2019
4. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-042j-mathematics-for- computer-science-fall-2010/video-lectures Accessed on Feb. 27, 2019

Subject Code: PCCS-104

Subject Name: Computer Architecture and Microprocessors

Programme: B.Tech. (CSE)	**L:** 3 **T:** 0 **P:** 0
Semester: 4	**Teaching Hours:** 38
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design Problems:** 20%
External Marks: 60	**Duration of End Semester Exam (ESE):** 3 hrs
Total Marks: 100	**Elective Status:** Compulsory

Prerequisites: Digital Circuit and Logic Design

Additional Material Allowed in ESE: NIL

On Completion of the course, the student will have the ability to:

CO#	Course Outcome
CO1	Identify computer systems, memory organization, Microprocessor and assembly language programming.
CO2	Clarify instruction formats, RISC and CISC architecture and different addressing Modes.
CO3	Solve basic binary math operations by using the instructions of microprocessor.
CO4	Compare between pipelining and parallelism.
CO5	Design structured, well commented, understandable assembly language programs to provide solutions to real-world problems.
CO6	Classify the trends and developments of microprocessor technology.

Detailed Contents

Part-A

Data Representation: Data types, Complements, Fixed point representation, Floating point representation, Error detection and correction. **[3 Hours]**

Register Transfer and Micro-operations: Addition, Subtraction, Multiplication and division algorithms and hardware, Register transfer language and operations, Arithmetic micro-operations, Logic micro-operations, Shift micro-operations, Arithmetic logic shift unit. **[4 Hours]**

Computer Organization and Design: Instruction codes, Computer registers, Computer instructions, Timing and control, Instruction cycle, Memory reference instructions, Input/ Output and interrupts, Design and working of a complete basic computer, Control functions, Design of accumulator logic. **[4 Hours]**

Central Processing Unit and Input-Output Organization: General register organization, Stack organization, Addressing modes, RISC and CISC architecture, I/O interface, Asynchronous data transfer, Modes of transfer, Priority interrupt, DMA, I/O processor. **[4 Hours]**

Memory Organization: Memory hierarchy, Main memory, Auxiliary memory, Associative memory, Cache memory, Virtual memory, Memory management hardware. **[4 Hours]**

Part-B

Microprocessor Architecture: Introduction to microprocessors, 8085 microprocessor architecture – Bus structure, Register organization. **[5 Hours]**

Programming with 8085: Addressing modes, Instruction classification, Instruction formats, Data transfer operations, Arithmetic operations, Logical operations, Branch operations, Stack and subroutine operations, looping, counting and indexing operations. **[5 Hours]**

Interfacing: Memory and I/O mapped I/O, Programmable interfaces – 8255 programmable peripheral interface, 8259 interrupt controller, and 8237 DMA controller. **[5 Hours]**

Microprocessor Applications: Interfacing of keyboards and seven segment LED display, Study of traffic light system, stepper motor controller. **[4 Hours]**

Text Books:

1. M. Morris Mano, "Computer System Architecture", Pearson Education.
2. William Stallings, "Computer Organization and Architecture", Pearson Education.
3. Ramesh Gaonkar, "Microprocessor Architecture, Programming, and Applications with 8085", Penram International Publication.

Reference Books:

1. B. Ram, "Microprocessors and Microcomputers", Dhanpat Rai Publications.
2. K. Vani, "Computer Architecture with MIPS", Notion Press.
3. A.P. Mathur, "Introduction to Microprocessors", Tata McGrawHill.

4. P.K. Ghosh and P.R. Sridhar, “0000 To 8085: Introduction to Microprocessors for Engineers and Scientists”, PHI Learning.

Books and online learning material:

1. Computer Architecture by Wikipedia
   https://en.wikipedia.org/wiki/Computer_architecture Accessed on Feb 22, 2019
2. Computer Architecture by Princeton University
   https://www.coursera.org/learn/comparch Accessed on Feb 22, 2019

Online Courses and Video Lectures:

1. https://www.youtube.com/watch?v=4TzMyXmzL8M Accessed on Feb 22, 2019.

2.https://www.youtube.com/watch?v=So9SR3qpWsM Accessed on Feb 20, 2019

Subject Code: PCCS-105

Subject Name: Operating Systems

Programme: B.Tech. (CSE)	**L:** 3 **T: 1 P:** 0
Semester: 4	**Teaching Hours:** 36
Theory/Practical: Theory	**Credits:** 4
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 20%
External Marks: 60	**Duration of End Semester Exam (ESE):** 3 hrs
Total Marks: 100	**Elective Status:** Compulsory

Prerequisites: Basic knowledge of computer fundamentals and computer system architecture.

Additional Material Allowed in ESE:

On completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1	Understand the mechanisms of OS to handle processes and threads and their communication.
2	Compare and contrast the mechanisms involved in memory management Techniques
3	Use the components of Operating System in OS design
4	Evaluate different scheduling Techniques.
5	Investigate basic concepts towards process synchronization and related issues.
6	Understand the structure and organization of file system.

Detailed Contents:

Part - A

Introduction: Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS-Layered, Monolithic, Microkernel Operating

Systems, Case study on UNIX and WINDOWS Operating System. **[5 Hours]**

Process management: Concept of processes and threads, Definition, Process and Program, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching Thread: Definition, Various states, Benefits of threads. **[4 Hours]**

Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling algorithms: Pre-emptive and Non pre-emptive, FCFS, SJF, RR; Multiprocessor scheduling: Real Time scheduling. **[6 Hours]**

Inter-process Communication: Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution, Strict Alternation, Peterson's Solution, The Producer\ Consumer Problem, Semaphore **[4 Hours]**

Part - B

Deadlocks: Introduction to deadlocks, Conditions for deadlock, Resource allocation graphs, Deadlock prevention and avoidance, Deadlock detection and recovery. **[4 Hours]**

Memory Management: Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation – Fixed and variable partition–Internal and External fragmentation and Compaction; Paging: Principle of operation – Page allocation – Hardware support for paging, Protection and sharing, Disadvantages of paging, Segmentation. **[6 Hours]**

File Management: Concept of File, Access methods, File types, File operation, Directory structure, Allocation methods (contiguous, linked, indexed), Free-space management (bit vector, linked list, grouping), directory implementation (linear list, hash table), efficiency and performance. **[4 Hours]**

Secondary Storage: Disk structure**,** Disk scheduling – FCFS, SSTF, SCAN, C-SCAN, LOOK, C-LOOK, Disk Management, Disk Formatting, Boot blocks, Bad blocks. **[4 Hours]**

Text Books

1. A Silberschatz and Peter B. Galvin, "Operating System Concepts" Addison Wesley.
2. GaryNutt, "Operating Systems Concepts", Pearson Education Ltd.

Reference Books

1. Dhamdhere, “Systems Programming & Operating Systems” Tata McGraw Hill.

2. Tanenbaum A. S “Operating System Design & Implementation” Pearson Education.

3. Bhatt and Chandra “An introduction to operating systems concepts & Practices” Prentice Hall of India Publication.

E-Books and online learning material

1. http://www.uobabylon.edu.iq/download/M.S%202013-2014/Operating_System_Concepts,_8th_Edition%5BA4%5D.pdf Accessed on Aug. 05, 2019
2. http://dinus.ac.id/repository/docs/ajar/Operating_System.pdf Accessed on Aug. 05, 2019

Online Courses and Video Lectures

1. https://nptel.ac.in/courses/106106144/ Accessed on Aug. 05, 2019
2. https://www.coursera.org/learn/os-power-user Accessed on Aug. 05, 2019

Subject Code: PCCS-106

Subject Name: Data Structures

Programme: B.Tech. (CSE)	**L:** 3 **T:** 0 **P:** 0
Semester: 4	**Teaching Hours:** 37
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 50%
External Marks: 60	**Duration of End Semester Exam (ESE):** 3hrs
Total Marks: 100	**Elective Status:** Compulsory

Prerequisites: Knowledge of Programming for Problem Solving and OOPS

Additional Material Allowed in ESE: [Scientific Calculator]

On completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1	Apply knowledge of statistics and programming skills to solve complex engineering problems related to data structures.
2	Make use of Research based knowledge to identify the appropriate data structure and provide better solution to reduce space and time complexity.
3	Identify, Formulate and analyse data structure to develop skills and understand their applications to perform operations on it.
4	Design appropriate algorithm for autonomous realization of sub-programs to model complex engineering activities.
5	Demonstrate various methods of organizing large amounts of data and recognize systematic way to retrieve data and solve problems.
6	Formulate new solutions for programming problems or improve existing code using learned algorithms and data structures.

Detailed Contents:

Part-A

Basic concepts: Concept of data type, Linear and non-linear data structures, Data structures versus data types, Operations on data structures, Algorithm complexity and Asymptotic notations. **[2 Hours]**

Arrays: Linear and multi-dimensional arrays and their representation, Operations on arrays, Sparse matrices and their storage. **[2 Hours]**

Stacks: Sequential representation of stacks, Operations on stacks, Application of stacks – parenthesis checker, Evaluation of postfix expressions, Conversion from infix to postfix, Conversion from infix to prefix representation, Tower of Hanoi problem, implementingrecursive functions. **[5 Hours]**

Queues: Sequential representation of queue, Types of queue- Linear Queue, Circular Queue, Deque, Priority Queue, Operations on each types of Queues and their algorithms, Applications of Queues. **[4 Hours]**

Linked List: Definition and representation of Linked list, Types of Linked list- Linear linked list, Doubly linked list, Circular linked list and Header linked list and their operations, Application of linked lists, Garbage collection and compaction, Linked representation of Stack and Queues and their algorithm. **[6 Hours]**

Part-B

Trees: Basic terminology, Sequential and linked representations of trees, Different types of Trees- Binary Tree, Binary search tree, Threaded binary tree, AVL tree and B-tree. Operations on each of the trees. Application of Binary Trees. **[5 Hours]**

Graphs: Basic terminology, Representation of graphs – Adjacency matrix, Adjacency list. Operations on graph, Traversal of a graph – Breadth first search, Depth first search. Shortest path algorithms – Dijkstra's and Floyd. Minimum spanning tree – Prim and Kruskal. Applications of graphs. **[4 Hours]**

Heaps: Representing a heap in memory, Operations on heaps, Application of heap in implementing priority queue and Heap sort algorithm. **[2 Hours]**

Hashing and Hash Tables: Introduction to hash table, Hash functions, Concept of collision and its resolution using open addressing and separate chaining, Double hashing, Rehashing. **[2 Hours]**

Searching and Sorting: Linear and binary search techniques, Sorting methods – Bubble sort, Selection sort, Insertion sort, Quick sort, Merge sort, Shell sort and radix sort. Complexities of searching and sorting algorithms. **[5 Hours]**

Text Books

1. Seymour Lipschutz, "Data Structures", Schaum's Outline Series, Tata McGraw Hill.
2. Sartaj Sahni, "Data Structures, Algorithms and Applications in C++", Tata McGraw Hill.

Reference Books

1. Michael T. Goodrich, Roberto Tamassia, & David Mount, "Data Structures and Algorithms in C++", Wiley India.
2. Kruse, "Data Structures &ProgramDesign", Prentice Hall of India.
3. Y. Langsa, M.J. Augenstein, A.M. Tanenbaum, "Data structures using C and C++", Prentice Hall of India.
4. Vishal Goyal, LaliGoyal, Pawan Kumar, " Simplified Approach to Data Structures", Shroff Publications and Distributors

E-Books and online learning material

1. Data Structures and Algorithms: byGranville Barnett, and Luca Del Tongo. https://apps2.mdp.ac.id/perpustakaan/ebook/Karya%20Umum/Dsa.pdf
2. Data Structures and Algorithms in JAVA :by Michael T. Goodrich and Roberto Tamassiahttp://enos.itcollege.ee/~jpoial/algorithms/GT/Data%20Structures%20and%20Algorithms%20in%20Java%20Fourth%20Edition.pdf

Online Courses and Video Lectures

1. https://nptel.ac.in/courses/106102064/
2. https://nptel.ac.in/courses/106106133/
3. https://nptel.ac.in/courses/106106145/
4. https://www.youtube.com/watch?reload=9&v=YWnBbNj_G-U

Subject Code: PCCS-107

Subject Name: Software Engineering

Programme: B.Tech. (CSE)	**L:** 3 **T:** 1 **P:** 0
Semester:	**Teaching Hours:** 36
Theory/Practical: Theory	**Credits:** 4
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 10%
External Marks: 60	**Duration of End Semester Exam (ESE):** 3 hrs
Total Marks: 100	**Elective Status:** Compulsory

Prerequisites: Knowledge of System Analysis

Additional Material Allowed in ESE: NIL

On completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1	Plan a software engineering process life cycle, including the specification, design, and implementation.
2	Elicit, analyze and specify software requirements through a productive working relationship with various stakeholders of the project.
3	Analyze and translate a specification into a design, and then realize that design practically, using an appropriate software engineering methodology.
4	Develop the code from the design and effectively apply relevant standards for quality management and practice.
5	Formulate a testing strategy for a software system, employing techniques such as unit testing, test driven development and functional testing.
6	Identify modern engineering tools necessary for software reengineering and reverse engineering.

Detailed Contents:

Part-A

Introduction: Evolution and impact of software engineering, Software myths, Software application domains, Software crisis – Problem and causes. **[4 Hours]**

Software Process Models: Software process, Software process models – Waterfall model, Prototype model, Spiral model, Evolutionary model, RAD model, V-model and Component based model. **[4 Hours]**

Requirements Engineering: Feasibility study, Problem analysis, Requirement elicitation and specification, Functional and non-functional requirements, Software requirements specification document, Requirement validation and management. **[4 Hours]**

Project Management and Risk Analysis: Project planning, Cost estimation techniques– Size metrics, Empirical estimation, Heuristic estimation and analytical estimation, Project monitoring and control– Work breakdown structure, Activity chart, Gantt charts, PERT charts, Critical path method, Manpower management, Risk management- Identification, Analysis, Planning and Monitoring. **[6 Hours]**

Part-B

Software Design: Modular design– Coupling, Cohesion and abstraction, Function oriented design– Data flow diagrams, Structure chart, Object oriented design–Objects and object classes, Relationships between classes, User interface design. **[7 Hours]**

Coding & Testing: Coding standards and code reviews, Testing – Need of testing, Unit testing, Integration testing, System testing, White-Box testing, Black-box testing, Alpha, Beta and acceptance testing, Smoke testing, Sanity testing, Regression testing, Cyclometric Complexity. Verification and validation. **[6 Hours]**

Maintenance and Re-engineering: Software maintenance, Software re-engineering, Reverse engineering, Forward engineering, PSP and Six sigma. **[5 Hours]**

Text Books

1. Roger S. Pressman R., "Software Engineering, A Practitioner's Approach", McGraw Hill International.
2. Rajib Mall, "Fundamentals of Software Engineering", Prentice Hall of India.

Reference Books

1. Ian Sommerville , Software Engineering, Addison-Wesley Publishing Company.
2. Jalote P., "An Integrated Approach to Software Engineering", Naros

E-Books and online learning material

1. https://nptel.ac.in/courses/Webcourse contents/IIT%20Kharagpur/Soft%20Engg/New_index1.html Accessed on March 1, 2019
2. https://nptel.ac.in/downloads/106105087/ Accessed on March 1, 2019

Online Courses and Video Lectures

https://onlinecourses.nptel.ac.in/noc18_cs43/preview Accessed on March 1, 2019

Subject Code: LPCCS-103

Subject Name: Computer Architecture and Microprocessors Laboratory

Programme: B.Tech.(CSE)	**L:** 0 **T:** 0 **P:** 2
Semester: 4	**Teaching Hours:** 24
Theory/Practical: Practical	**Credits:** 1
Internal Marks: 30	**Percentage of Numerical/Design/Programming Problems:**100%
External Marks: 20	**Duration of End Semester Exam (ESE):** 2 hrs
Total Marks: 50	**Elective Status:** Compulsory

Prerequisites: Fundamentals of Computers.

On Completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1	Utilize the concept of binary & hexadecimal number systems including computer arithmetic.
2	Demonstrate the error detection & correction mechanism in computer architecture.
3	Understand the functional units of the processor such as the register file and arithmetic logical unit.
4	Examine the fundamentals of assembly language programming.
5	Understand the concept of computer arithmetic instruction set by designing code for arithmetic, logical and data transfer operations.
6	Solve basic binary math operations by using the instructions of microprocessor.

Special Instruction related to resources requirement: GNUsim8085 simulator should be installed to perform 4 to 11 practicals.

Sr.No.	Name of Practical
1.	**Combinational Circuit:** To study Half Adder.

2.	**Combinational Circuit:** To study Full Adder (7483).
3.	**Register Transfer and Micro-operations:** Write a program for binary multiplication.
4.	**Memory Organisation:** Write programs to simulate the mapping techniques of Cache memory. a. Direct Mapped cache b. Associative Mapped cache c. Set Associative Mapped cache
5.	**Functional design of ALU:** Analyzing the architecture and design of ALU, study the working of ALU and examine its functionality.
6.	**Working of 8085 simulator GNUsim8085 :** Introduction to 8085 microprocessor, study of components of GNUsim8085 and step wise assembly program execution using it.
7.	**Complete instruction set of 8085 :** Representation of instruction in computer system, Types of instructions - control instructions, logical instructions, branching, arithmetic and data transfer instructions.
8.	**Data transfer instructions:** Assembly language code in GNUsim8085 to implement various data transfer instructions like MOV, MVI, LXI, LDA, STA, IN, OUT etc.
9.	**Arithmetic instructions:** Assembly language code in GNUsim8085 to implement various arithmetic instructions involving immediate addition, subtraction, increment, decrement operations.
10.	**Addition and subtraction using flag:** assembly language code in GNUsim8085 to add and subtract two 8 bit numbers stored in memory and also set various flags corresponding to the result.
11.	**Logical instructions:** Assembly language code in GNUsim8085 to implement various logical instructions involving comparing of memory and register contents, logical operations - AND, OR, XOR and rotate operations.
12.	**Stack and branch instructions:** Assembly language code in GNUsim8085 to implement

	various stack and branch instructions involving insertion and deletion of contents into stack, conditional and unconditional jump, call and return.

Reference Material

Manuals available in Lab.

Subject Code: LPCCS-104

Subject Name: Operating Systems Laboratory

Programme: B.Tech. (CSE)	**L:** 0 **T:** 0 **P:** 2
Semester: 4	**Teaching Hours:** 26
Theory/Practical: Practical	**Credits:** 1
Internal Marks: 30	**Percentage of Numerical/Design/Programming Problems:**100%
External Marks: 20	**Duration of End Semester Exam (ESE): 2** hrs
Total Marks: 50	**Elective Status:** Compulsory

Prerequisites: Fundamentals of Computers.

On Completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1	Analyse the services, architectures and principles used in the design of modern operating systems.
2	Execute Linux commands for files and directories, creating and viewing files, File comparisons and Disk related commands.
3	Utilize the concept of virtualization for creating a virtual machine and installing operating system on virtual machine.
4	Demonstrate shell programming by using shell variables and shell keywords for automated system tasks.
5	Identify the key characteristics of multiple approaches used for the design and development of the operating system.
6	Apply system commands for performing the file manipulation, program execution, and printing text.

Special Instruction related to resources requirement: Any programming language like C, C++, Java can be used to simulate the programs.

Sr.No.	Name of Practical
1.	Installation process of various Operating Systems.

2.	Virtualization, Installation of virtual machine software and installation of Operating System on virtual machine.
3.	Overview of single user systems, network operating system and multiuser system.
4.	Write a program for the simulation of following CPU scheduling algorithms to find turnaround time and waiting time. a) FCFS b) SJF c) Round Robin d) Priority
5.	Write a program for the simulation of producer-consumer problem using semaphores.
6.	Write a program for the simulation of Banker's algorithm for the purpose of deadlock avoidance.
7.	Write a program for the simulation of following contiguous memory allocation techniques a) Worst-fit b) Best-fit c) First-fit
8.	Write a program for the simulation of following page replacement algorithms a) FIFO b) LRU c) Optimal
9.	Write a program for the simulation of following disk scheduling algorithms a) FCFS b) SCAN c) C-SCAN
10.	Write a program for the simulation of following file allocation strategies a) Sequential b) Indexed c) Linked
11.	To study the features of Windows and Linux operating system.
12.	Execute various basic Linux commands, commands for files and directories, creating and viewing files, File comparisons, Disk related commands.
13.	Basics of Shell programming, various types of shell, Shell Programming in bash.
14.	Implement conditional statements, looping statement, case statements and functions in Shell programming

Reference Material

Manuals available in Lab.

Subject Code: LPCCS-105

Subject Name: Data Structures Laboratory

Programme: B.Tech. (CSE)	**L:** 0 **T:** 0 **P:** 4
Semester: 4	**Teaching Hours:** 40
Theory/Practical: Practical	**Credits:** 2
Internal Marks: 30	**Percentage of Numerical/Design/Programming Problems:**100%
External Marks: 20	**Duration of End Semester Exam (ESE):** 2 hrs
Total Marks: 50	**Elective Status:** Compulsory

Prerequisites: Fundamentals of Computers.

On Completion of the course, the student will have the ability to:

CO#	Course Outcomes(CO)
1	Apply knowledge of mathematics and programming skills to implement and analyze different data structures.
2	Evaluate and analyze the time and space complexity of linear and non linear data structures.
3	Design and implement efficient algorithms to solve computing problems in a high level programming language.
4	Utilize knowledge of different data structures to identify and apply the appropriate data structures to solve a real world problem.
5	Compare and analyze different solutions of complex engineering activities with an understanding of their advantages and limitations.
6	Developing an awareness of the data structure for storing data and handling various operations on different applications in the broadest context of technology change.

Special Instruction related to resources requirement: Any programming language like C, C++, can be used for the programs.

Sr.No.	Name of Practical
1.	Design, Develop and Implement a menu driven Program for the following Array operations a. Creating an Array of N Integer Elements b. Display of Array Elements with Suitable Headings c. Inserting an Element (ELEM) at a given valid Position (POS) d. Deleting an Element at a given valid Position(POS) e. Exit.
2.	Design, Develop and Implement a menu driven Program for the following operations on STACK of Integers (Array Implementation of Stack with maximum size MAX) a. Push an Element on to Stack b. Pop an Element from Stack c. Demonstrate how Stack can be used to check Palindrome d. Demonstrate Overflow and Underflow situations on Stack Display the status of Stack f. Exit Support the program with appropriate functions for each of the above operations
3.	Design, Develop and Implement a Program for converting an Infix Expression to Postfix Expression. Program should support for both parenthesized and free parenthesized expressions with the operators: +, -, *, /, %(Remainder), ^(Power) and alphanumeric operands.
4.	Design, Develop and Implement a Program for the following Stack Applications a. Evaluation of Suffix expression with single digit operands and operators: +, -, *, /, %, ^ b. Solving Tower of Hanoi problem with n disks
5.	Design, Develop and Implement a menu driven Program for the following operations on Circular QUEUE of Characters (Array Implementation of Queue with maximum size MAX) a. Insert an Element on to Circular QUEUE b. Delete an Element from Circular QUEUE c. Demonstrate Overflow and Underflow situations on Circular QUEUE d. Display the status of Circular QUEUE

	e. Exit Support the program with appropriate functions for each of the above operations .
6.	Design, Develop and Implement a menu driven Program for the following operations on Singly Linked List (SLL) of Student Data with the fields: USN, Name, Branch, Sem, PhNo a. Create a SLL of N Students Data by using front insertion. b. Display the status of SLL and count the number of nodes in it c. Perform Insertion / Deletion at End of SLL d. Perform Insertion / Deletion at Front of SLL(Demonstration of stack) e. Exit
7.	Design, Develop and Implement a menu driven Program for the following operations on Doubly Linked List (DLL) of Employee Data with the fields: SSN, Name, Dept, Designation, Sal, PhNo a. Create a DLL of N Employees Data by using end insertion. b. Display the status of DLL and count the number of nodes in it c. Perform Insertion and Deletion at End of DLL d. Perform Insertion and Deletion at Front of DLL e. Demonstrate how this DLL can be used as Double Ended Queue f. Exit
8.	Design, Develop and Implement a Program for the following operationson Singly Circular Linked List (SCLL) with header nodes a. Find the sum of two polynomials POLY1(x,y,z) and POLY2(x,y,z) and store the result in POLYSUM(x,y,z)
9.	Design, Develop and Implement a menu driven Program for the following operations on Binary Search Tree (BST) of Integers a. Create a BST of N Integers: 6, 9, 5, 2, 8, 15, 24, 14, 7, 8, 5, 2 b. Traverse the BST in Inorder, Preorder and Post Order c. Search the BST for a given element (KEY) and report the appropriate message e. Exit
10.	Design, Develop and Implement a Program for the following operations on Graph(G) of Cities

	a. Create a Graph of N cities using Adjacency Matrix. b. Print all the nodes reachable from a given starting node in a digraph using DFS/BFS method
11.	Write a Program to finds the position of an element in an array using Linear Search Algorithm and Binary search Algorithm.
12.	Write a program to sort list using different sorting algorithms (bubble, selection, insertion, radix, merge and quick sort) and compare them.

Reference Material

Manuals available in Lab.

Subject Code: MCCS-101

Subject Name: Environmental Sciences

Programme: B.Tech. (CSE)	**L:** 2 **T: 0 P:** 0
Semester: 4	**Teaching Hours:** 28
Theory/Practical: Theory	**Credits:** 0
Internal Marks: 40 + 10*	**Percentage of Numerical/Design/Programming Problems:**
External Marks:	**Duration of End Semester Exam (ESE):**
Total Marks: 50	**Elective Status:** Compulsory

* 10 marks will be awarded based upon the performance in debates/seminar/field study related to the contemporary issues of the subject.

Prerequisites: NIL

Additional Material Allowed in ESE: NA

On completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1	Measure environmental variables and interpret results.
2	Evaluate local, regional and global environment topics related to resource use and management.
3	Propose solutions to environmental problems related to resource use and management.
4	Interpret the results of scientific studies of environmental problems.
5	Describe threats to global biodiversity, their implications and potential solutions.

Detailed Contents:

Natural resources: Renewable and non-renewable resources, Natural resources and associated problems, Forest resources- Use and over-exploitation, deforestation, case studies, Timber extraction, mining, dams and their effects on forests and tribal people. **[3 Hours]**

Water resources: Use and over-utilization of surface and ground water, floods, drought, conflicts over water, dam's benefits and problems, Food resources- World food problems, changes caused by agriculture and over gazing, effects of modern agriculture, Fertilizers- pesticides problems, water logging, sanity, case studies, Energy Resources- Growing energy needs, renewable and non-renewable energy sources, use of alternate energy sources, case studies, Land Resources- Land as a resource, land degradation, man induced

landslides, soil erosion and desertification. **[4 Hours]**

Eco systems: Concept of an ecosystem, Structure and function of an ecosystem, Producers, consumers, decomposers, Energy flow in the ecosystem, Ecological succession, Food chains, Food webs and ecological pyramids, Introduction, types, characteristic features, structure and function of the following ecosystems- Forest ecosystem, Grass land ecosystem, Desert ecosystem, Aquatic ecosystem (ponds, streams, lakes, rivers, oceans, estuaries **[4 Hours]**

Biodiversity and its conservation: Introduction- Definition- genetics, species and ecosystem diversity, Biogeographical classification of India, Value of biodiversity- competitive use, productive use, social, ethical, aesthetic and option values, Biodiversity at global, national and local level, India as a mega diversity nation, Hot spots of biodiversity, Threats to biodiversity- habitat loss, poaching of wildlife, man wildlife conflicts, Endangered and endemic species of India, Conservation of biodiversity- in-situ and ex-situ conservation of biodiversity. **[4 Hours]**

Environmental Pollution: Definition, causes, effects and control measures of – Air pollution, water pollution, soil pollution, marine pollution, noise pollution, thermal pollution, nuclear hazards, Solid waste management- Causes, effects and control measures of urban and industrial wastes, Role of an individual in prevention of pollution, case studies. **[4 Hours]**

Social issues and the Environment: From unsustainable to sustainable development, Water conservation, rain water harvesting, water shed management, Resettlement and rehabilitation of people- its problems and concerns, case studies, Environmental Ethics- issues and possible solutions, Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust, case studies, Environmental protection act, Air (prevention and control of pollution) act, Water (prevention and control of pollution) act, Wildlife protection act, Forest conservation act. [**5 Hours]**

Human population and the Environment: Population growth and variation among nations, Population explosion- family welfare program, Environment and human health, Human rights, value education, HIV/AIDS, Women and child welfare. **[4 Hours]**

Reference Books

1. Erach Barucha, “Textbook of Environmental studies”, UGC.
2. D.D. Mishra, “Fundamental concepts in Environmental Studies”, S Chand and Co Ltd.
3. K.C. Agarwal, “Environment Biology”, Nidi Publ. Ltd.
4. Cunnighan, “Principle of Environment Science”, W.P

Subject Code: PCCS-108

Subject Name: Artificial Intelligence

Programme: B.Tech.(CSE)	**L:** 3 **T:** 0 **P:** 0
Semester: 5th	**Teaching Hours:** 38
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 60%
External Marks: 60	**Duration of End Semester Exam (ESE):** 3 hours
Total Marks: 100	**Course Status:** Compulsory

Prerequisites: Knowledge of problem solving using different algorithms and basic programming.

Additional Material Allowed in ESE: [NIL]

On completion of the course, the student will have the ability to:

CO#	Course Outcome
1.	Understand the concept of Artificial Intelligence, Agents, their types and structure.
2.	Apply and analyze search strategies to solve the informed and uninformed problems.
3.	Design and evaluate intelligent expert models for perception and prediction from intelligent environment.
4.	Formulate valid solutions for problems involving uncertain inputs or outcomes by using decision making techniques.
5.	Demonstrate and enrich knowledge to select and apply AI tools to synthesize information and develop models within constraints of application area.
6.	Examine the issues involved in knowledge bases, reasoning systems and planning.

Detailed Contents:

Part A

Introduction: Intelligence, Foundations of artificial intelligence (AI). History of AI, Agents and Environments, Rationality of Agents, Nature and Structure of Agents, Communication among Agents. **[3 Hours]**

Problem Formulation and solution: Problem types, States and operators, State space, Uninformed Search Strategies, Informed Search Strategies– Best first search, A* algorithm, Heuristic functions, Iterative deepening A*(IDA), Small memory A*(SMA). **[5 Hours]**

Game playing: Perfect Information game, Imperfect Information game, Evaluation function, Minimax algorithm, Alpha-beta pruning. **[3 Hours]**

Logical Reasoning: Inference in Propositional logic and First order Predicate logic, Resolution, Logical reasoning, Forward chaining, Backward chaining; Knowledge representation techniques: semantic networks, Frames. **[7 Hours]**

Part B

Planning: Basic representation of plans, Partial order planning, Planning in the blocks world, Hierarchical planning, Conditional planning, Representation of time, schedule and resource constraints, Measures, temporal constraints. **[5 Hours]**

Uncertainty: Basic probability, Bayes rule and its use, Belief networks, Default reasoning, Fuzzy sets and fuzzy logic; Decision making– Utility theory, Utility functions, Decision theoretic expert systems. **[5 Hours]**

Inductive learning: Decision trees, Rule based learning, Current-best-hypothesis search, Least commitment search, Neural networks, Reinforcement learning, Genetic algorithms. **[6 Hours]**

Applications: Areas of AI, Natural language processing, Case study of existing expert systems. **[4 Hours]**

Text Books

1. Stuart Russell and Peter Norvig, "Artificial Intelligence: A Modern Approach", Prentice Hall.
2. Saroj Kaushik, "Artificial Intelligence", Cengage Learning India.

Reference Books

1. Elaine Rich and Kevin Knight, "Artificial Intelligence", Tata McGraw Hill.
2. Trivedi, M.C., "A Classical Approach to Artifical Intelligence", Khanna Publishing House, Delhi.
3. David Poole and Alan Mackworth, "Artificial Intelligence: Foundations for Computational Agents", Cambridge University.

E-Books and online learning material

1. HandBook of Artificial Intelligence Edited by Avron Barr and Edward A. Feigenbaum, Computer Science Department, Stanford University.
   https://stacks.stanford.edu/file/druid:qn160ck3308/qn160ck3308.pdf

Online Courses and Video Lectures

1. https://www.coursera.org/courses?query=artificial%20intelligence
   Accessed on May 20,2020.
2. https://nptel.ac.in/courses/106/105/106105077/ Accessed on May20,2020.
3. https://nptel.ac.in/courses/106/102/106102220/ Accessed on May20,2020.
4. https://www.youtube.com/watch?v=bV4t4r3SGuI Accessed on May 20,2020.
5. https://www.youtube.com/watch?v=iF1tOCEXLXY Accessed on May20,2020.

Subject Code: PCCS-109

Subject Name: Database Management Systems

Programme: B.Tech. (CSE)	**L:** 3 **T:** 0 **P:** 0
Semester: 5	**Teaching Hours:40**
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** Nil
External Marks: 60	**Duration of End Semester Exam (ESE):** 3hrs
Total Marks: 100	**Elective Status:** Compulsory

Additional Material Allowed in ESE: Not Any

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Analyze the Information Systems as socio-technical systems, its need and advantages as compared to traditional file based systems.
2	To study the physical and logical database designs, database modeling, relational, hierarchical, and network models.
3	Analyze Database design using E-R data model by identifying entities, attributes, relationships, generalization and specialization along with relational algebra.
4	To understand and use data manipulation language to query, update, and manage a database.
5	Apply and create Relational Database Design process with Normalization and De-normalization of data.
6	To develop an understanding of essential DBMS concepts such as: database security, integrity, concurrency.

Detailed Contents:

Part-A

Introduction to Database Concepts: Introduction, Database systems versus file systems, Difference between Database and non-database system, Characteristics of Database Approach, Advantages and Disadvantages of Using DBMS. Data base users and administrators, Schemas and Instances, DBMS Architecture, components of a database system, Data Independence, Database Language and Interfaces, Classification of Database Management Systems. Introduction to NoSQL database. **[5 Hours]**

Entity Relationship Model: Data models, Entity types, Entity sets, Attributes and keys, Relationship types, Relationship sets, Roles and structural constraints, Weak entity types, Design choices for ER conceptual design, Comparison of Models. **[4 Hours]**

Relational Model: Relational model concepts, Constraints, Update operations, Transaction and dealing with constraint violations. Relational Algebra –Unary relational operations, Operations from Set theory,

Binary relational operations, DIVISION operation and additional relational operations. Relational Calculus – Tuple relational calculus and Domain relational calculus, Queries related to Relational Algebra and Relational Calculus. **[7 Hours]**

SQL: SQL Data Definition and data types, specifying constraints in SQL, Schema change statements, Basic queries in SQL, Set operations, Aggregate functions and views, Complex queries in SQL, Additional features of SQL. **[7 Hours]**

Part-B

Relational Database Design: Informal design guidelines for Relational Schemas, Functional dependencies, Inference rules for functional dependencies, Equivalence of set of functional dependencies, 2QMinimal cover, Normal forms based on primary keys– (1^{st}NF, 2^{nd}NF, 3^{rd}NF, 4^{th}NF and 5^{th}NF) Decomposition into normalized relations. Physical Database Design – File structures (Sequential files, Indexing, B tree). **[6 Hours]**

Transaction Management and Concurrency Control: Introduction to Transaction Processing, Transaction and System Concepts, need of concurrency control, ACID properties, Schedules, Characterizing schedules based on recoverability and serializability, Two - phase locking techniques for concurrency control. **[4 Hours]**

Database Recovery and Security: Need of recovery, Recovery concepts, Recovery techniques Deferred update, Immediate update, Shadow paging. Database security – Threats to databases, Control measures, Database security and DBA, Discretionary access control based on granting and revoking privileges, Mandatory access control, Introduction to Statistical Database Security, Encryption and decryption.

[7 Hours]

Text Books

1. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, "Database System Concepts", McGraw Hill Education.
2. RamezElmasri, Shamkant B Navathe, "Fundamentals of Database Systems", Pearson Education.
3. Connolly, "Specifications of Database Systems: A Practical Approach to Design, Implementation and Management", Pearson India.
4. Alexis Leon, Mathews Leon, "Database Management Systems" Leon Press.
5. S.K. Singh, "Database Systems Concepts, Design and Applications, Pearson Education.
6. Raghu Ramakrishnan, Johannes Gehrke, "Database Management Systems", Tata McGrawHill.

Reference Books

1. SQL,PL/SQL ,The programming language of oracle, Ivan Bayross BPB Publication
2. An introduction to database system by C.J.Date (Addison Welsey, Publishing house).
3. An introduction to Database Systems by Bipin C. Desai, Galgotia publications.
4. Prateek Bhatia, Database Management system, Kalayani Publishers

E-Books and online learning material

1. Database Management system. 2nd Ed.

https://ff.tu-sofia.bg/~bogi/knigi/BD/Database%20Management%20Systems.%202nd%20Ed.pdf

2. Fundamentals of Database Management Systems eBook.

https://circuitmix.com/free-download-fundamentals-of-database-management-systems-ebook/

Online Courses and Video Lectures

1. https://nptel.ac.in/courses/106/106/106106220/ Accessed on Jan22, 2020
2. https://www.youtube.com/watch?v=5TU7zH0Z8 Accessed on Jan 22, 2020
3. https://www.youtube.com/watch?v=Z2Zx2G02aI4 Accessed on Jan 22, 2020
4. https://www.youtube.com/watch?v=Kmp76uRHz9c Accessed on Jan 22, 2020
5. https://www.youtube.com/watch?v=QYd6ZjHpzBg Accessed on Jan 22, 2020

Subject Code: PCCS-110

Subject Name: Formal Language and Automata Theory

Programme: B.Tech. (CSE)	**L:** 3 **T:** 1 **P:** 0
Semester: 5	**Teaching Hours:** 40
Theory/Practical: Theory	**Credits:** 4
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 35%
External Marks: 60	**Duration of End Semester Exam (ESE):** 3hrs
Total Marks: 100	**Elective Status:** Compulsory

Additional Material Allowed in ESE: Not Any

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Apply the knowledge of mathematics and statistics to solve complex engineering problems related to automata theory.
2	Identify, formulate and analyze uses and Constraints of various computational models used in engineering practice.
3	Make use of research-based knowledge to abstract the models of computing and their powers to recognize the grammars.
4	Design and evaluate abstract machines that demonstrate the properties of physical machines and be able to specify the possible inputs, processes and outputs of these machines.
5	Compare and analyze different computational models including prediction and modeling to complex engineering activities with an understanding of the limitations.
6	Recognize and comprehend formal reasoning about machines and languages to engage in independent and life-long learning in the broadest context of technological change.

Detailed Contents:

Part-A

Finite Automata: Deterministic Finite Automata, Acceptance by Finite Automata, Transition systems, Non-Deterministic Finite Automata, Equivalence of DFA and NDFA, Moore and Mealy machines, Equivalence of Moore and Mealy machine, Minimization of Finite Automata, Applications and limitations of Finite Automata. **[6 Hours]**

Formal Languages: Basics of strings, Alphabets, grammar, Formal language, Chomsky classification of languages, Languages and their relation, Operations on languages, Closure properties of language classes. **[4 Hours]**

Regular Grammar: Regular grammars, Regular expressions, Algebraic method using Arden's theorem, Equivalence of Finite Automata and Regular expressions, Properties of regular languages, Pumping lemma. **[5 Hours]**

Context Free Language: Derivation, Ambiguity, Simplification of context free grammar, normal forms– Chomsky Normal Form, Greibach Normal Form, Pumping lemma. **[5 Hours]**

Part-B

Push Down Automata: Description and definition, Acceptance by Push Down Automata, Equivalence of Push Down Automata and context free grammars and languages. **[6 Hours]**

Turing Machine: Definition and Model, Representation of Turing Machine, Design of Turing Machine, Variants of Turing Machine, Decidability and recursively enumerable languages, Halting problem, Post correspondence problem. **[6 Hours]**

Context Sensitive Language: Context sensitive language, Model of linear bounded automata, Relation between linear bounded automata and context sensitive language. **[5 Hours]**

Text Books

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, "Introduction to Automata Theory, Languages and Computation", Pearson Education.
2. K.L.P. Mishra and N. Chandrasekaran, "Theory of Computer Science", Third Edition, PHI Learning Private Limited.
3. K.V.N. Sunitha, N. Kalyani, "Formal Languages and Automata Theory", McGraw-Hill.

Reference Books

1. Daniel, A.Cohen, "Introduction to Computer Theory", Wiley India Pvt. Ltd.
2. M. Sipser, "Introduction to the Theory of Computation", Second Edition, Cengage Learning.
3. M. A. Harrison, "Introduction to Formal Language Theory", Addison-Wesley
4. Peter Linz, "An Introduction to Formal Languages and Automata", Jones and Bartlett Publishers.

E-Books and online learning material

1. Jean-Eric Pin, "Mathematical Foundations of Automata Theory",Lecture notes LIAFA, Université Paris https://www.irif.fr/~jep//PDF/MPRI/MPRI.pdf
2. Michael Sipser, "Introduction to the Theory of Computation",Thomson Course Technology http://en.bookfi.net/book/1139836

3. Anil MaheshwariandMichielSmid, "Introduction to Theory of Computation", School of Computer Science, Carleton University, Ottawa Canada https://cglab.ca/~michiel/TheoryOfComputation/TheoryOfComputation.pdf
4. Michael Levet, "Theory of Computation", Lecture Notes,University of South Carolina-Columbia http://people.math.sc.edu/mlevet/Lecture_Notes.pdf

Online Courses and Video Lectures

1. https://nptel.ac.in/courses/106/104/106104028/ Accessed on May23, 2020
2. https://nptel.ac.in/courses/106/106/106106049/ Accessed on May23, 2020
3. https://nptel.ac.in/courses/106/104/106104148/ Accessed on May23, 2020

Subject Code: PCCS-111

Subject Name: Design and Analysis of Algorithms

Programme: B.Tech. (CSE)	**L:** 3 **T:** 1 **P:** 0
Semester: 5	**Teaching Hours:** 40
Theory/Practical: Theory	**Credits:** 4
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 90%
External Marks: 60	**Duration of End Semester Exam (ESE):** 3hrs
Total Marks: 100	**Elective Status:** Compulsory

Prerequisites: Knowledge of Data Structures and Algorithms

Additional Material Allowed in ESE: [Scientific Calculator]

On completion of the course, the student will have the ability to:

CO#	COURSE OUTCOMES
1	Understand and learn the fundamental techniques for designing algorithms.
2	Learn various advanced techniques to design algorithms for solving complex problems.
3	Design the algorithms using basic and advanced algorithm design techniques.
4	Identify and design various existing algorithms based on advanced techniques.
5	Analyze the algorithms based on time and space complexity to find optimal algorithm for a given problem.
6	Categorize various problems based on the complexity and properties of algorithms that solves these problems.

Detailed Contents:

Part-A

Introduction: Algorithms, Algorithm Specification, Performance Analysis: Space complexity,Time complexity, Asymptotic Notations- Big-Oh notation (O), Omega notation (Ω), Theta notation (Θ), and Little-oh notation (o), Mathematical analysis of Non-Recursive and recursive Algorithms with Examples. **[4 Hours]**

Divide and Conquer: General method, solving recurrences using recurrence trees, repeated substitution, statement of Master Theorem, applications – Binary search, Merge sort, Quick sort, Strassen's Matrix Multiplication, Finding the maximum and minimum. **[5 Hours]**

Greedy Algorithms: Greedy choice, optimal substructure property, minimum spanning trees-Prims and Kruskals, Dijkstra shortest path using arrays and heaps, fractional knapsack, Travelling salesperson problem and Huffman coding. **[5 Hours]**

Dynamic Programming: Introduction to dynamic programming and application of the algorithm to solve multistage graphs, edit distance, matrix chain multiplication, All pairs shortest path problem and Knapsack problem. **[5 Hours]**

Part-B

Backtracking: General method, N-Queens problem, Sum of subsets problem, Graph coloring, Hamiltonian cycles. **[4 Hours]**

Application of Graph Traversal Techniques: Representation of graphs, BFS (as a method for SSSP on unweighted graphs), DFS, connected components, topological sorting of DAGs, biconnected components, and strongly connected components in directed graphs. **[5 Hours]**

String Matching: Introduction, Brute Force algorithm, Rabin-Karp algorithm, KMP algorithm, Boyer-Moore algorithm. **[5 Hours]**

NP Completeness: classes NP, P, NP-complete, and polynomial time reductions, Introduction to approximation algorithms, Absolute approximations, E-approximations. **[7 Hours]**

Text Books:

1. Ellis Horowitz, Sartaj Sahni and S. Rajasekharan, "Fundamentals of Computer Algorithms, Universities Press.
2. P. H. Dave, H. B. Dave, "Design and Analysis of Algorithms", Pearson Education.

Reference Books:

1. M. T. Goodrich and R. Tomassia, "Algorithm Design: Foundations, Analysis and Internet examples", John Wiley and sons.
2. S. Sridhar, "Design and Analysis of Algorithms", Oxford Univ. Press
3. Aho, Ullman and Hopcroft, "Design and Analysis of algorithms", Pearson Education.
4. R. Neapolitan and K. Naimipour, "Foundations of Algorithms", 4th edition, Jones and Bartlett Student edition.
5. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "Introduction to Algorithms, 3rd Edition", PHI

E-Books and online learning material:

1. Ellis Horowitz, Sartaj Sahni and S. Rajasekharan, "Fundamentals of Computer Algorithms, 2nd Edition", Universities Press.

https://nasirmir.files.wordpress.com/2012/09/fundamentals-of-computer-algorithms-by-ellis horowitz-1984.pdf

Online Courses and Video Lectures:

1. https://nptel.ac.in/content/storage/MP4/new/106106131/mod01lec01.mp4
2. https://nptel.ac.in/content/storage/MP4/new/106106131/mod01lec05.mp4
3. https://nptel.ac.in/content/storage/MP4/new/106106131/mod01lec06.mp4
4. https://nptel.ac.in/content/storage/MP4/new/106106131/mod02lec09.mp4

5. https://nptel.ac.in/content/storage/MP4/new/106106131/mod02lec13.mp4
6. https://nptel.ac.in/content/storage/MP4/new/106106131/mod03lec18.mp4
7. https://nptel.ac.in/content/storage/MP4/new/106106131/mod09lec44.mp4
8. https://nptel.ac.in/content/storage/MP4/new/106106131/mod10lec50.mp4

Subject Code: LPCCS-106

Subject Name: Artificial Intelligence Laboratory

Programme: B.Tech. (CSE)	**L:** 0 **T:** 0 **P:** 2
Semester: 5	**Teaching Hours:** 24
Theory/Practical: Practical	**Credits:** 1
Internal Marks: 30	**Percentage of Numerical/Design/Programming Problems:** 100%
External Marks: 20	**Duration of End Semester Exam (ESE):** 2 hrs
Total Marks: 50	**Elective Status:** Compulsory

Prerequisites:

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Design and implement efficient Uninformed search techniques to solve problems.
2	Apply the knowledge of different informed search to identify and implement the appropriate techniques in problem solving
3	Utilize knowledge and techniques of game playing to develop single player game.
4	Handle uncertainty by designing the Bayesian network and inferring from the given data.
5	Utilize the knowledge and techniques of Artificial Intelligence while working in multidisciplinary teams.
6	Design and develop projects using Artificial Intelligence tools and techniques.

List of Practicals:

1. Introduction to python Interpreter.
2. Programs to implement input output and control flow tools in python.
3. Programs to implement different Data Structures in Python.
4. Introduction to Standard Library, Virtual Environments and packages in Python.
5. Write a program to implement Breadth First search for water jug problem.
6. Write a program to implement Depth First search for water jug problem.
7. Write a Program to implement Best First Search.
8. Write a program to implement A*algorithm.
9. Write a Program to implement tic tac toe game for 0 and X.
10. Write a Program to construct a Bayesian network from given data.
11. Write a Program to infer from the Bayesian network

Project:

Students are required to develop an expert system for real life problems/games, Expert system; implement a production system, medical diagnosis expert system, agriculture expert system, troubleshooting of computer systems, and implementation of neural/fuzzy network.

Subject Code: LPCCS-107
Subject Name: Database Management Systems Laboratory

Programme: B.Tech. (CSE)	**L:** 0 **T:** 0 **P:** 2
Semester: 5	**Teaching Hours:** 24
Theory/Practical: Practical	**Credits:** 1
Internal Marks: 30	**Percentage of Numerical/Design/Programming Problems:** 100%
External Marks: 20	**Duration of End Semester Exam (ESE):** 2 hrs
Total Marks: 50	**Elective Status:** Compulsory

Prerequisites: Fundamentals of Computers.

On Completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Understand, analyze and apply common SQL statements including DDL, DML and DCL statements to perform different operations.
2	Design different views of tables for different users and to apply embedded and nested queries.
3	Design and implement a database for a given problem according to well known design principles that balance data retrieval performance with data consistency.
4	Demonstrate and understand relational algebra in Database which is helpful to design related database software components.
5	Identify the user requirements from a typical business situation, and to document them.
6	Emphasize on team work and developing database applications using modern database tools

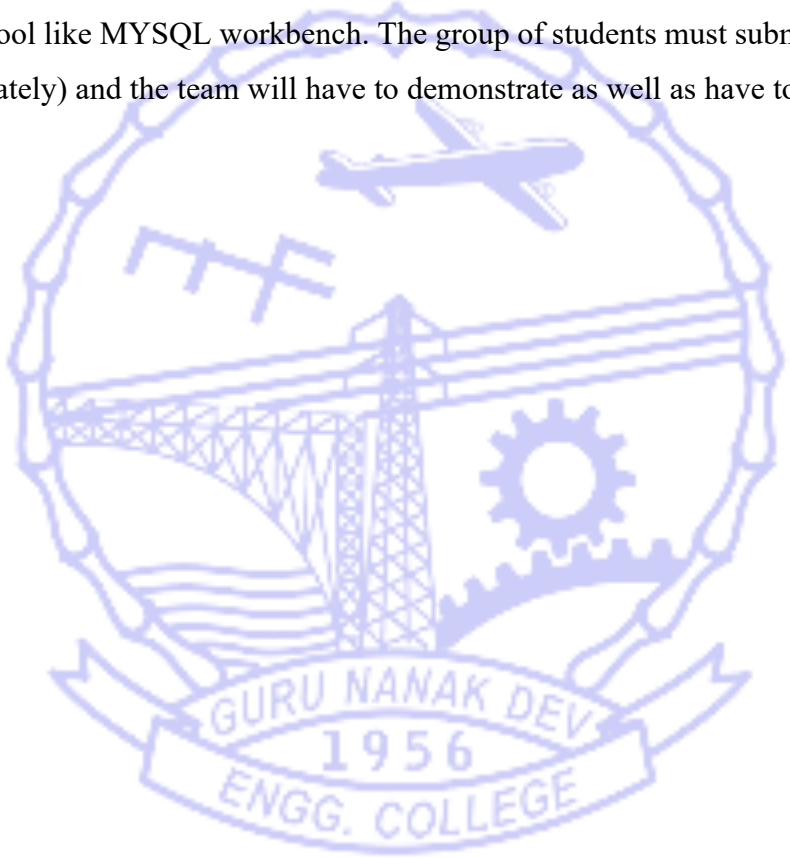
Special Instruction related to resources requirement: MY SQL,SQL Server, Oracle can be used for the queries.

List of Practicals:

1. Write the queries for Data Definition (create, drop, alter and rename) and Data Manipulation Language (select, insert, update and delete).
2. Write SQL queries using logical operators (,=etc).
3. Write SQL queries using SQL operators (between, and, or, in, like, null).
4. Write SQL query using character, number, date and group functions
5. Write SQL queries for Relational Algebra (union, intersect, and minus, etc.)
6. Write SQL queries for extracting data from more than one table (equi-Join, non-equi-join, outer join)
7. Write SQL queries for sub queries, nested queries. 8. Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.

8. Queries (along with sub Queries) using any, all, in, exists, notexists, union, intersect, constraints. Example - Select the roll number and name of the student who secured fourth rank in the class.
9. Queries using aggregate functions (count, sum, avg, max and min), group by, having and creation and dropping of views.
10. Queries using conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next_day, add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date).
11. Write SQL queries to create views and also apply different operations on views.

Minor Project: By using standard database design rules, database has to be designed for a specific assigned problem to a group of two to three students. ER diagram related to project must also be prepared with an open source database tool like MYSQL workbench. The group of students must submit a project report of 8 to 10 pages (approximately) and the team will have to demonstrate as well as have to give a presentation of the same.



Scheme and syllabus of B.Tech. (2018 batch onwards)

Subject Code: LPCCS-108

Subject Name: Design and Analysis of Algorithms Laboratory

Programme: B.Tech. (CSE)	**L:** 0 **T:** 0 **P:** 2
Semester: 5	**Teaching Hours:** 24
Theory/Practical: Practical	**Credits:** 1
Internal Marks: 30	**Percentage of Numerical/Design/Programming Problems:** 100%
External Marks: 20	**Duration of End Semester Exam (ESE):** 2 hrs
Total Marks: 50	**Elective Status:** Compulsory

Prerequisites: Basic Programming

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Construct algorithms using basic design techniques for searching, sorting and graph algorithms.
2	Design algorithm using advanced techniques for solving complex problems.
3	Identify the given problem and formulate and design algorithm for solving given problem.
4	Use modern engineering tools and latest programming language to implement the designed algorithms.
5	Apply knowledge and function on multi-disciplinary teams through mini projects based on various problems.
6	Analyse the performance of various algorithms to choose the optimum algorithm

List of Practicals:

1. Write a program to find out a roll number from college database using binary search algorithm.
2. Write a program to sort the class roll numbers of your class using merge sort algorithm and determine the time required to sort the elements.
3. Write a program to sort the university roll numbers of your class using Quick sort method and determine the time required to sort the elements.
4. Write a program to solve 0/1 knapsack using Greedy algorithm.
5. Write a program to find minimum cost to set the phone lines to connect all the cities of your state using Prim's algorithm.
6. Write a program to find the minimum cost of connecting all the engineering colleges in your state using Kruskal's algorithm.
7. Write a program to find minimum route for a newspaper distributer of your locality using Greedy

algorithm.

8. Write a program to find shortest path from your home to college using Dijkstra's algorithm.
9. Write a program to find shortest path from your home to college using Bellman-Ford algorithm.
10. Write a program to solve 0/1 knapsack using dynamic programming.
11. Write a program to find the shortest path of the multistage graph using dynamic programming.
12. Write a program to find minimum distance between different cities of your state using FloydWarshall algorithm.
13. Write a program to find the solution to the 8 queen's problem using the backtracking.
14. Write a program to solve subset sum problem using Backtracking.
15. Write a program to use a queue to store the node and mark it as 'visited' until all its neighbours (vertices that are directly connected to it) are marked. Implement by using bfs algorithm for a graph.
16. Write a program to implement the dfs algorithm for a graph.
17. Write a program to match the pattern by using Brute Force algorithm, Rabin-Karp algorithm, KMP algorithm and Boyer-Moore algorithm.

Subject Code: PCCS-112

Subject Name: Compiler Design

Programme: B.Tech. (CSE)	**L:** 3 **T:** 1 **P:** 0
Semester: 6	**Teaching Hours:** 38
Theory/Practical: Theory	**Credits:** 4
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 40%
External Marks: 60	**Duration of End Semester Exam (ESE):** 3 hours
Total Marks: 100	**Course Status:** Compulsory

Prerequisites: Knowledge of problem solving using different algorithms and basic programming.

Additional Material Allowed in ESE: [NIL]

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1.	Apply knowledge of system programming and mathematics to solve problems related to language translation.
2.	Identify, formulate and solve engineering problems in the area of language translation and compiler design.
3.	Formulate machine code by considering the system design components and functionalities involved in compilation.
4.	Inspect runtime structure used to represent constructs of programming language during compilation process.
5.	Use of compiler phases to develop an understanding of their use in building tools used for engineering practice.
6.	Developing an awareness of the functionality and complexity of modern compilers to engage in independent and life-long learning in the broadest context of technological change.

Detailed Contents:

Part A

Introduction to Compiler: Language Processors, The Structure of a Compiler, The Grouping of Phases into Passes, Applications of Compiler Technology, Programming Language Basics. [**3 Hours**]

Lexical Analysis: Role of lexical analyzer, Tokens, Patterns, and Lexemes, Attributes for Tokens, Lexical Errors, Input Buffering, Sentinels, Specification of Tokens, Recognition of Tokens, The Lexical-Analyzer Generator Lex, Finite Automata. [**5 Hours**]

Syntax Analysis: Introduction, Role of the parser, Context-Free Grammars (CFG), Writing a Grammar, Writing a Grammar, Top down parsing –Backtracking, LL(1), Recursive descent parsing, Non-recursive

Predictive Parsing. Bottom-up parsing – Shift reduce parsing, LR parsers, SLR parser. Canonical LR parser, LALR parser, Introduction to The Parser Generator Yacc. [**6 Hours**]

Syntax-Directed Translation: Syntax-Directed Definitions, Evaluation Orders for SDD's, Applications of Syntax-Directed Translation, Syntax-Directed Translation Schemes, Implementing L-Attributed SDD's. [**6 Hours**]

Part-B

Intermediate Code Generation: Variants of Syntax Trees, Three-Address Code, Types and Declarations, Translation of Expressions, Type Checking, Control Flow, Backpatching, Switch-Statements, Intermediate Code for Procedures. [**6 Hours**]

Code Generation: Issues in the Design of a Code Generator, The Target Language, Addresses in the Target Code , Basic Blocks and Flow Graphs, Optimization of Basic Blocks, A Simple Code Generator, Peephole Optimization , Register Allocation and Assignment. [**6 Hours**]

Machine-Independent Optimizations: The Principal Sources of Optimization, Introduction to Data-Flow Analysis, Foundations of Data-Flow Analysis, Constant Propagation, Partial-Redundancy Elimination, Loops in Flow Graphs. [**6 Hours**]

Text Books:

1. Alfred V. Aho, Monica S. Lam, Ravi Seth, Jeffrey D. Ullman, “Compilers, Principles, Techniques, & Tools”, Second Edition, Pearson.

Reference Books:

1. Alfred Aho, Ravi Sethi, Jeffrey D Ullman, “Compilers Principles, Techniques and Tools, Pearson Education Asia, 2003.
2. C. Fischer and R. LeBlanc., “Crafting a Compiler”, Benjamin Cummings, 1991.
3. S. Chattopadhyay, “Compiler Design”, PHI, 2011.
4. C. Holub., “Compiler Design in C”, Prentice-Hall Inc., 1993.
5. Appel., “Modern Compiler Implementation in C: Basic Design”, Cambridge Press,2004.

E-Books and online learning material:

1. https://nptel.ac.in/courses/106/104/106104123/
2. http://index-of.es/Varios-2/Compilers.pdf
3. http://hjemmesider.diku.dk/~torbenm/Basics/basics_lulu2.pdf

Online Courses and Video Lectures:

1. https://nptel.ac.in/courses/106/108/106108113/
2. https://nptel.ac.in/courses/106/104/106104072/
3. https://www.youtube.com/playlist?list=PLrjkTql3jnm-wW5XdvumCa1u9LjczipjA
4. https://www.youtube.com/watch?v=h1LSoF_kUzc
5. https://freevideolectures.com/course/3051/compiler-design

Subject Code: PCCS-113

Subject Name: Computer Graphics

Programme: B.Tech. (CSE)	**L:** 3 **T:** 1 **P:** 0
Semester: 6	**Teaching Hours:** 36
Theory/Practical: Theory	**Credits:** 4
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 25%
External Marks: 60	**Duration of End Semester Exam (ESE):** 3hrs
Total Marks: 100	**Elective Status:** Compulsory

Prerequisites: NIL

Additional Material Allowed in ESE: [Scientific Calculator]

On completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1.	Apply the concepts of mathematical foundations and programming to solve diverse problems related to computer graphics.
2.	Compare and contrast various computer graphic algorithms and their suitability to real world problems.
3.	Utilize models for transformation of 2D and 3D objects.
4.	Identify the areas of computer graphics to apply advance algorithmic techniques for changing the formations of geometrical objects.
5.	Apply mathematics and physics in the design and development of graphics applications.
6.	Justify the application of computer graphics concepts in the development of computer games, information visualization, and business applications.

Detailed Contents:

Part A

Introduction: Overview of computer graphics, Computer graphics applications, Different I/O devices with specialized graphics features, Elements of graphics. Graphic systems – Video display devices, Raster scan systems, Random scan systems. Video basics – Video controller, Raster-scan display processor. **[6 Hours]**

2D Primitives: Scan conversion basics, Algorithm for scan converting a point, Scan converting a line – Digital differential analyser algorithm, Bresenham's line algorithm. Scan converting circle – Bresenham's circle drawing algorithm, Midpoint circle drawing algorithm. Scan converting ellipse– Midpoint ellipse algorithm. Filling Techniques – Scan line polygon fill algorithm, Boundary-fill, Flood-fill. Anti-aliasing. **[6 Hours]**

2-D Transformations: Geometric and coordinate transformations. Geometric transformations – Scaling, Rotation, Translation, Reflection, Shear. Matrix representations, Homogeneous coordinates, Composite

transformations. **[6 Hours]**

Part B

2D Viewing and Clipping: The viewing pipeline, Window-to-viewport transformation, Point clipping, Line clipping algorithms – Cohen-Sutherland, Liang-Barsky, Nicholl-Lee-Nicholl. Polygon clipping algorithms –Sutherland-Hodgeman, Weiler-Atherton. Curve and text clipping. **[5 Hours]**

3D Transformations and Viewing: 3D geometric transformations – Scaling, Rotation, Translation, Reflection, Shear. Composite transformations, 3D viewing, Viewing pipeline, Parallel projections, perspective projections, classifications of projections. **[5 Hours]**

Visible-Surface Detection: Classification of visible-surface detection algorithms. Techniques for efficient visible-surface algorithms–Back face detection, Depth-buffer method, A-buffer method, Scan-line method, Depth sorting method, BSP tree Method, Area-subdivision method, Octree Methods, Ray-casting method. **[4 Hours]**

Surface Rendering: Light sources, Surface lighting effects, Illumination models, Polygon rendering methods – Constant-intensity shading, Gouraud shading, Phong shading, Fast Phong shading. **[4 Hours]**

Text Books:

1. D. Hearn and M.P. Baker, "Computer Graphics", Second Edition, PHI/Pearson Education.
2. Zhigang Xiang, Roy Plastock, "Theory and Problems of Computer Graphics", Second Edition, Tata McGraw-Hill.
3. C. Foley, Van Dam, Feiner and Hughes, "Computer Graphics Principles & Practice", Second Edition, Pearson Education.
4. Amarendra N. Sinha, Arun D. Udai, "Computer Graphics", First Edition, Tata McGraw-Hill.
5. N. Krishnamurthy, "Introduction to Computer Graphics", First Edition, Tata McGraw-Hill.

Reference Books

1. Malay K. Pakhira, "Computer Graphics, Multimedia and Animation", Second Edition, PHI.
2. Rogers, Adams, "Mathematics Elements for Computer Graphics", Second Edition, Tata Mc-Graw Hill.

E-Books and online learning material

1. Notes for a Computer Graphics Programming Course by Steve Cunningham
   https://www.cs.csustan.edu/~rsc/NSF/Notes.pdf
2. https://www.tutorialspoint.com/computer_graphics/index.htm
3. https://www.javatpoint.com/computer-graphics-tutorial
4. https://www.geeksforgeeks.org/computer-graphics-2/
5. http://www.svecw.edu.in/Docs%5CCSECGLNotes2013.

Online Courses and Video Lectures

1. https://www.youtube.com/watch?v=fwzYuhduME4 Accessed on Feb 02, 2021
2. https://www.coursera.org/learn/interactive-computer-graphics Accessed on Feb 02, 2021

3. https://www.tutorialspoint.com/computer_graphics Accessed on Feb 02, 2021
4. https://nptel.ac.in/courses/106/106/106106090 Accessed on Feb 02, 2021

Subject Code: PCCS-114

Subject Name: Machine Learning

Programme: B.Tech. (CSE)	**L:**3 **T:**0 **P:**0
Semester: 6	**Teaching Hours:** 36
Theory/Practical: Theory	**Credits:** 03
Internal Marks:40	**Percentage of Numerical/Design/Programming Problems:** 30%
External Marks:60	**Duration of End Semester Exam(ESE):** 3 Hours
Total Marks:100	**Course Status:** Compulsory

Prerequisites: Data Mining Techniques

On Completion of the course the student should be able to:

CO#	COURSE OUTCOMES
1.	Implement probability concepts in learning problems with hypothesis and version spaces.
2.	Illustrate the features and algorithms of machine learning with real world problems.
3.	Characterize the machine learning algorithms as supervised learning and unsupervised learning and apply and analyze the various algorithms of supervised and unsupervised learning.
4.	Analyze the concept of neural networks for learning linear and non-linear activation functions.
5.	Apply the concepts of Bayesian analysis from probability models and methods.
6.	Explain and design genetic algorithms for engineering problems with their analysis using evaluation measures.

Detailed Contents

Part A

Introduction: Well defined learning problems, defining a learning system, perspectives and issues in machine learning, the concept learning task, concept learning as search, Find-S: finding a maximally specific hypothesis, version spaces and the candidate elimination algorithm, Inductive bias, probability theory. **[4 Hours]**

Supervised Learning: Basic methods: Distance based methods, Nearest- Neighbors, Decision Trees, Naive Bayes, and Linear models: Linear regression, Logistic Regression, Generalized Linear Models, Support Vector Machines, Nonlinearity and kernel Methods. **[5 Hours]**

Unsupervised Learning: Clustering: k-means/ kernel k-means, Dimensionality Reduction: PCA and

kernel PCA, Matrix Factorization and Matrix Completion, Generative models (mixture models and latent factor models). **[5 Hours]**

Decision Tree Learning: Introduction, decision tree representation, appropriate problems for decision tree learning, the basic decision tree learning algorithm, hypothesis space search in decision tree learning, Inductive bias in decision tree learning, Ensemble methods- Bagging, Gradient Boosting, Random Forest. **[5 Hours]**

Part B

Artificial Neural Networks: Introduction, Neural network representation, appropriate problems for neural network learning, perceptron, gradient descent and the delta rule, Adaline, Multilayer networks, Derivation of Back propagation rule, back propagation algorithm, Initialization, Training & Validation. **[5 Hours]**

Bayesian Learning: Introduction, Bayes theorem and concept learning, Maximum likelihood and least squared error hypothesis for predicting probabilities, minimum description length principle, Bayes optimal classifier, Naive Bayes classifier, Bayesian belief networks. **[6 Hours]**

Genetic Algorithms: Motivation, Genetic algorithms, an illustrative example, hypothesis space search, genetic programming, models of evolution and learning. **[3 Hours]**

Design and Analysis of Algorithms: Study of factors and responses related with experimentation, Hypothesis testing, performance analysis, Evaluation measures-bootstrapping & cross-validation, ROC curve. **[3 Hours]**

Textbooks:

1. Tom M. Mitchell, Machine Learning, McGraw Hill, First Edition.
2. Ethern Alpaydin, Introduction to Machine Learning, MIT Press, 3rd Edition.
3. Aditya Dwivedi, Machine Learning Textbook, Kindle Edition, Dec 2019.

Reference Books:

1. Chris Bishop, Pattern Recognition and Machine Learning, Springer.
2. Kevin Murphy, Machine Learning: A Probabilistic Perspective, MIT Press, 2nd Edition

E-Books and Online Learning Material

1. Introduction to Machine Learning by Nils J. Nilsson
https://ai.stanford.edu/~nilsson/MLBOOK.pdf
2. Lecture Notes on Machine Learning by Sebastian Raschka
https://sebastianraschka.com/pdf/lecture-notes/stat479fs18/01_ml-overview_notes.pdf
3. https://www.tutorialspoint.com/machine_learning/machine_learning_tutorial.pdf

Online Courses and Video Lectures:

1. https://nptel.ac.in/courses/106106139/ Accessed on February 17, 2021
2. https://nptel.ac.in/courses/106106213/ Accessed on February 17, 2021
3. https://www.coursera.org/lecture/machine-learning/welcome-to-machine-learning-zcAuT

Accessed on February 17, 2021

4. https://www.udacity.com/course/intro-to-machine-learning-with-tensorflow-nanodegree--nd230
   Accessed on February 17, 2021
5. https://www.cs.ox.ac.uk/people/nando.defreitas/machinelearning/
   Accessed on February 17, 2021

Subject Code: PCCS-115

Subject Name: Cyber Security

Programme: B.Tech. CSE	**L:** 3 **T:** 0 **P:**0
Semester: 6	**Teaching Hours:** 36
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 0%
External Marks: 60	**Duration of End Semester Exam(ESE):** 3hrs
Total Marks: 100	**Elective Status:** Compulsory

Prerequisites: NIL
Additional Material Allowed in ESE: NIL
On completion of the course, the student will have the ability to:

CO#	Course Outcomes (CO)
1	Analyze and illustrate the security policies, as well as protocols to implement security features.
2	Analyze the network and system attacks, defences against them.
3	Incorporate the approaches for risk management and needful practices.
4	Classify the principles of web security.
5	Determine computer networks and examine secure software practices.
6	Design key terms and concepts in cyber security, protect intellectual property and decrease cyber-crimes.

Detailed Contents:

Part A

Introduction to Cyber Space: History of cyber space, Cyber Crime, Information Security, Computer Ethics and Security for users, Familiarization with secure web browser and guidelines to choose, Role of Antivirus, Guidelines for Secure password, Two-steps authentication, Introduction to Password Manager, Wi-Fi Security. **[7 Hours]**

Secure Social Media usage and security: Best practices for safer Social Networking, Basic Security for Windows, User Account Password Smartphone Security, Android Security, IOS Security. **[6 Hours]**

E-commerce Security: Familiarization: Online Banking Security, Mobile Banking Security, Security

of Debit and Credit Card, UPI Security. **[6 Hours]**

Part B

Micro ATM, e-wallet and POS Security: Security of Micro ATMs, e-wallet Security Guidelines, Security Guidelines for Point of Sales (POS), Cyber Security Exercise, Cyber Security Incident Handling, Cyber Security Assurance. **[5 Hours]**

Social Engineering, Threat Landscape and Techniques: Social Engineering, Types of Social Engineering, How Cyber Criminal Works, How to prevent for being a victim of Cyber Crime, Cyber Security Threat Landscape, Emerging Cyber Security Threats, Cyber Security Techniques, Firewall.

[6 Hours]

Information Recovery Tools: Recovering from Information Loss, Destroying Sensitive Information, CCleaner for Windows, Various Case Studies. **[6 Hours]**

Text Books

1. William Easttom II, Computer Security Fundamentals, 4th edition, Pearson.
2. Sunit Belapure Nina Godbole, Cyber Security, 1st edition, Wiley.
3. Christopher Hadnagy, Social Engineering, The Science of Human Hacking, 2nd edition, John Wiley & Sons.
4. Thomas A. Johnson, Cyber Security, 1st edition, CNC Press.
5. Sanjib Sinha, Beginning Ethical Hacking, 1st edition, Apress.

Reference Books

1. Nina Godbole, Information Systems Security: Security Management, Metrics, Frameworks and Best Practices, Wile, 1st edition.
2. Jon Erickson, The art of Exploitation, Starch Press, 2nd edition.

E-Books and online learning material:

1. Cyber Attacks and Counter Measures: http://uou.ac.in/progdetail?pid=CEGCS-17Meilir Page-Jones: Fundamentals.
2. Introduction to Cyber Security available at http://uou.ac.in/foundation-course.
3. Fundamentals of Information Security http://uou.ac.in/progdetail?pid=CEGCS-17.
4. Cyber Security Techniques http://uou.ac.in/progdetail?pid=CEGCS-17.
5. https://www.cybersecurity.ox.ac.uk/resources/videos

Online Courses and Video Lectures

1. https://nptel.ac.in/courses/106/106/106106129/ Accessed on February 12, 2021
2. https://www.utep.edu/information-resources/iso/security-awareness/videos/security-awareness-videos.html Accessed on February 12, 2021
3. https://www.utep.edu/technologysupport/ServiceCatalog/SEC_EmailEncryption.html Accessed on February 12, 2021
4. https://nptel.ac.in/courses/106/105/106105031/ Accessed on February 12, 2021

Subject Code: LPCCS-109

Subject Name: Computer Graphics Laboratory

Programme: B.Tech. (CSE)	**L:** 0 **T:** 0 **P:** 2
Semester: 6	**Teaching Hours:** 24
Theory/Practical: Practical	**Credits:** 1
Internal Marks: 30	**Percentage of Numerical/Design/Programming Problems:** 100%
External Marks: 20	**Duration of End Semester Exam (ESE):** 2 hrs
Total Marks: 50	**Elective Status:** Compulsory

Prerequisites: Fundamentals of computers and knowledge of any programming language like C/C++.

On Completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Apply mathematics and logic to develop computer programs for elementary graphic operations.
2	Implement scan conversion problems using a programming language.
3	Outline the concepts of different type of geometric transformation of objects in 2D and 3D.
4	Implement clipping and filling techniques for modifying an object.
5	Gain experience in creating interactive graphics applications using one or more graphics application programming interfaces.
6	Develop scientific and strategic approach to solve complex problems in the domain of computer graphics.

Special Instruction related to resources requirement: Any programming language like C/C++ could be used for the programs.

List of Practicals:

1. Write a program for creating a simple two-dimensional shape of any object using lines, circle, etc.
2. Write a program to Draw a color cube and spin it using transformation matrices.
3. Implement the DDA algorithm for drawing line (programmer is expected to shift the origin to the center of the screen and divide the screen into required quadrants).
4. Write a program to input the line coordinates from the user to generate a line using Bresenham's Algorithm.
5. Write a program to generate a complete moving wheel using Midpoint circle drawing algorithm and DDA line drawing algorithm.

6. Write a program to draw an ellipse using the Midpoint ellipse generation algorithm for both the regions.
7. Write a program to draw any 2-D object and perform the transformations on it according to the input parameters from the user, namely: Translation, Rotation and Scaling.
8. Write a program to rotate a triangle about any one of its end coordinates.
9. Write program to draw a house like figure and perform the following operations.

a) Scaling about the origin followed by translation.

b) Scaling with reference to an arbitrary point.

10. Write a program to draw a 4×4 chessboard rotated 45˚ with the horizontal axis. Use Bresenham's algorithm to draw all the lines. Use seed fill algorithm to fill black squares of the rotated chessboard.
11. Write a program to perform clipping on a line against the clip window using any line clipping algorithm. The output must be twofold showing the before clipping and after clipping images.
12. Write a program to implement the Sutherland-Hodgeman Polygon Clipping algorithm for clipping any polygon.

Reference Material

Manuals available in Lab.

Subject Code: LPCCS-110

Subject Name: Machine Learning Laboratory

Programme: B.Tech. (CSE)	**L:** 0 **T:** 0 **P:** 2
Semester: 6	**Teaching Hours:** 24
Theory/Practical: Practical	**Credits:** 1
Internal Marks: 30	**Percentage of Numerical/Design/Programming Problems:** 100%
External Marks: 20	**Duration of End Semester Exam (ESE):** 2 hrs
Total Marks: 50	**Elective Status:** Compulsory

Prerequisites: Knowledge of Python

On Completion of the laboratory course student should be able to:

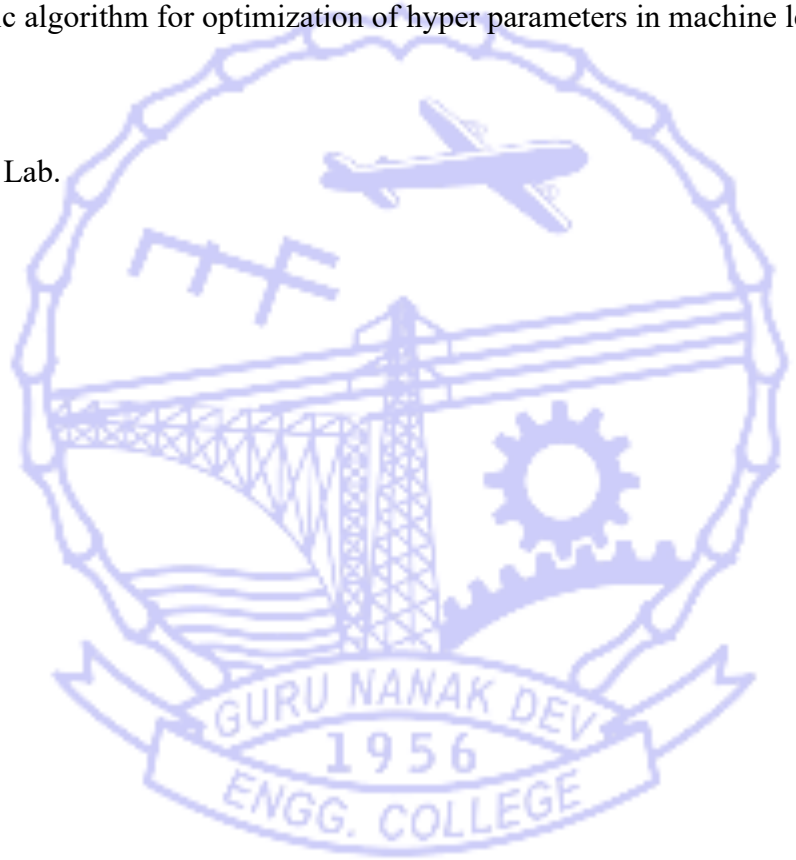
CO#	COURSE OUTCOMES
1.	Develop, analyze and visualize the implementation of machine learning algorithms
2.	Design and develop various algorithms for specific problems with appropriate datasets
3.	Analyze and identify the need for machine learning techniques for specific domain
4.	Develop solutions of real time problems with the prediction and visualization
5.	Apply and analyze Genetic Algorithms for optimization of engineering solutions
6.	Develop and analyze Genetic Algorithms for optimization of engineering solutions

List of Practicals:

1. Write a program to demonstrate **FIND-S algorithm** for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.
2. Write a program for **Candidate Elimination algorithm** for finding the consistent version space based on a given set of training data samples. The training data is read from a .CSV file.
3. Build an Artificial Neural Network by implementing the **Back propagation algorithm** and test the same using appropriate data sets.
4. Write a program to implement the **naïve Bayesian classifier** for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.
5. Assuming a set of documents that need to be classified, use the **naïve Bayesian Classifier** model to perform this task. Calculate the accuracy, precision, and recall for your data set.
6. Write a program to construct a **Bayesian network** considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set.

7. Apply **EM algorithm** to cluster a set of data stored in a .CSV file. Use the same data set for clustering using ***k*-Means algorithm**. Compare the results of these two algorithms and comment on the quality of clustering.
8. Write a program to implement ***k*-Nearest Neighbour algorithm** to classify the iris data set. Print both correct and wrong predictions.
9. Implement the non-parametric **Locally Weighted Regression algorithm** in order to fit data points. Select appropriate data set for your experiment and draw graphs.
10. Write a program to predict high risk patients based on variables (e.g. blood pressure, age etc.) and discriminate them from low risk patients.
11. Develop a genetic algorithm for optimization of hyper parameters in machine learning.

Resource Material:
Manuals available in Lab.



Subject Code: PECS-101

Subject Name: Software Project Management

Programme: B.Tech.(CSE)	**L:** 3 **T:** 0 **P:** 0
Semester: 5th	**Teaching Hours:** 36
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 30%
External Marks: 60	**Duration of End Semester Exam (ESE):** 3 hours
Total Marks: 100	**Course Status:** Elective

Prerequisites:

Additional Material Allowed in ESE: NIL

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1.	Understand and apply the activities involved in the management of software projects.
2.	Analyse the various software development environments and risk management.
3.	Develop and apply the key strategies to monitor, control and quality assurance of software projects.
4.	Select the appropriate planning and estimation models to better evaluate the software projects.
5.	Create a strong working knowledge of ethics and professional responsibility.
6.	Develop effective organisational, leadership and change skills for managing projects, teams and stakeholders.

Detailed Contents:

Part-A

Introduction to Project Management: The characteristics of software projects, Objectives of project management: time, cost and quality, Basics of Project Management, Stakeholders, Stages of Project, The Feasibility Study, Cost-benefit Analysis, Planning, Project Execution, Project and Product Life Cycles, Project Management Knowledge areas, Project Management Tools & Techniques, Project success factors, role of project manager **[5 Hours]**

Project Evaluation and Planning: Activities in Software project management, Project evaluation – Cost benefit analysis, Cash flow forecasting, Cost benefit evaluation techniques, Risk evaluation. Project planning – Stepwise project planning, Software processes and process models. Project costing, COCOMO II, Staffing pattern, Effect of schedule compression, Putnam's equation, Capers Jones

estimating rules of thumb. **[6 Hours]**

Project Scheduling and Risk Management: Project sequencing and scheduling activities, Scheduling resources, Critical path analysis, Network planning, Risk management – Nature and types of risks, Risk planning and control, Risk assessment, Hazard identification, Hazard analysis, PERT and Monte Carlo simulation techniques. **[6 Hours]**

Part-B

Monitoring and Control: Collecting data, Review techniques, Project termination review, Visualizing progress, Cost monitoring, Earned value analysis, Change control, Software Configuration Management (SCM), Managing contracts and acceptance. **[7 Hours]**

People Management: Introduction, Understanding behaviour, Organizational behaviour, Recruitment process, Motivation, The Oldman – Hackman Job Characteristics model, Stress, Health and safety. Working in teams, Decision making, Leadership, Organization and team structures. **[7 Hours]**

Software Quality Management: ISO Standards, Process capability models, Testing and software reliability, Quality plans, Test automation, Overview of project management tools. **[6 Hours]**

Text Books

1. Bob Hughes, Mike Cotterell, “Software Project Management”, Tata McGraw Hill.
2. Royce, “Software Project Management”, Pearson Education.
3. Robert K. Wysocki, “Effective Software Project Management”, Wiley.

Reference Books

1. Ian Sommerville, “Software Engineering”, Pearson Education.
2. R.S. Pressman, “Software Engineering: A Practitioner's Approach”, Tata McGraw Hill.
3. Kassem, “Software Engineering”, Cengage Learning

E-Books and online learning material:

1. Dwayne Phillips, “The Software Project Manager's Handbook: Principles That Work at Work”, Wiley-IEEE Press, URL: https://ieeexplore.ieee.org/servlet/opac?bknumber=5989544

2. Mark Christensen; Richard H. Thayer, “The Project Manager's Guide to Software Engineering's Best Practices, Wiley-IEEE Press”, URL: https://ieeexplore.ieee.org/servlet/opac?bknumber=5989224

3. Walker Royce, “Software Project Management: A Unified Framework”, Addison-Wesley Professional”, URL: https://dl.acm.org/doi/book/10.5555/1942096

Online Courses and Video Lectures

1. https://nptel.ac.in/noc/courses/noc19/SEM2/noc19-cs70/

2. https://nptel.ac.in/courses/106/105/106105182/

Subject Code: PECS-102

Subject Name: Software Testing and Quality Assurance

Programme: B.Tech. CSE	**L:** 3 **T:** 0 **P:** 0
Semester: 6	**Teaching Hours:** 36
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 10%
External Marks: 60	**Duration of End Semester Exam (ESE):** 3 hrs
Total Marks: 100	**Elective Status:** Elective

Additional Material Allowed in ESE: [NIL]

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1.	Test the software by applying testing techniques to deliver a product free from bugs.
2.	Investigate the scenario and to select the proper testing technique.
3.	Explore the test automation concepts and tools and estimation of cost, schedule based on standard metrics.
4.	Test the software by applying testing techniques to deliver a product free from bugs.
5.	Choose appropriate quality assurance models and develop quality.
6.	Ability to conduct formal inspections, record and evaluate results of inspections.

Detailed Contents:

Part A

Introduction: Overview of Software Engineering, Software Process, Process Models, Overview of Project Management Process and its Phases**.** **[3 Hours]**

Testing principles and basic concepts: Testing Concepts: Purpose of Software Testing, Testing Principles, Goals of Testing, Testing aspects: Requirements, Test Scenarios, Test cases, Test scripts/procedures, Strategies for Software Testing, Testing Activities, Mistakes, Faults & Failures, Planning for Verification and Validation, Software Inspections, Automated Static Analysis, Verification and Formal Methods, Levels of Testing White-Box Testing: Test Adequacy Criteria, Static Testing, Structural Testing, Code Complexity Testing, Mutation Testing, Data Flow Testing Black-Box Testing: Test Case Design Criteria, Requirement Based Testing, Positive and Negative Testing, Boundary Value Analysis, Equivalence Partitioning State Based Testing, Domain Testing **[9 Hours]**

Test planning and Execution: Test Plan, Test Management, Test Execution and Reporting, Test Specialist Skills, Tester's Workbench and Tool Categories, Test Maturity Model and Test Process Assessment, Debugging & Root Cause Analysis, Software Items, Component & Units, Test Bed, Traceability and Testability, Attributes of Testable Requirements, Test Matrix, Types of Testing Documentation, Verification Testing, Validation Testing, Integration Testing, System and Acceptance Testing, GUI Testing, Regression Testing, Selection, Minimization and Prioritization of Test Cases for Regression Testing, Creating Test Cases from Requirements and Use cases, Software Defects: Origins of Defects, Defect Classes, Defect Repository / Test Design, Defect Repository. **[7 Hours]**

Part B

Quality Assurance: The software quality challenge, Meaning of software quality, Software quality factors , Software Quality Lessons Learned, The components of the software quality assurance system, Pre-project software quality components: Contract Review, Development and quality plans, SQA components in the project life cycle: Integrating quality activities in the project life cycle, Assuring the quality of software maintenance components, Assuring the quality of external participants' contributions, CASE tools, Software quality infrastructure components, Pareto Principles, Total Quality Management, Ishikawa's Seven Basic Tools **[9 Hours]**

Software Quality Assurance Management: Management components of software quality: Project progress control, Software quality metrics, Costs of software quality, Standards, certification and assessment: Quality management standards, SQA project process standards – IEEE software engineering standards, Management and its role in software quality assurance, The SQA unit and other actors in the SQA system, Inspection as an Up-Front Quality Technique, Software Audit Methods, Software Safety and Its Relation to Software Quality Assurance, SQA for Small Projects, Development Quality Assurance, Quality Management in IT, Introduction to ITIL, Software Quality Assurance Metrics, Software Benchmarks and Baselines. **[8 Hours]**

Text Books

1. Burnstein, "Practical Software Testing", Springer International Edition, ISBN 81-8128- 089-X
2. William E. Perry, " Effective Methods for Software Testing", Third edition, John Wiley and Sons, ISBN 9971-51-345-5
3. Kshirasagar Naik, Priyadarshi Tripathy, "Software Testing and Quality Assurance-Theory and Practice", Second edition, John Wiley & Sons, Inc., 2008, ISBN 978-0-471-78911-6

Reference Books

1. Fenton, Pfleeger, "Software Metrics: A Rigourous and practical Approach", Third edition, Thomson Brooks/Cole, ISBN 981-240-385-X.
2. Desikan, Ramesh, "Software Testing: principles and Practices", sixth edition, Pearson Education, ISBN 81-7758-121-X.
3. Anne Mette, Jonassen Hass, Guide to Advanced Software Testing, Second Edition, ARTECH HOUSE, INC., 2008, ISBN-13: 978-1-59693-285-2

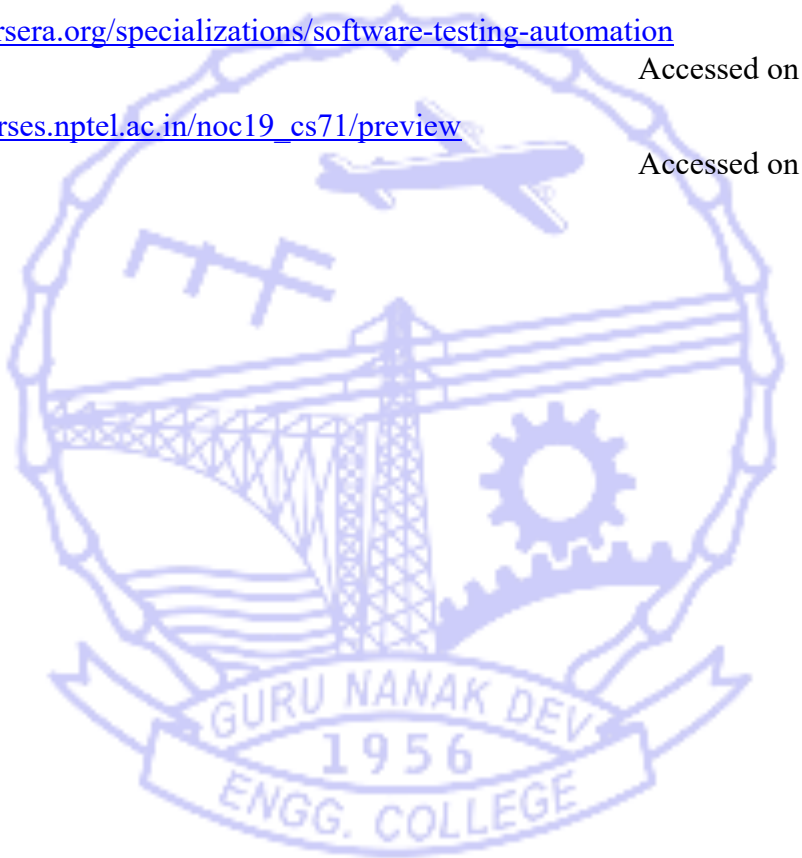
4. Ian Molyneaux, The Art of Application Performance Testing, First edition, O'Reilly Media, Inc., 2009, ISBN: 978-0-596-52066-3
5. Jamie L. Mitchell, Rex Black, Advanced Software Testing—Vol. 3, 2nd Edition, Rocky Nook, 2015, ISBN: 978-1-937538-64-4
6. G. Gordon Schulmeyer, Handbook of Software Quality Assurance, Fourth Edition, ARTECH HOUSE, INC., 2008, ISBN-13: 978-1-59693-186-2

E-Books and online learning material

1. Software Testing and Quality Assurance, Theory and Practice by Kshirasagar Naik and Priyadarshi Tripathy. https://www.softwaretestinggenius.com/download/staqtpsn.pdf

Online Courses and Video Lectures

1. https://www.coursera.org/learn/introduction-software-testing
Accessed on December 26, 2020
2. https://www.coursera.org/specializations/software-testing-automation
Accessed on December 26, 2020.
3. https://onlinecourses.nptel.ac.in/noc19_cs71/preview
Accessed on December 26, 2020.



Scheme and syllabus of B.Tech. (2018 batch onwards)

Subject Code: LPECS-101

Subject Name: Software Testing and Quality Assurance Laboratory

Programme: B.Tech. CSE	**L:** 0 **T:** 0 **P:** 2
Semester: 6	**Teaching Hours:** 24
Theory/Practical: Practical	**Credits:** 1
Internal Marks: 30	**Percentage of Numerical/Design/Programming Problems:** 100%
External Marks: 20	**Duration of End Semester Exam (ESE):** 2 hrs
Total Marks: 50	**Elective Status:** Elective

Prerequisites: NIL

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Ability to conduct formal inspections, record and evaluate results of inspections.
2	Adapt to various test processes, types of errors and fault models and methods of test generation from requirements for continuous quality improvement of the software system along with Software Quality best practices usage.
3	Apply software testing cycle in relation to software development and project management focusing incidents.
4	Apply risks management within a project towards efficient delivery of software solutions.
5	Implement improvements in the software development processes by making use of standards and baselines.
6	Test the software by applying testing techniques to deliver a product free from bugs.

List of Practicals:

1. To Prepare Test Plan for the implemented system under test. The Test Plan shall be based on System Requirement Specification. The Test plan consists of following issues.
    a. Purpose of the test. /Location and schedule of the test.
    b. Test descriptions. /Pass and Fail Criteria.
2. To identify and narrate Test cases, Test scripts/procedures and Test incident report identifier for the system under test. Refer Use case analysis document to prepare mentioned/ identified test documents.
3. To perform Unit testing especially indicating the traced Independent data paths, control paths and Error handling paths. Prepare control flow graphs for the unit under test. Compute the cyclomatic complexity of the unit.
4. To perform Data Flow testing for the Program Segments by identifying the Definition-Use chain and

type of data flow anomaly.

5. To perform Mutation Analysis of the Program Segments along with mutant history, mutation score and type of mutation by using any Code analysis Tool / Mutation Testing Tool.
6. To perform Black-Box Testing for all the units contained in the architectural segments using Equivalence Partitioning, Boundary Value Analysis and Orthogonal Array testing methods. Study exploratory Testing for the Module under Test and merits/demerits of this technique.
7. To perform Regression Testing of the System under construction with Unit and Integration profiles by using any Functional Testing Tool.

Reference Material

Manuals available in Lab.

Subject Code: PECS-107

Subject Name: Advanced Computer Networks

Programme: B.Tech.(CSE)	**L:** 3 **T:** 0 **P:** 0
Semester: 5	**Teaching Hours:** 36 Hours
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 30%
External Marks: 60	**Duration of End Semester Exam (ESE):** 3hours
Total Marks: 100	**Course Status:** Elective

Prerequisites: Basics of computer networks

Additional Material Allowed in ESE: scientific calculator

On completion of the course, the student will have the ability to:

CO #	Course Outcomes
1.	Understand the core ideas of networks thoroughly with network architecture and performance metrics for network designing.
2.	Apply the knowledge of various modes of communication to solve problems of data communication over different medium using various technologies.
3.	Understand and utilize various communication protocols that provide reliable, ordered, and error-checked delivery of a stream of octets.
4.	Design and implement various algorithms of network to ease the communication problems over different geographical areas.
5.	Compare different routing protocols and propose the optimal solution concerning different structures of networks.
6.	Design and implementation of routing and transport layer protocols for advanced multi hop networks for smooth flow of data over different networks.

Detailed Contents:

Part-A

Introduction: Basics, History of Internet, Requirements: perspectives, scalable connectivity, cost effective resource sharing, support for common services, manageability, network architecture: layering and protocols, Internet architecture, network performance: bandwidth, latency, high-speed networks, application performance needs. **[4 Hours]**

Internetworking: half and full duplex**,** Ethernet at physical layer: standard Ethernet, fast Ethernet,

gigabit Ethernet, Ethernet cabling-straight-through, crossover and rolled cable, Data encapsulation. Ethernet at data link layer: CSMA, CSMA/CD and CSMA/CA. **[4 Hours]**

Wireless LANs: Introduction: architecture comparison, characteristics, access control. IEEE 802.11: architecture, MAC Sublayer, Physical layer. Bluetooth: architecture and its layers. **[3 Hours]**

Switching: Switching andbridging: datagrams, virtual circuit switching, source routing, Switches: Basics, its function, types of switches, Spanning Tree Protocol (STP), Virtual LANs (VLANs): purpose, memberships, configuration, connection between switches, advantages, types of VLANs: static and dynamic. **[5 Hours]**

Part-B

TCP Protocols: Internet Protocol (IP): service model, global addresses, datagram forwarding in IP, subnetting and classless addressing, Address Translation (ARP), Host Configuration (DHCP), Error reporting (ICMP). ` **[5 Hours]**

Routing: Network as a graph, Distance Vector (RIP), Link state (OSPF), metrics. Inter-domain routing: routing policies, routing protocols (BGP), Intra-domain routing: routing policies, routing protocols (DVMRP). **[5 Hours]**

Transport Service and Protocols: User Datagram Protocol (UDP): header format, services, and applications, Transmission Control Protocol (TCP): transport service characteristics; transport protocol: features, segment, TCP connection. **[3 Hours]**

Wireless Ad hoc Networks: Mobile Ad hoc Networks (MANETs): features, advantages, routing in MANETs, applications of MANETs, Recent trends in networks: green networking, social networks, software data networks and vehicular ad hoc networks (VANETs). **[3 Hours]**

Text Books:

1. L. L. Peterson, B. S. Davie, Computer Networks. A Systems Approach, Morgan Kaufmann Publishers Inc
2. B.A. Forouzan, Data Communications and Networking, Mc-Graw Hill Education
3. Andrew S. Tanenbaum, “Computer Networks”, Pearson Education

Reference Books:

1. J. F. Kurose, K. W. Ross, Computer Networking. A Top-Down Approach, Addison Wesley Longman
2. G. Antoniou, F. van Harmelen, Semantic Web Primer, The MIT Press
3. J. Day, Pattterns in Network Architecture. A Return to Fundamentals, Prentice Hall
4. Douglas E. Comer, “Internetworking with TCP/IP”, Volume-I, Pearson Education.
5. W. Stallings, “Data and Computer Communication”, Prentice Hall of India.

E-Books and online learning material

- An Introduction to Computer Networks by Peter L Dordal, Department of Computer Science, Loyola University Chicago.

http://intronetworks.cs.luc.edu/current/ComputerNetworks.pdfAccessed on May 21, 2020

Online Courses and video lectures

https://swayam.gov.in/nd1_noc20_cs23/preview Accessed on May 21, 2020

https://nptel.ac.in/courses/106105081/ Accessed on May 21, 2020

https://nptel.ac.in/courses/106105081/2 Accessed on May 21, 2020

https://nptel.ac.in/courses/106105081/5 Accessed on May 21, 2020

https://nptel.ac.in/courses/106105081/16 Accessed on May 21, 2020

https://nptel.ac.in/courses/106105081/31 Accessed on May 21, 2020

Subject Code: PECS-108

Subject Name: Network Security and Cryptography

Programme: B.Tech. (CSE)	**L:** 3 **T:** 0 **P:** 0
Semester: 6	**Teaching Hours:** 36
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 40%
External Marks: 60	**Duration of End Semester Exam(ESE):** 3 hours
Total Marks: 100	**Course Status:** Elective

Prerequisites: Computer Networks

Additional Material Allowed in ESE: Scientific Calculator

On completion of the course the student will have the ability to:

CO #	Course Outcomes
1.	Apply the knowledge of existing authentication protocols and key management techniques to provide security solutions.
2.	Identify and analyze network security attacks and counter measures to prevent those attacks.
3.	Evaluate network security models using available solutions such as PGP, SSL, IPSec to provide robust framework for security threats.
4.	Assess impact of system and web security threats to ensure secure transmission of data.
5.	Analyze the security requirements and solutions for maintaining Data integrity using modern techniques for data transmission.
6.	Testing and verification of cryptography aspects by integrating people, processes and technologies.

Detailed Contents:

Part-A

Introduction to Security: Essentials of network security, Architecture, Security goals, cryptographic attacks: cryptanalytic, non-cryptanalytic attacks, active attack and passive attack, security Services and security mechanism, Fundamental Security design principles, Network security model, standards. **[5 Hours]**

Number Theory: Integer Arithmetic, Euclidean Algorithm, Extended Euclidean Algorithm, Modular Arithmetic, Matrices, Linear Congruence, Prime numbers, Fermat's and Eular's Theorem, Factorization, Chinese Remainder Theorem. **[6 Hours]**

Classical Encryption Techniques: Encryption, Decryption, Plaintext, Cipher text, Key range and Size, Symmetric cipher model, Substitution techniques: Mono-alphabetic ciphers (additive, Caesar,

Multiplicative, affine), polyalphabetic cipher (autokey, playfair, Hill Cipher) Transposition techniques (keyless, keyed, combined approaches) **[6 Hours]**

Part-B

Modern Symmetric-key Ciphers: Modern Block cipher, components of block cipher, two classes of product cipher, Feistal structure, Data Encryption Standard (DES). Modern stream ciphers, Advanced Encryption Standard (AES), Stream ciphers – RC4. **[5 Hours]**

Public Key Cryptography and RSA: Symmetric – Key vs Asymmetric-key cryptosystems, Principles of public key cryptosystems, RSA algorithm and its attacks, Diffie Hellman Key Exchange. **[4 Hours]**

Data Integrity and Authentication: Message: Hash function (SHA-I), Message Authentication (MD5), Digital Signature: services, attacks on digital signature, RSA Digital signature scheme. **[4 Hours]**

Internet Security Protocols: General structure of Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET), Email Security: Pretty Good Privacy (PGP), IP Security – Overview, IP security architecture modes, security protocols: Authentication header(AH) and Encapsulation security payload (ESP). **[6 Hours]**

Text Books:

1. William Stallings, "Cryptography and Network Security Principles and Practice", Pearson Education, 6th Edition.
2. Behrouz A. Forouzan, "Cryptography & Network Security", McGraw-Hill Education, 3rd Edition.
3. Atul Kahate, "Cryptography & Network Security", Tata Mc Graw Hill, 3rd Edition.

Reference Books:

1. Wenbo Mao, "Modern Cryptography: Theory and Practice", Hewlett-Packard Company.
2. William Stallings, "Network Security Essentials, Applications and Standards ", Pearson Education.
3. Alfred J. Menezes, Paul C. van Oorschot and Scott A. Vanstone, "Handbook of Applied Cryptography", CRC Press.
4. Trappe & Washington, "Introduction to Cryptography with Coding Theory", Prentice-Hall.

E-Books and online learning material

1. Modern Cryptography by P. Rogaway
   https://web.cs.ucdavis.edu/~rogaway/classes/227/winter00/
2. A Graduate Course in Applied Cryptography by Dan Boneh
3. Lecture Notes on Cryptography by S. Goldwasser and M. Bellare
   http://cseweb.ucsd.edu/~mihir/papers/gb.pdf

Online Courses and Video Lectures

1. https://crypto.stanford.edu/~dabo/courses/OnlineCrypto/ Accessed on Feb. 16, 2021
2. https://nptel.ac.in/courses/106/105/106105031/ Accessed on Feb. 16, 2021
3. https://nptel.ac.in/courses/106/105/106105162/ Accessed on Feb. 16, 2021
4. https://www.slideshare.net/ayyakathir/cryptography-and-network-security-52030354 Accessed on Feb. 16, 2021
5. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-046j-design-and-analysis-of-algorithms-spring-2015/lecture-videos/lecture-21-cryptography-hash-functions/ Accessed on Feb. 16, 2021
6. https://freevideolectures.com/course/3027/cryptography-and-network-security Accessed on Feb. 16, 2021
7. https://saweis.net/crypto.html Accessed on Feb. 16, 2021

Subject Code: LPECS-104

Subject Name: Network Security and Cryptography Laboratory

Programme: B.Tech.(CSE)	**L:** 0 **T:** 0 **P:** 2
Semester: 6	**Teaching Hours:** 24
Theory/Practical: Practical	**Credits:** 1
Internal Marks: 30	**Percentage of Numerical/Design/Programming Problems:**100%
External Marks: 20	**Duration of End Semester Exam(ESE):** 2 hours
Total Marks: 50	**Course Status:** Elective

Prerequisites: Computer Networks

On completion of the course the student will have the ability to:

CO #	Course Outcomes
1.	Implement encryption and decryption techniques for providing security solutions.
2.	Analyze the impact of public key cryptosystems for secure exchange of information.
3.	Analyze and design Network Security protocols for information exchange over unsecure network.
4.	Apply security principles for implementing authentication applications.
5.	Analyze the security requirements and solutions for maintaining Data integrity using modern techniques for data transmission.
6.	Testing and verification of cryptography aspects by integrating people, processes and technologies.

Special Instruction related to resources requirement: Any programming language like C, C++, can be used for the programs.

List of Practicals:

1. Implement the following Symmetric key cipher techniques :
   a. Caesar Cipher
   b. Multiplicative Cipher
   c. Affine Cipher
   d. Playfair Cipher
   e. Hill Cipher
   f. Rail fence – Row & Column Transformation etc.
2. Implement Diffie-Hellman Key exchange algorithm.
3. Implement RSA Public Key algorithm.
4. Implement Stream cipher algorithm – RC4.
5. Mini Project related to cryptography and network security with the team of 2-4 members

Reference Material

Manuals available in Lab.

Scheme and syllabus of B.Tech. (2018 batch onwards)

Subject Code: PECS-111

Subject Name: Statistics for Data Science

Programme: B.Tech.(CSE)	**L:** 3 **T:** 0 **P:** 0
Semester: 5	**Teaching Hours:** 36 Hours
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 60%
External Marks: 60	**Duration of End Semester Exam(ESE):** 3 hours
Total Marks: 100	**Course Status:** Elective

Additional Material Allowed in ESE: [NIL]

On completion of the course the student will have the ability to:

CO #	Course Outcomes
1.	Able to understand the basic knowledge on fundamental probability concepts, probability of an event, additive rules and conditional probability, Bayes' Theorem, Combinatorial Analysis, Permutations ,Combinations, Binomial Coefficients.
2.	To understand the concept of random variables, properties of common types of random variables, how to identify them and apply them to solve probabilistic problems.
3.	To apply the knowledge of various sampling distributions to compute confidence intervals for the population parameters.
4.	To solve different types of Statistics related problems with well-defined solutions, and tackle open-ended problems that belong to the disciplinary-area boundaries;
5.	To understand basic components of hypothesis testing and perform hypothesis tests on population means, variances and proportions.
6.	To perform Statistical analysis in several circumstances and interprets the results in an applied context.

Detailed Contents:

PART A

Random Variables and Probability Distributions: Random Variables, Discrete Probability Distributions, Distribution Functions for Random Variables, Distribution Functions for Discrete Random Variables, Continuous Random Variables, Joint Distributions, Independent Random Variables, Change of Variables,

Probability Distributions of Functions of Random Variables, Convolutions, Conditional Distributions, Applications to Geometric Probability. **[8 Hours]**

Special ProbabilityDistributions- Binomial Distribution, Normal Distribution, Poisson Distribution, The Central Limit Theorem, Multinomial Distribution, Hyper geometric Distribution, Uniform Distribution, Cauchy Distribution, Gamma Distribution, Beta Distribution, The Chi-Square Distribution, Student's t Distribution, F Distribution, Relationships Among Chi-Square, t, and F Distributions, Bivariate Normal Distribution. **[10 Hours]**

Introduction to Statistics: Population and sample, parameters and statistics, Simple descriptive statistics - Mean, Median, Quantiles, percentiles, and quartiles, Variance and standard deviation, Standard errors of estimates, Interquartile range. **[5 Hours]**

PART B

Correlation: Definition of Correlation, Types of Correlation, Scatter Diagram Method, Karl Person's Correlation Coefficients, Correlation Coefficients for Bivariate frequency distribution, Probable error for Correlation Coefficients, Rank Correlation Co-efficient. **[3 Hours]**

Regression: Definition of Regression, Regression lines, Regression Coefficients, Properties of regression Coefficients, and Fitting of regression lines and estimation for Bivariate frequency distribution, Multiple Linear Regression. **[4 Hours]**

Testing of hypothesis: Meaning, Basic concepts, Flow diagram, Power of a hypothesis test, Important parametric tests,Types of hypothesis (null and alternate), Limitations of tests of hypothesis. **[3 Hours]**

Statistical analysis: Parametric tests, Non-parametric tests, Students t-test, chi square test, analysis of variance (ANOVA). **[3 Hours]**

Text Books

1. C.R Kothari, "Research Methodology: Methods and Techniques", New age international.
2. S.P. Gupta, "Statistical Methods", S. Chand & company.

Reference Books

1. S.C. Gupta, V.K. Kapoor, "Fundamental of Applied Statistic", Sultan Chand Publication.
2. H.R. Vyas, "Business Statistics", B.S. Shah Prakashan.
3. Michael Baron, "Probability and Statistics for Computer Scientists", Chapman and Hall/CRC.
4. John Schiller, R. Alu Srinivasan, Murray Spiegel, "Probability and Statistics", McGraw-Hill Education.

E-Books and online learning material

1. Michael Baron, "Probability and Statistics for Computer Scientists", https://www.academia.edu/35869356/Probability_and_Statistics_for_Computer_Scientists.
2. John Schiller, R. Alu Srinivasan, Murray Spiegel, "Probability and Statistics", McGraw-Hill Education, https://www.academia.edu/35869356/Probability_and_Statistics_for_Computer_ Scientists.

Online Courses and Video Lectures

1. https://www.youtube.com/watch?v=VudrNXCYJt4
2. https://nptel.ac.in/content/storage2/MP4/111106112/mod03lec12.mp4
3. https://nptel.ac.in/content/storage2/MP4/111106112/mod04lec18.mp4
4. https://nptel.ac.in/content/storage2/MP4/111106112/mod04lec19.mp4
5. https://nptel.ac.in/content/storage2/MP4/111106112/mod04lec20.mp4
6. https://youtu.be/0AE_oSOXSC4
7. https://youtu.be/vGn6boqvmpw

Scheme and syllabus of B.Tech. (2018 batch onwards)

Subject Code: PECS-114

Subject Name: Advanced Database Management Systems

Programme: B.Tech.(CSE)	**L:** 3 **T:** 0 **P:** 0
Semester: 6	**Teaching Hours:** 36
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 20%
External Marks: 60	**Duration of End Semester Exam (ESE):** 3hrs
Total Marks: 100	**Elective Status:** Elective

Prerequisites: Basic Knowledge of Computer Fundamentals and Database Management Systems.
Additional Material Allowed in ESE: Nil
On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Implement PL/SQL programming using concept of Cursor Management, Error Handling, Package and Triggers. (Change from Level-2 to Level 3 or above)
2	Apply and Relate the concept of transaction, concurrency control and recovery in database.
3	Recognize the purpose of query processing and optimization and also demonstrate the basic of query evaluation.
4	Illustrate the concept of object oriented database and have experience with object oriented modeling, design and implementation. (Change from Level-2 to Level 3 or above)
5	List the principles of distributed systems and describe the problems and challenges associated with these principles.
6	Evaluate the association rules for mining the data.

Detailed Contents:

Part - A

Introduction to SQL Programming Techniques: Database Programming: Issues and Techniques, Embedded SQL, Dynamic SQL, Database Programming: Data Types, Variables, Constraints, Operators, Conditions, Loops, Strings, Arrays, Procedures, Functions, Cursors, Records, Exceptions, Triggers, Packages, Database Stored Procedures. **[5 Hours]**

Transaction Processing and Optimization: Transaction Processing Concepts, Concurrency Control Techniques, Timestamp ordering, Multiversion Concurrency Control Techniques, Validation (Optimistic) Concurrency Control Techniques, Granularity of Data Items and Multiple Granularity Locking. **[4 Hours]**

Query Processing and Optimization: Query Processing, Syntax Analyzer, Query decomposition, Query Optimization, Heuristic Query Optimization, Algorithms for SELECT and JOIN Operations,

Algorithms for PROJECT and Set Operations, Implementing Aggregate Operations and OUTER JOINS. Using Selectivity and Cost Estimation in Query Optimization. Semantic Query Optimization. **[5 Hours]**

Object-Oriented DBMS: Introduction Advanced Database Applications, Weakness of RDBMS, Storing Objects in Relational Database. Next- Generation Database Systems, OODBMS Perspectives, Persistence, Issues in OODBMS, Advantages and Disadvantages of OODBMS, Object- Oriented Database Design, Comparison of ORDBMS and OODBMS. **[4 Hours]**

Part - B

Distributed Databases and Client-Server Architectures: Distributed Database Concepts, Data Fragmentation, Replication, and Allocation techniques for Distributed Database Design. Types of Distributed Database Systems, Query Processing in Distributed Databases, Overview of Concurrency Control and Recovery in Distributed Databases. **[6 Hours]**

Overview of Data Warehousing and OLAP: Introduction, Characteristics of Data Warehouses, Data Modeling for Data Warehouses, Building a Data Warehouse, Typical Functionality of a Data Warehouse, Data Warehouse versus Views, Problems and Open Issues in Data Warehouses. **[5 Hours]**

Data Mining Concepts: Overview of Data Mining Technology, Association rules, Classification, Clustering, Approaches to Other Data Mining Problems, Application of Data Mining, Commercial Data Mining Tools. **[4 Hours]**

Emerging Database Technologies and Applications: Mobile Databases, Multimedia Databases, Geographical Information Systems (GIS), Genome Data Management. **[3 Hours]**

Text Books:

1. SQL,PL/SQL ,The programming language of oracle, Ivan Bayross, BPB Publication
2. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, "Database System Concepts", McGraw Hill Education.
3. Connolly, "Specifications of Database Systems: A Practical Approach to Design, Implementation and Management", Pearson India.
4. Alexis Leon, Mathews Leon, "Database Management Systems", Leon Press.
5. Raghu Ramakrishnan, Johannes Gehrke, "Database Management Systems", Tata McGraw.

Reference Books

1. SQL,PL/SQL ,The programming language of oracle, Ivan Bayross BPB Publication
2. An introduction to database system by C.J.Date (Addison Welsey, Publishing house).
3. An introduction to Database Systems by Bipin C. Desai, Galgotia publications.
4. Prateek Bhatia, Database Management system, Kalayani Publishers
5. S.K. Singh, "Database Systems Concepts, Design and Applications", Pearson Education.

E-Books and online learning material

1. Database Management system. 2nd Ed. http://fdjpkc.fudan.edu.cn/_upload/article/files/38/18/68cfc4494aa8a05490d9c94b84e8/37986de0-6e42-4a65-ad14-91d6d49e20cf.pdf
2. Fundamentals of Database Management Systems eBook. https://www.circuitmix.com/free-download-fundamentals-of-database-management-systems-ebook/

Online Courses and Video Lectures

1. https://nptel.ac.in/courses/106/106/106106093/ Accessed on Feb. 10, 2021
2. https://www.youtube.com/watch?v=075XblZxQts&list=PLV8vIYTIdSnadoY3-LdIJ8pzxgpdBVbHI Accessed on Feb. 10, 2021
3. https://www.youtube.com/watch?v=SdW5RKUboKc&list=PLSE8ODhjZXjasmrEd2_Yi1deeE360zv5O Accessed on Feb. 10, 2021
4. https://www.youtube.com/watch?v=hKljaVcCMgg&list=PLLANTs44t4TVFZ6i8fIu0wOBv3FVUMc89 Accessed on Feb. 10, 2021
5. https://www.youtube.com/watch?v=5qTtecS682A&list=PLir19lgiavA2XaNFoBIYdLtZWXfneVYW6 Accessed on Feb. 10, 2021

Subject Code: LPECS-107

Subject Name: Advanced Database Management Systems Laboratory

Programme: B.Tech.(CSE)	**L:** 0 **T: 0 P:** 2
Semester: 6	**Teaching Hours:** 24
Theory/Practical: Practical	**Credits:** 1
Internal Marks: 30	**Percentage of Numerical/Design/Programming Problems:** 100%
External Marks: 20	**Duration of End Semester Exam(ESE):** 2 hrs
Total Marks: 50	**Elective Status:** Elective

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Implement PL/SQL programming using concept of Cursor Management, Error Handling, Package and Triggers. (Change from Level-2 to Level 3 or above)
2	Apply and Relate the concept of transaction, concurrency control and recovery in database.
3	Recognize the purpose of query processing and optimization and also demonstrate the basic of query evaluation.
4	Illustrate the concept of object oriented database and have experience with object oriented modeling, design and implementation. (Change from Level-2 to Level 3 or above)
5	List the principles of distributed systems and describe the problems and challenges associated with these principles.
6	Evaluate the association rules for mining the data.

List of Practicals:

1. Insert data to a table using character type variable
    a. Which will get the salary of an employee with particular id from emp table and display it on the screen.
    b. Which creates two variables in the outer block and assign their product to the third variable created in the inner block.
2. Write a PL/SQL procedure to calculate the incentive on a target achieved and display the message either the record updated or not.
3. Write PL/SQL code to count the number of employees in a particular department and check whether this department have any vacancies or not by using functions. Assume there are 45 vacancies in this department.

4. Write a PL/SQL procedure to accepts a BOOLEAN parameter and uses a CASE statement to print Unknown if the value of the parameter is NULL, Yes if it is TRUE, and No if it is FALSE.
    a. which use the relational operators to compare character values for equality or inequality.
5. Write a program in PL/SQL to update the salary of a specifc employee by 8% if the salary exceeds the mid range of the salary against this job and update up to mid range if the salary is less than the mid range of the salary, and display a suitable message.
6. Write a program in PL/SQL to print 1st n numbers with a difference of 3 and starting from 1.
    a. which uses FOR loop to insert ten rows into a database table.
    b. to demonstrate the use of 'WHILE loop'
    c. to demonstrate the use of 'Nested loop'
    d. to demonstrate the use of ' Labeling loop'
    e. to demonstrate the use of 'GOTO statement'
7. Write a program in PL/SQL to insert records from one table to another.
    a. Which uses a cursor to select the five highest paid employees from the emp table.
    b. that uses implicit cursor attributes to update the salary of employees in emp table.
    c. to illustrate the use of different types of Explicit cursors
    d. to illustrate the use of Triggers.
    e. to illustrate the use of Packages
    a. to handle exceptions
8. **Minor Project:** By using standard database design rules, a small database has to be designed for a specific assigned problem by a group of two to three students. Design meaningful Pl/SQL queries related to your project and execute them. Each must submit a project report of 8 to 10 pages (approximately) and the team will have to demonstrate as well as to give a presentation of the same.

Reference Material

Manuals available in Lab.

Subject Code: PECS-119

Subject Name: Information Retrieval

Programme: B.Tech.(CSE)	**L:** 3 **T:** 0 **P:** 0
Semester: 5	**Teaching Hours:** 36 Hours
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:**20%
External Marks: 60	**Duration of End Semester Exam(ESE):** 3hours
Total Marks: 100	**Course Status:** Elective

Additional Material Allowed in ESE: [NIL]

On completion of the course the student will have the ability to:

CO #	Course Outcomes
1.	Outline basic terminology and components in information retrieval systems.
2.	Understand the issues involved in providing an IR service on a web Scale.
3.	Compare and contrast information retrieval models and internal mechanisms.
4.	Evaluate information retrieval algorithms and give an account of the difficulties of evaluation.
5.	Identify and analyze the various aspects of a specific problem and apply the concepts of information retrieval to develop a model.
6.	Develop the ability to develop a complete IR system from scratch.

Detailed Contents:

Part A

Introduction: Introduction, History of IR, Components of IR, The IR Problem, The IR System, The Software Architecture of the IR System, The impact of the web on IR, The role of artificial intelligence (AI) in IR, IR Versus Web Search, Components of a Search engine. **[5 Hours]**

Basic IR Models: Boolean and vector-space retrieval models; ranked retrieval; text-similarity metrics; TF-IDF (term frequency/inverse document frequency) weighting; cosine similarity. **[5 Hours]**

Experimental Evaluation of IR: Performance metrics: recall, precision, and F-measure; Evaluations on benchmark text collections. **[3 Hours]**

Retrieval Utilities, Indexing and Searching: Relevance feedback; clustering; Passage-Based Retrieval; N-Grams, Regression Analysis; Thesauri; Semantic Networks; Parsing, Searching Introduction; Inverted Files; Other indices for text; Boolean queries; Sequential searching; Structural

queries; Compression. **[6 Hours]**

Part B

Cross Language Information Retrieval and Efficiency, Integrating Structured Data and Text: Introduction; Crossing the language barrier; Cross Language retrieval strategies; Cross language utilities. Duplicate Document Detection. Review of the relational model; a historical progression; Information retrieval as a relational application; Semi-structured search using a relational schema. **[6 Hours]**

Parallel Information Retrieval and Distributed Information Retrieval: Parallel text scanning; parallel indexing; Clustering and classification; Large parallel systems; A theoretic model of distributed information retrieval; Web search; Result fusion; Other architectures. **[6 Hours]**

Multimedia IR: Introduction; data modeling; Query languages; Spatial access methods; A general multimedia indexing approach; One-dimensional time series; Two-dimensional color images. **[5 Hours]**

Text Books:

1. Ricardo Baeza -Yates and Berthier Ribeiro - Neto, Modern Information Retrieval: The Concepts and Technology behind Search, ACM Press Books.
2. Bruce Croft, Donald Metzler and Trevor Strohman, Search Engines: Information Retrieval in Practice, Addison Wesley.
3. Mark Levene, An Introduction to Search Engines and Web Navigation, Wiley.

Reference Books:

1. Stefan Buettcher, Charles L. A. Clarke, Gordon V. Cormack, Information Retrieval: Implementing and Evaluating Search Engines, The MIT Press.
2. Ophir Frieder "Information Retrieval: Algorithms and Heuristics: The Information Retrieval Series ", Springer.
3. Manu Konchady, "Building Search Applications: Lucene, Ling Pipe",Gate Mustru Publishing.

E-Books and online learning material

1. C. Manning, P. Raghavan, and H. Schütze, Introduction to Information Retrieval, Cambridge University Press, 2008.

http://www.cs.utexas.edu/users/mooney/ir-course

Online Courses and video lectures

1. https://www.coursera.org/learn/text-retrieval
2. https://www.youtube.com/watch?v=sNlGHK1gz-I

Subject Code: PECS-120

Subject Name: Natural Language Processing

Programme: B.Tech. CSE	**L:** 3 **T:** 1 **P:** 0
Semester: 6	**Teaching Hours:** 36
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** Nil
External Marks: 60	**Duration of End Semester Exam (ESE):** 3 hrs
Total Marks: 100	**Elective Status:** Elective

Prerequisites: Knowledge of regular languages and parsing

Additional Material Allowed in ESE: [Scientific Calculator]

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1.	Apply the knowledge of mathematics and engineering to understand the computational properties of natural languages and the commonly used algorithms for processing linguistic information.
2.	Examine natural language processing models and algorithms using both the traditional symbolic and the more recent statistical approaches.
3.	Discuss the key concepts from natural language processing and to describe and analyze language, POS tagging and context free grammar for English language.
4.	Discover the capabilities and limitations of current natural language technologies, and some of the algorithms and techniques that underlie these technologies.
5.	Recognize the significance of models and methods of statistical natural language processing for common NLP tasks.
6.	Illustrate the concepts of morphology, syntactic analysis, semantic interpretation and pragmatics of the language, demonstrating them with different approaches.

Detailed Contents:

Part-A

Introduction: Introduction to natural language and speech processing, Steps for processing natural languages, Issues and challenges for processing of natural languages, Elements of information theory, Brief history of natural language processing. **[5 Hours]**

Morphological Analysis: Inflectional and Derivational morphology, Morphological parsing, Lexicon and Morphotactics, Finite state transducers, N-gram language models, N-gram smoothing, Entropy. **[7 Hours]**

Part-of-Speech Tagging: Word classes, Part-of-speech tagging, Tagsets, POS tagging Techniques – Rule-

based, Stochastic, Transformation-based. **[6 Hours]**

Part-B

Syntactic Analysis: Introduction to parsing, Basic parsing strategies, Top-down parsing, Bottom-up parsing, Dynamic programming – CYK parser, Issues in basic parsing methods, Earley algorithm, Parsing using Probabilistic Context Free Grammars. **[7 Hours]**

Semantic Analysis: Lexical semantics, Lexemes, Relations among lexemes and their senses, WordNet, Word Sense Disambiguation – Supervised and Un-supervised approaches. Information Extraction – Introduction to Named Entity Recognition and Relation Extraction. **[4 Hours]**

Pragmatics: Discourse, Discourse structure. Dialogue – Acts, structure, conversational agents. Language generation, Architecture for generation. **[4 Hours]**

Applications: Different application areas of natural language processing – Machine translation, Machine learning, Text categorisation and summarisation, Speech synthesis, Speech recognition, Optical character recognition, Database access, etc. **[3 Hours]**

Text Books:

1. D. Jurafsky and J. H. Martin, "Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition", Pearson Education.
2. J. Allen, "Natural Language Understanding", Second Edition, Addison Wesley.
3. Andrew Radford, Martin Atkinson, David Britain, Harald Clahsen, Andrew Spencer "Linguistics, An Introduction**"**, Second Edition, Cambridge University Press.

Reference Books:

1. T. Siddiqui and U.S. Tiwary, "Natural Languauge Processing and Information Retrieval", First Edition, Oxford University Press.
2. J. Handke, "The Structure of the Lexicon: Human Versus Machine (Natural Language Processing)", First Edition, Mouton de Gruyter.
3. Bharati, V. Chaitanya and R. Sangal, "Natural Language Processing: A Paninian Perspective", Third Edition, Prentice Hall of India.

E-Books and online learning material:

1. https://lecturenotes.in/subject/371/natural-language-processing-nlp
2. http://www.cs.virginia.edu/~kc2wc/teaching/NLP16/slides/01-intro.pdf

Online Courses and Video Lectures:

1. https://nptel.ac.in/courses/106/101/106101007/mod01lec01.mp4
2. https://nptel.ac.in/courses/106/101/106101007/mod01lec02.mp4
3. https://nptel.ac.in/courses/106/101/106101007/mod01lec03.mp4
4. https://nptel.ac.in/courses/106/101/106101007/mod01lec04.mp4
5. https://www.coursera.org/learn/language-processing

Subject Code: LPECS-110

Subject Name: Natural Language Processing Laboratory

Programme: B.Tech. CSE	**L:** 0 **T:** 0 **P:** 2
Semester: 6	**Teaching Hours:** 20
Theory/Practical: Practical	**Credits:** 1
Internal Marks: 30	**Percentage of Numerical/Design/Programming Problems:** 100%
External Marks: 20	**Duration of End Semester Exam (ESE):** 2 hrs
Total Marks: 50	**Elective Status:** Elective

Prerequisites: Experience with programming and machine learning

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1.	Apply the knowledge of engineering to understand the computational properties of natural languages and to implement the algorithms for processing linguistic information.
2.	Utilize the models and methods of statistical natural language processing for common NLP tasks such as speech recognition, machine translation, text classification, spell checking etc.
3.	Understand the key concepts of morphology, syntactic analysis for implementing POS tagging algorithms and context free grammar for English language.
4.	Identify and apply natural language processing algorithms to solve real world problems
5.	Understanding semantics and pragmatics of English language for processing.
6.	Implement, and apply state-of-the-art techniques to novel problems involving natural language data.

List of Practicals:

1. Use Naïve Bayes method to classify positive or negative sentiment in tweets.
2. Apply the orthographic e-insertion rule for morphological analysis.
3. Implement the auto-complete algorithm using an N-gram model.
4. Implement a simple auto-correct algorithm using minimum edit distance and dynamic programming.
5. Use the auto-correct algorithm to implement a simple spell checker, using unigram frequency to sort options at similar edit distance.
6. Apply the Viterbi algorithm for Part of Speech tagging.
7. Implement the top down and bottom up parsing algorithms.

8. Implement the CYK Parser using Dynamic programming.
9. Implement the Earley Algorithm using suitable example.
10. Implement named entity recognition in information extraction.

Reference Material
Manuals available in Lab.

Subject Code: PECS-121

Subject Name: System Programming

Programme: B.Tech.(CSE)	**L:** 3 **T:** 0 **P:** 0
Semester: 5	**Teaching Hours:** 36 Hours
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:**
External Marks: 60	**Duration of End Semester Exam(ESE):** 3hours
Total Marks: 100	**Course Status:** Elective

Additional Material Allowed in ESE: [NIL]

On completion of the course, the student will have the ability to:

CO #	Course Outcomes
1.	Understand the relationship between system software and machine architecture.
2.	Study the architecture of a hypothetical machine, its assembly language and macro language.
3.	Identify the need and implementation of macro processor, linkers and loaders.
4.	Determine the basics of compiler design and their applications in programming languages.
5.	Analyze the process of scanning and parsing techniques.
6.	Identify the most common pitfalls of code using debuggers and be able to locate, analyze, and fix the errors.

Detailed Contents:

Part-A

Overview of System Software: System Software, Application Software, Systems Programming, Recent Trends in Software Development, Levels of System Software, Evolution of Operating Systems, Operating System & Kernel, Functions of Operating System, Machine Structure, Evolution of components of a programming system -Assembler, Loader, Macros, Compiler. **[5 Hours]**

Language Processors: Fundamentals of Language Processing & Language Specification, Language Processing Activities, Data Structures for Language Processing - Search Data structures, Allocation Data Structures. **[4 Hours]**

Introduction to Assembly Language Program: Elements of Assembly Language Programming, A Simple Assembly Scheme, Pass Structure of Assemblers, One-Pass and Two-Pass Assemblers with reference to IBM 360 machines. **[6 Hours]**

Macro and Macro Processors: Introduction, Macro Definition and Call, Macro Expansion, Nested Macro calls, Advanced Macro Facilities, Design of a Macro Pre-processor, design of a Macro Assembler, Functions of a Macro Processor. **[5 Hours]**

Part-B

Linkers and Loaders: Linkers - Relocation of Linking Concept, Design of a Linker, Self-Relocating Programs, Linking of Overlay Structured Programs, Dynamic Linking. Loaders - Different Loading Schemes, Sequential and Direct Loaders, Compile-and-Go Loaders, General Loader Schemes, Absolute Loaders, Relocating Loaders, Linker v/s Loader. **[6 Hours]**

Scanning and Parsing: Programming Language Grammars, Classification of Grammar, Ambiguity in Grammar Specification, Scanning, Parsing, Top Down Parsing, Bottom up Parsing, Language Processor Development Tools - LEX, YACC. **[4 Hours]**

Compilers: Causes of Large Semantic Gap, Compiler and its phases – lexical, syntax and semantic analysis, intermediate code generation, code optimization and code generation. **[7 Hours]**

Interpreters and Debuggers: Interpreters - Overview of interpreters, Benefits of Interpretation. Types of Errors, Debugging Procedures, Classification of Debuggers, Dynamic/Interactive Debuggers. **[3 Hours]**

Text Books

1. Donovan J.J., “Systems Programming”, New York, Mc-Graw Hill.
2. D. M. Dhamdhere, “Systems Programming and Operating Systems”, Tata McGraw-Hill.
3. Santanu Chattopadhyay, “System Software”, Prentice-Hall India.

Reference Books

1. Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, “Compilers: Principles, Techniques, and Tools”, Pearson Education Asia.
2. Kenneth C. Louden,” Compiler Construction”, Cengage Learning.
3. Leland L. Beck, “System Software: An Introduction to Systems Programming”, Pearson Education.
4. Adam Hoover, “System Programming with C and Unix”, Pearson Education.

E-Books and online learning material

1. https://www.engineerstudyhub.com/books/systems_programming_by_donovan.pdf
2. https://lecturenotes.in/subject/29/system-programming-sp
3. https://www.tutorialsduniya.com/notes/system-programming-notes/

Online Courses and Video Lectures

1. https://nptel.ac.in/courses/106/108/106108052/ Accessed on 28/07/2020
2. https://www.coursera.org/learn/application-systems-programming Accessed on 28/07/2020
3. https://www.virtuq.com/module_details/linux-system-programmingAccessed on 28/07/2020

Subject Code: PECS-126

Subject Name: Java Programming

Programme: B.Tech. (CSE)	**L:** 3 **T:** 0 **P**: 0
Semester: 6	**Teaching Hours:** 36
Theory/Practical: Theory	**Credits:** 03
Internal Marks: 40	**Percentage of Numerical/ Design/ Programming Problems:** 30%
External Marks: 60	**Duration of End Semester Exam (ESE)**: 3 Hours
Total Marks: 100	**Course Status:** Elective

Prerequisites: Object Oriented Programming

On Completion of the course student should be able to:

CO#	Course Outcomes
1.	Apply object oriented programming techniques to propose solution pertaining to real world problems.
2.	Identify and analyze the various aspects of a specific problem and apply the concepts of classes and objects to develop object oriented model.
3.	Utilize the concept of inheritance and polymorphism to formulate a solution for complex analytical problem.
4.	Examine the errors in the developed system and resolve them by applying the knowledge of exception handling.
5.	Design console based, GUI based and web based applications by implementing various concepts like event handling, applets and database connectivity.
6.	Utilize the concept of networking to develop systems for establishing communication between client and server.

Detailed Contents

Part-A

Introduction: History of Java, Importance of Java to the internet, Java's Magic – The Byte code features of Java, Overview of Java. **[3 Hours]**

Java Basics: Data-types, Variables, Arrays, Operators, Expressions, Control statements, Type conversion, Concepts of classes and objects, Constructors, Methods, Access control, this keyword, Garbage collection, Overloading methods and constructors, Parameter passing, Recursion, Understanding static, Introducing nested and inner classes, Using command line arguments, Introduction to string handling. **[5 Hours]**

Inheritance: Basics of inheritance, Types of inheritance, Member access rules, Using super, Using final

with inheritance, Method overriding, Dynamic method dispatch, Using abstract classes. **[3 Hours]**

Packages and Interfaces: Defining a package, Accessing a package, Understanding CLASSPATH, Importing packages, Differences between classes and interfaces, defining an interface, implementing interface, Variables in interface, Extending interfaces. **[5 Hours]**

Part-B

Exception Handling: Concepts of exception handling, Exception types, Using try, catch, throw, throws and finally, Java's built in exceptions, Creating own exception subclasses. **[3 Hours]**

Multithreading: Java thread life cycle, Creating threads, Using isAlive() and join(), Synchronization, Interthread communication, Suspending, resuming, stopping threads. **[3 Hours]**

Event Handling: Delegation event model, Event classes, Sources of events, Event listeners, Handling mouse and keyboard events, Adapter classes, Inner classes. The AWT class hierarchy, User interface components – Labels, Button, Canvas, Scrollbars, Text components, Check box, Check box groups, Choices. Lists panels – Scrollpane, Dialogs, Menubar, Graphics. Understanding layout managers – Flow Layout, BorderLayout, GridLayout and CardLayout. **[8 Hours]**

Applets: Basics of applets, Differences between applets and applications, Life cycle of an applet, Types of applets, The HTML applet tag, Creating applets, Passing parameters to applets. **[4 Hours]**

Text Books:

1. E Balagurusamy "Programming with Java" 6th Edition, Tata McGraw-Hill.
2. Herbert Schildt , "The Complete Reference Java " , Tata McGraw-Hill.

Reference Books:

1. Joyce Farrell, "Java for Beginners", Cengage Learning.
2. J. Nino and F.A. Hosch, "An Introduction to programming and OO design using Java", John Wiley & Sons.
3. Y. Daniel Liang, "Introduction to Java programming", Pearson education.

E-Books and Online Learning Material

1. Introduction to Programming using Java by David J. Eck

https://www.iitk.ac.in/esc101/share/downloads/javanotes5.pdf

2. Lecture Notes on Java Programming by Surendra Baswana, IIT Kanpur
https://www.iitk.ac.in/esc101/08Jul/notes.htmlR\
3. https://www.tutorialspoint.com/java/java_pdf_version.htm Accessed on Dec. 29, 2020

Online Courses and Video Lectures:

1. https://nptel.ac.in/courses/106/105/106105191/ Accessed on Dec. 29, 2020
2. https://www.udemy.com/course/java-tutorial/ Accessed on Feb. 17, 2021
3. https://www.coursera.org/learn/object-oriented-java? Accessed on Feb. 17, 2021

Subject Code: LPECS-113

Subject Name: Java Programming Laboratory

Programme: B.Tech. CSE	**L:** 0 **T:** 0 **P:** 2
Semester: 6	**Teaching Hours:** 20
Theory/Practical: Practical	**Credits:** 1
Internal Marks:30	**Percentage of Numerical/Design/Programming Problems:** 100%
External Marks: 20	**Duration of End Semester Exam (ESE):** 2 hrs
Total Marks: 50	**Elective Status:** Elective

Prerequisites:

On Completion of the laboratory course student should be able to:

CO#	COURSE OUTCOMES
1.	Apply the knowledge of JAVA language syntax and semantics to write and execute Java programs.
2.	Create Java programs based on object oriented principles like classes, objects, constructors and inheritance.
3.	Implement the concept of applets and event handling with development of GUI interfaces for a computer program to interact with users and event based GUI handling principles.
4.	Implement exception handling techniques to make the system bug free.
5.	Apply the knowledge of event handling, applets, networking features and database connectivity to develop business oriented web based solution.
6.	Design Java programs to design a system to meet industrial needs and to solve real world problems based on client-server communication.

List of Practicals:

1. Write a program to create a pattern of stars(triangle)
2. Create a program to display the sum of digits of a given number.
3. Develop a program in java to print the factorial of a given number.
4. Write a program in java to find if a input number is palindrome or not.
5. Define a class '*student'* with its attributes, describe its constructor, overload the constructors and instantiate its objects. Demonstrate use of '*this*' keyword.

6. Demonstrate the use of inheritance with its types and also use '*super*' keyword
7. Demonstrate initialization of an array of objects using constructors and methods.
8. Demonstrate the concept of method overloading and method overriding.
9. Write a program for the demonstration of extending and implementing interfaces.
10. Create threads by using *Runnable* Interface and extending '*Thread*' class.
11. Demonstrate the use of try ,catch, throw and finally for exception handling
12. Write a program in java to demonstrate JDBC Connectivity
13. Create Applet for configuring '*Applet'* class by passing different parameters
14. Create an application to implement the concept of socket programming in Java
15. Create number counter in an Applet using '*Thread*' and '*Applet*' class
16. Write an Applet that illustrates how to process mouse click, enter, exit, press and release events. The background color changes when the mouse is entered, clicked, pressed, released or exited.
17. Implementation of a scientific calculator using event-driven programming
18. Write a java program that handles all keyboard events and shows the event name at the center of the window when keyboard event is executed.

Resource Material:

Manuals available in Lab.

Subject Code: OECS-101

Subject Name: Software Project Management

Programme: B.Tech. (CSE)	**L:** 3 **T:** 0 **P:** 0
Semester: 6	**Teaching Hours:** 36
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 30%
External Marks: 60	**Duration of End Semester Exam(ESE):** 3hr
Total Marks: 100	**Elective Status:** Open Elective

Prerequisites:
Additional Material Allowed in ESE: NIL
On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Apply the activities involved in the management of software projects.
2	Analyze the various software development environments and risk management.
3	Develop and apply the key strategies to monitor, control and quality assurance of software projects.
4	Select the appropriate planning and estimation models to better evaluate the software projects.
5	Create a strong working knowledge of ethics and professional responsibility.
6	Develop effective organizational, leadership and change skills for managing projects, teams and stakeholders.

Detailed Contents:

Part-A

Introduction to Project Management: The characteristics of software projects, Objectives of project management: time, cost and quality, Basics of Project Management, Stakeholders, Stages of Project, The Feasibility Study, Cost-benefit Analysis, Planning, Project Execution, Project and Product Life Cycles, Project Management Knowledge areas, Project Management Tools &Techniques, Project success factors, role of project manager **[5 Hours]**

Project Evaluation and Planning: Activities in Software project management, Project evaluation – Cost benefit analysis, Cash flow forecasting, Cost benefit evaluation techniques, Risk evaluation. Project planning – Stepwise project planning, Software processes and process models. Project costing, COCOMO II, Staffing pattern, Effect of schedule compression, Putnam's equation, Capers Jones estimating rules of thumb. **[6 Hours]**

Project Scheduling and Risk Management: Project sequencing and scheduling activities, Scheduling resources, Critical path analysis, Network planning, Risk management – Nature and types

of risks, Risk planning and control, Risk assessment, Hazard identification, Hazard analysis, PERT and Monte Carlo simulation techniques. **[6 Hours]**

Part-B

Monitoring and Control: Collecting data, Review techniques, Project termination review, Visualizing progress, Cost monitoring, Earned value analysis, Change control, Software Configuration Management (SCM), Managing contracts and acceptance. **[7 Hours]**

People Management: Introduction, Understanding behaviour, Organizational behaviour, Recruitment process, Motivation, The Oldman – Hackman Job Characteristics model, Stress, Health and safety. Working in teams, Decision making, Leadership, Organization and team structures. **[6 Hours]**

Software Quality Management: ISO Standards, Process capability models, Testing and software reliability, Quality plans, Test automation, Overview of project management tools. **[6 Hours]**

Text Books:

1. Bob Hughes, Mike Cotterell, "Software Project Management", Tata McGraw Hill.

2. Royce, "Software Project Management", Pearson Education.

3. Robert K. Wysocki, "Effective Software Project Management", Wiley.

Reference Books

1. Ian Sommerville, "Software Engineering", Pearson Education.

2. R.S. Pressman, "Software Engineering: A Practitioner's Approach", Tata McGraw Hill.

3. Kassem, "Software Engineering", Cengage Learning

E-Books and online learning material

1. Dwayne Phillips, "The Software Project Manager's Handbook: Principles That Work at Work", Wiley-IEEE Press, URL: https://ieeexplore.ieee.org/servlet/opac?bknumber=5989544

2. Mark Christensen; Richard H. Thayer, "The Project Manager's Guide to Software Engineering's Best Practices, Wiley-IEEE Press", URL: https://ieeexplore.ieee.org/servlet/opac?bknumber=5989224

3. Walker Royce, "Software Project Management: A Unified Framework", Addison-Wesley Professional", URL: https://dl.acm.org/doi/book/10.5555/1942096

Online Courses and Video Lectures

1. https://www.coursera.org/specializations/product-management
2. https://www.edx.org/learn/project-management

Subject Code: OECS-102

Subject Name: Object Oriented Programming using Java

Programme: B.Tech. (CSE)	**L:** 3 **T:** 0 **P:** 0
Semester: 6	**Teaching Hours:** 36
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 40%
External Marks: 60	**Duration of End Semester Exam(ESE):** 3hours
Total Marks: 100	**Course Status:** Open Elective

Additional Material Allowed in ESE: [NIL]

On completion of the course the student will have the ability to:

CO #	Course Outcomes
1.	Apply object oriented programming techniques to propose solution pertaining to real world problem.
2.	Identify and analyze the various aspects of a specific problem and apply the concepts of classes and objects to develop object oriented model.
3.	Utilize the concept of inheritance and interfaces to formulate a solution for complex analytical problem.
4.	Demonstrate an understanding of multithreaded programming.
5.	Design console based, GUI based and web based applications by implementing various concepts like applets.
6.	Examine the errors in the developed system and resolve them by applying the knowledge of exception handling.

Detailed Contents:

Part-A

Introduction to Object Oriented Programming: Difference between procedural and object oriented programming, Object oriented programming concepts – Class, Object, Data abstraction, Encapsulation, Data hiding, Inheritance and polymorphism. **[4 Hours]**

Basics of Java: Importance of Java to Internet, Bytecode, Java Virtual Machine, Difference between Java and C++, Data types, Declaration of variable, Scope and lifetime of variable, Operators – Arithmetic, Bit-wise, Relational, Boolean. Operator precedence, one dimensional and multidimensional arrays, Type conversion and casting, Control statements. **[6 Hours]**

Classes, Methods and Objects: Introduction to classes, Declaring objects, Methods, Constructors, this keyword, Overloading constructors, Garbage collection, Passing parameters to methods, Recursion, Nested and inner classes, Exploring string class. **[7 Hours]**

Part-B

Inheritance and Packages: Types of inheritance, Access modifiers – Private, Public, Protected; Overriding, Super and this keyword, Final variable, Final classes and methods, Static variable, Static method, Abstract methods and classes, Packages and interfaces, importing packages, implementing interfaces, some uses of interfaces, overloading. **[7 Hours]**

Exception Handling: Exception handling mechanism, Exception types, Uncaught exceptions, try and catch, throw and throws, finally, Built in exceptions, Creating own exception subclasses. [**4 Hours**]

Multithreading: Multithreaded programming, Thread priorities, Synchronization, Inter-thread communication, Thread class methods, Runnable interface, Suspending, Resuming and Stopping threads. [**4 Hours**]

Applets: What are Applets, The Applet Class, The Applet and HTML, Life Cycle of an Applet, The Graphics Class, Painting the Applet, User Interfaces for Applet, Adding Components to user interface, AWT (Abstract Windowing Toolkit) controls. **[4 Hours]**

Text Books:

1. Balagurusamy, "Programming in JAVA", BPB Publications.
2. T. Budd, "An Introduction to OOP", Pearson Education.
3. Patrick Naughton, Herbert Schildt, "The Complete Reference Java 2", Tata McGraw Hill Edition

Reference Books:

1. Introduction to Java Programming (Comprehensive Version), Daniel Liang, Seventh Edition, Pearson.
2. Programming in Java, Sachin Malhotra & Saurabh Chaudhary, Oxford University Press.
3. Murach's Beginning Java 2, Doug Lowe, Joel Murach and Andrea Steelman, SPD.
4. Core Java Volume-I Fundamentals, Eight Edition, Horstmann & Cornell, Pearson Education.
5. The Complete Reference, Java 2 (Fourth Edition), Herbert Schild, TMH. 6 Java Programming, D. S. Malik, Cengage Learning.

E-Books and online learning material:

1. E. Balagurusamy, "Object Oriented Programming with JAVA", Tata McGraw Hill. "https://wdn2.ipublishcentral.com//tata_mcgraw_hill/viewinsidehtml/501275211640634"
2. Simon Kendal, "Object Oriented Programming with JAVA", Simon Kendal & Ventus publishing ApS.
3. By Stephen Wong , Dung Nguyen, "Principles of Object Oriented Programming", "http://www.opentextbooks.org.hk/system/files/export/8/8163/pdf/Principles_of_ObjectOriented_Programming_8163_r.pdf"
4. R. Morelli and R. Walde, "Java, Object-Oriented Problem Solving, Third Edition", "http://www.cs.trincoll.edu/~ram/jjj/jjj-os-20170625.pdf"

Online Courses and video lectures:

1. Programming in Java, Prof. Debasis Samanta, IIT Kharagpur, Link: https://nptel.ac.in/courses/106/105/106105191/
2. Programming in C++, Prof. Partha Pratim Das, IIT Kharagpur, Link: https://nptel.ac.in/courses/106/105/106105151/
3. Java, Spoken Tutorial, IIT Bombay Link: https://spoken-tutorial.org/tutorial-search/?search_foss=Java&search_language=English

Subject Code: OECS-103

Subject Name: Cyber Laws and Ethics

Programme: B.Tech. (CSE)	**L:** 3 **T:** 0 **P:** 0
Semester: 6	**Teaching Hours:** 36
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** Nil
External Marks: 60	**Duration of End Semester Exam (ESE):** 3 hrs
Total Marks: 100	**Elective Status:** Elective

Additional Material Allowed in ESE: [NIL]

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1.	Apply the knowledge of cyber security systems to solve the complex problems of cyber crime.
2.	Make use of Intellectual Property Rights and commit to professional ethics and responsibilities and norms of the engineering practice.
3.	Recognize the need for patents and to engage in life-long learning in the broadest context of cyber security.
4.	Identify Professional and ethical issues and responsibilities.
5.	Examine the legal and policy developments in various countries for cyber space and synthesis of the information to provide valid conclusions.
6.	Analyze national and international cyber issues reaching substantiated conclusions using first principles of cyber security.

Detailed Contents:

Part-A

Cyber World and Security: Introduction to Cyberspace and Cyber law, Different components of cyber laws, Cyber law and Netizens, Attacks and Malware – The Zero-Day Attack and Mutation in delivery, Crimeware Toolkits and Trojans, Sophisticated Malware. Defensive measures for Cybersecurity – The Firewall, The Intrusion Detection System (IDS) and The Intrusion Prevention System (IPS), Virtual Private Networks (VPN), integrated defence for an enterprise network. **[6 Hours]**

Intellectual Property Rights: IPR regime in the digital society, International treaties and conventions, Business software patents, Domain name disputes and resolution, Intellectual property issues in cyberspace – Domain names and related issues, Copyright in the digital media. **[6 Hours]**

Ethics and Patents: Objectives, Rights, Assignments, Defences in case of infringement, Copyright – Objectives, Rights, Transfer of copyright. Work of employment infringement, Defences for infringement. Trademarks – Objectives, Rights, Protection of Goodwill, Infringement, Passing off and Patents in the cyber world. **[6 Hours]**

Part-B

Professional and Ethical Issues and Responsibility: Relationships with Professional Societies, codes of professional conduct, Engineering as Social Experimentation, Engineers as responsible experimenters, Ethics and history of ethics, whistle-blowing, workplace issues (harassment, discrimination), identify theft, ethical hacking. **[7 Hours]**

IT ACT 2000: Aim and objectives, Overview of the Act, Information Technology Act-2000-1 Information Technology Act-2000-2, Information Technology Act-2000-3, Information Technology Act-2000-4, Information Technology Act-2000-5, Information Technology Act2000-6, IT Act 2020 Amendments, Jurisdiction, Role of certifying authority, Regulators under IT Act, Cyber crimes – offences and contraventions and Grey areas of IT Act. **[8 Hours]**

Case Study: Case studies of infringement of cyber laws and IPR in Government sector, Corporate sector, Financial sector. **[[3 Hours**

Text Books:

1. Nandan Kamath, "Guide to information technology act, rules and regulations", Universal Law Pub.
2. Chwan-Hwa (John) Wu, J. David Irwin, "Introduction to Computer Networks and Cyber Security", CRC Press.
3. Vikas Vashishth, "Bharat's law & practice of intellectual property in India", Bharat Law House.

Reference Books:

1. William Rudolph Cornish, David Llewellyn, Tanya Aplin, "Intellectual property: patents, copyrights, trademarks and allied rights" Thomson Reuters, Sweet and Maxwell.
2. Deepti Chopra, Kieth Merrill, "Cyber cops, cyber criminals and the Internet", I.K. International.
3. Robert McGinn, "The Ethically Responsible Engineer: Concepts and Cases for Students and Professionals" John Wiley and Sons.

E-Books and online learning material:

1. Zeinab KarakeShalhoub, "Cyber law and cyber security in developing and emerging economies", Edward Elgar, USA. https://www.pdfdrive.com/download.pdf?id=185019171&h=f9b11e92d249b458f29a3e25bd0da620&u=cache&ext=pdf.

Online Courses and Video Lectures:

1. https://www.youtube.com/watch?v=xeDZalBaLkQ.
2. https://www.youtube.com/watch?v=yNRIfo3q4Zo.
3. https://www.youtube.com/watch?v=74XncBZEnN4.
4. https://www.youtube.com/watch?v=s-SFB-EW_4A.
5. https://www.youtube.com/watch?v=1vQhSm5_UqY.

Subject Code: OECS-104
Subject Name: Data Structures

Programme: B.Tech.(CSE)	**L:** 3 **T:** 0 **P:** 0
Semester: 6	**Teaching Hours:** 37
Theory/Practical: Theory	**Credits:** 3
Internal Marks: 40	**Percentage of Numerical/Design/Programming Problems:** 50%
External Marks: 60	**Duration of End Semester Exam (ESE):** 3hrs
Total Marks: 100	**Elective Status:** Open Elective

Prerequisites: Knowledge of Programming for Problem Solving and OOPS

Additional Material Allowed in ESE: [Scientific Calculator]

On completion of the course, the student will have the ability to:

CO#	Course Outcomes
1	Apply knowledge of statistics and programming skills to solve complex engineering problems related to data structures.
2	Make use of Research based knowledge to identify the appropriate data structure and provide better solution to reduce space and time complexity.
3	Identify, Formulate and analyse data structure to develop skills and understand their applications to perform operations on it.
4	Design appropriate algorithm for autonomous realization of sub-programs to model complex engineering activities.
5	Demonstrate various methods of organizing large amounts of data and recognize systematic way to retrieve data and solve problems.
6	Formulate new solutions for programming problems or improve existing code using learned algorithms and data structures.

Detailed Contents:

Part-A

Basic concepts: Concept of data type, Linear and non-linear data structures, Data structures versus data types, Operations on data structures, Algorithm complexity and Asymptotic notations. **[2 Hours]**

Arrays: Linear and multi-dimensional arrays and their representation, Operations on arrays, Sparse matrices and their storage. **[2 Hours]**

Stacks: Sequential representation of stacks, Operations on stacks, Application of stacks – parenthesis checker, Conversion from infix to postfix, Evaluation of postfix expressions, implementing recursive functions. **[5 Hours]**

Queues: Sequential representation of queue, Types of queue- Linear Queue, Circular Queue, Deque, Priority Queue, Operations on each types of Queues, Applications of Queues. **[4 Hours]**

Linked List: Definition and representation of Linked list, Types of Linked list- Linear linked list, Doubly linked list, Circular linked list and Header linked list and their operations, Application of linked lists, Garbage collection and compaction, Linked representation of Stack and Queues. **[6 Hours]**

Part-B

Trees: Basic terminology, Sequential and linked representations of trees, Different types of Trees- Binary Tree, Binary search tree. Operations on each of the trees. Application of Binary Trees. **[5 Hours]**

Graphs: Basic terminology, Representation of graphs – Adjacency matrix, Adjacency list. Operations on graph, Traversal of a graph – Breadth first search, Depth first search. Shortest path algorithms – Dijkstra's and Floyd. Minimum spanning tree – Prim and Kruskal. Applications of graphs. **[4 Hours]**

Heaps: Representing a heap in memory, Operations on heaps. **[2 Hours]**

Hashing and Hash Tables: Introduction to hash table, Hash functions. **[2 Hours]**

Searching and Sorting: Linear and binary search techniques, Sorting methods – Bubble sort, Selection sort, Insertion sort, Quick sort, Merge sort, Shell sort and radix sort. Complexities of searching and sorting algorithms. **[5 Hours]**

Text Books

1. Seymour Lipschutz, "Data Structures", Schaum's Outline Series, Tata McGraw Hill.
2. Sartaj Sahni, "Data Structures, Algorithms and Applications in C++", Tata McGraw Hill.

Reference Books

1. Michael T. Goodrich, Roberto Tamassia, & David Mount, "Data Structures and Algorithms in C++", Wiley India.
2. Kruse, "Data Structures & Program Design", Prentice Hall of India.
3. Y. Langsa, M.J. Augenstein, A.M. Tanenbaum, "Data structures using C and C++", Prentice Hall ofIndia.
4. Vishal Goyal, Lali Goyal, Pawan Kumar, "Simplified Approach to Data Structures", ShroffPublications and Distributors

E-Books and online learning material

1. Data Structures and Algorithms: by Granville Barnett, and Luca Del Tongo.
https://apps2.mdp.ac.id/perpustakaan/ebook/Karya%20Umum/Dsa.pdf

2. Data Structures and Algorithms in JAVA :by Michael T. Goodrich and Roberto Tamassiahttp://enos.itcollege.ee/~jpoial/algorithms/GT/Data%20Structures%20and%20Algorithms%20in%20Java%20Fourth%20Edition.pdf

Online Courses and Video Lectures

1. https://nptel.ac.in/courses/106102064/
2. https://nptel.ac.in/courses/106106133/
3. https://nptel.ac.in/courses/106106145/