

**Bachelor of Technology (Computer Science & Engineering)**  
**Scheme of Studies/Examination**  
**Semester IV**

S. No.	Course No.	Subject	L:T:P	Hours/Week	Examination Schedule (Marks)				Duration of Exam (Hrs)
					Major Test	Minor Test	Practical	Total	
1	AS-201 N	Mathematics-III	3:1:0	4	75	25	0	100	3
2	CSE-202 N	Object Oriented Programming	3:1:0	4	75	25	0	100	3
3	CSE-204 N	Internet Fundamental	3:0:0	3	75	25	0	100	3
4	CSE-206 N	Digital Data Communication	3:1:0	4	75	25	0	100	3
5	CSE-208 N	Microprocessor & Interfacing	3:1:0	4	75	25	0	100	3
6	CSE-210 N	Operating System	3:1:0	4	75	25	0	100	3
7	CSE-212 N	Object Oriented Programming Lab	0:0:3	3	0	40	60	100	3
8	CSE-214 N	Microprocessor Lab	0:0:3	3	0	40	60	100	3
9	CSE-216 N	Internet Lab	0:0:3	3	0	40	60	100	3
		<b>Total</b>		<b>32</b>	<b>450</b>	<b>270</b>	<b>180</b>	<b>900</b>	

10	MPC 201 N	Environment Studies*	3:0:0	3	75	25		100	3
----	-----------	----------------------	-------	---	----	----	--	-----	---

**\*MPC-201 is a mandatory course which will be a non credit subject and student has to get pass marks in order to qualify for the Degree award**

GEETA ENGINEERING COLLEGE

AS-201 N	Mathematics-III					
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
3	1	-	75	25	100	3
<b>Purpose</b>	<b>To provide the conceptual knowledge of Engineering mathematics</b>					
<b>Course Outcomes</b>						
<b>CO 1</b>	To study various fundamental concepts of Fourier series and Fourier Transformation.					
<b>CO 2</b>	To study and understand the functions of a complex variables.					
<b>CO 3</b>	To study the Probability Distributions.					
<b>CO 4</b>	To study the linear programming problem formulation.					

#### UNIT – I

Fourier series: Euler's Formulae, Conditions for Fourier expansions, Fourier expansion of functions having points of discontinuity, change of interval, Odd & even functions, Half-range series.

Fourier Transforms: Fourier integrals, Fourier transforms, Fourier cosine and sine transforms. Properties of Fourier transforms, Convolution theorem, Parseval's identity, Relation between Fourier and Laplace transforms, Fourier transforms of the derivatives of a function, Application to boundary value problems.

#### UNIT-II

Functions of Complex Variables: Functions of a complex variable, Exponential function, Trigonometric, Hyperbolic and Logarithmic functions, limit and continuity of a function, Differentiability and analyticity.

Cauchy-Riemann equations, Necessary and sufficient conditions for a function to be analytic, Polar form of the Cauchy-Riemann equations, Harmonic functions, Application to flow problems, Conformal transformation, Standard transformations (Translation, Magnification & rotation, inversion & reflection, Bilinear).

#### UNIT-III

Probability Distributions : Probability, Baye's theorem, Discrete & Continuous probability distributions, Moment generating function, Probability generating function, Properties and applications of Binomial, Poisson and normal distributions.

#### UNIT-IV

Linear Programming : Linear programming problems formulation, Solution of Linear Programming Problem using Graphical method, Simplex Method, Dual-Simplex Method.

#### **Text Book**

1. Higher Engg. Mathematics: B.S. Grewal
2. Advanced Engg. Mathematics: E. Kreyzig

#### **Reference Book**

1. Complex variables and Applications: R.V. Churchill; Mc. Graw Hill
2. Engg. Mathematics Vol. II S.S. Sastry; Prentice Hall of India.
3. Operation Research: H.A. Taha
4. Probability and statistics for Engineer: Johnson. PHI.

CSE-202 N	Object Oriented Programming					
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
3	1	-	75	25	100	3 Hour
<b>Purpose</b>	<b>To introduce the principles and paradigms of Object Oriented Programming Language for design and implement the Object Oriented System</b>					
<b>Course Outcomes (CO)</b>						
<b>CO 1</b>	To introduce the basic concepts of object oriented programming language and the its representation					
<b>CO 2</b>	To allocate dynamic memory, access private members of class and the behavior of inheritance and its implementation.					
<b>CO 3</b>	To introduce polymorphism, interface design and overloading of operator.					
<b>CO 4</b>	To handle backup system using file, general purpose template and handling of raised exception during programming					

#### Unit-1

Introduction to C++, C++ Standard Library, Illustrative Simple C++ Programs. Header Files, Namespaces, Application of object oriented programming.

Object Oriented Concepts, Introduction to Objects and Object Oriented Programming, Encapsulation, Polymorphism, Overloading, Inheritance, Abstract Classes, Accessifier (public/ protected/ private), Class Scope and Accessing Class Members, Controlling Access Function, Constant, Class Member, Structure and Class

#### Unit-2

Friend Function and Friend Classes, This Pointer, Dynamic Memory Allocation and Deallocation (New and Delete), Static Class Members, Constructors, parameter Constructors and Copy Constructors, Destructors, Introduction of inheritance, Types of Inheritance, Overriding Base Class Members in a Derived Class, Public, Protected and Private Inheritance, Effect of Constructors and Destructors of Base Class in Derived Classes.

#### Unit-3

Polymorphism, Pointer to Derived class, Virtual Functions, Pure Virtual Function, Abstract Base Classes, Static and Dynamic Binding, Virtual Destructors.

Fundamentals of Operator Overloading, Rules for Operators Overloading, Implementation of Operator Overloading Like <<>> Unary Operators, Binary Operators.

#### Unit-4

Text Streams and binary stream, Sequential and Random Access File, Stream Input/ Output Classes, Stream Manipulators.

Basics of C++ Exception Handling, Try, Throw, Catch, multiple catch, Re-throwing an Exception, Exception specifications.

Templates: Function Templates, Overloading Template Functions, Class Template, Class Templates and Non-Type Template arguments.

#### Text Books:

- Object Oriented Programming in Turbo C++ by Robert Lafore, 1994, The WAITE Group Press.
- The complete reference C ++ by Herbert shieldt Tata McGraw Hill

#### References Books

- Shukla, Object Oriented Programming in c++, wiley india
- C++ How to Program by H M Deitel and P J Deitel, 1998, Prentice Hall
- Programming with C++ By D Ravichandran, 2003, T.M.H

CSE-204 N		Internet Fundamentals				
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
3	0	-	75	25	100	3
Purpose	To provide the conceptual knowledge of Internet and methodologies used in web and secure internet communication and networking.					
<b>Course Outcomes</b>						
CO 1	To study various fundamental concepts of Internetworking techniques with their characteristics.					
CO 2	To study and understand the requirements for world-wide-web formats and techniques.					
CO 3	To study the E-mail functioning and basics of HTML, XML and DHTML languages.					
CO 4	To study the functioning of Servers and Privacy and Security related mechanisms.					

#### UNIT-1 : THE INTERNET

Introduction to networks and internet, history, Internet, Intranet & Extranet, Working of Internet, Internet Congestion, internet culture, business culture on internet. Collaborative computing & the internet. Modes of Connecting to Internet, Internet Service Providers (ISPs), Internet address, standard address, domain name, DNS, IP.v6. Modems, Speed and time continuum, communications software; internet tools.

#### UNIT-II : WORLD WIDE WEB

Introduction, Miscellaneous Web Browser details, searching the www: Directories search engines and meta search engines, search fundamentals, search strategies, working of the search engines, Telnet and FTP, HTTP, Gopher Commands, TCP/IP. Introduction to Browser, Coast-to-coast surfing, hypertext markup language, Web page installation, Web page setup, Basics of HTML & formatting and hyperlink creation. Using FrontPage Express, Plug-ins.

#### UNIT-III : INTERNET PLATFORM AND MAILING SYSTEMS

Introduction, advantages and disadvantages, User Ids, Pass words, e-mail addresses, message components, message composition, mailer features, E-mail inner workings, E-mail management, MIME types, Newsgroups, mailing lists, chat rooms, secure-mails, SMTP, PICO, Pine, Library cards catalog, online ref. works.

**Languages:** Basic and advanced HTML, Basics of scripting languages – XML, DHTML, Java Script.

#### UNIT-IV : SERVERS

Introduction to Web Servers: PWS, IIS, Apache, Microsoft Personal Web Server. Accessing & using these servers.

**Privacy and security topics:** Introduction, Software Complexity, Attacks, security and privacy levels, security policy, accessibility and risk analysis, Encryption schemes, Secure Web document, Digital Signatures, Firewalls, Intrusion detection systems

#### Text Book:

- Internet & World Wide Programming, Deitel, Deitel & Nieto, 2012, Pearson Education
- Fundamentals of the Internet and the World Wide Web, Raymond Greenlaw and Ellen Hepp, TMH- 2012

#### Reference Books:

- Complete idiots guide to java script, Aron Weiss, QUE, 2013
- Network firewalls, Kironjeet syan -New Rider Pub.2014
- Networking Essentials – Firewall Media. Latest-2015
- www.secnf.com
- www.hackers.com
- Alfred Gkossbrenner-Internet 101 Computing MGH, 2013

CSE-206 N	Digital Data Communication					
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
3	1	-	75	25	100	3
<b>Purpose</b>	To provide the conceptual knowledge of data preparation and signal transmission methodologies used in data communication and networking.					
<b>Course Outcomes</b>						
<b>CO 1</b>	To study various analog communication techniques and with their characteristics.					
<b>CO 2</b>	To study and understand the requirements for analog/digital data to analog/digital signal conversion techniques.					
<b>CO 3</b>	To study the error and flow control techniques in communication and networking.					
<b>CO 4</b>	To study the concept of multiplexing and applied multiple access techniques specially in satellite communication.					

### UNIT-1

#### MODULATION TECHNIQUES

Basic constituents of Communication Systems need of modulation, Amplitude modulation, spectrum of AM wave, modulation index, DSBSC modulation, SSB Modulation, vestigial side band modulation.  
ANGLE MODULATION: Frequency and Phase Modulation, spectrum of FM Wave, modulation index and Bandwidth of FM Signal, NBFM and WBFM.

### UNIT-II

#### DATA ENCODING

Digital data, Digital signals: Encoding schemes: NRZ-L, NRZ-I, Manchester-Diff-Manchester-encoding, Pseudoternary-Bipolar-AMI, B8ZS- HDB3 – Evaluation factors- Digital data, analog signals: Encoding Techniques –ASK-FSK-PSK-QPSK-Performance comparison-Analog data, digital signals: Quantization-Sampling theorem-PCM-Delta modulation-Errors- comparison- Analog Data, analog signals: Need for modulation -0 Modulation methods – Amplitude modulation- Angle modulation- Comparison.

### UNIT-III

#### DIGITAL DATA COMMUNICATION TECHNIQUES

Asynchronous and synchronous transmission –Error Detection techniques: Parity checks – Cycle redundancy checks-Checksum-Error Correcting codes: Forwards and backward error corrections, Transmission media. Communication Topologies.  
DTE & DCE interface: Characteristics of DTE-DCE interface. Interfaces: Rs-232-C, Rs-449/422, A/423-A.

### UNIT-IV

#### SATELITE COMMUNICATION

**Multiplexing:** Advantages – Types of Multiplexing – FDM – Synchronous TDM – Statistical TDM or Asynchronous TDM, Study of their characteristics.

**Satellite Communication Systems:** Satellite parameters and configurations – Capacity allocation, Frequency Division FDMA; Time Division TDMA- Fixed assigned multiple access (FAMA), Demand assign multiple access (DAMA) – The concept of spread spectrum: FHSS, DSSS – CDMA – Transmission and reception.

#### TEXT BOOKS

1. Forouzen, "Data Communication & Networking", Tata Mcgraw Hill
2. Proakin, "Digital Communications", Mc Graw Hill.
3. W. Stallings, "Wireless Communication and Networks" Pearson.

#### REFERENCES

1. Stallings, "Data & computer Communications", PHI.
2. Roden, "Digital & Data Communication Systems", PHI.
3. Irvine, Data communications & Networks An engineering approach, wiley india

CSE-208 N						
Microprocessor & Interfacing						
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
3	1	-	75	25	100	3
<b>Purpose</b>	To learn the architecture and programming of Intel family microprocessors and its interfacing.					
<b>Course Outcomes</b>						
<b>CO 1</b>	To study the Architecture of 8085 microprocessors					
<b>CO 2</b>	To learn the architecture 8086 Microprocessor and its interfacing to memories					
<b>CO 3</b>	To learn the instruction set of 8086 Microprocessor and assembly language programming of 8086 Microprocessor.					
<b>CO 4</b>	To learn interfacing of interrupts, basic I/O and DMA with 8086 Microprocessor					

#### Unit I

Evolution of Microprocessor, Introduction to 8085 - 8085 architecture - Pin Details - Addressing Modes - Instruction Set and Assembler Directives, Instruction Timing Diagram.

#### UNIT-II

**8086 CPU ARCHITECTURE:** 8086 Block diagram; description of data registers, address registers; pointer and index registers, PSW, Queue, BIU and EU. 8086 Pin diagram descriptions. Generating 8086 CLK and reset signals using 8284. WAIT state generation. Microprocessor BUS types and buffering techniques, 8086 minimum mode and maximum mode CPU module. MAIN MEMORY SYSTEM DESIGN: Memory devices, 8086 CPU Read/Write timing diagrams in minimum mode and maximum mode. Address decoding techniques. Interfacing SRAMS; ROMS/PROMS. Interfacing and refreshing DRAMS.

#### UNIT-III

**8086 INSTRUCTION SET:** Instruction formats, addressing modes, Data transfer instructions, string instructions, logical instructions, arithmetic instructions, transfer of control instructions; process control instructions; Assembler directives.

**8086 PROGRAMMING TECHNIQUES:** Writing assembly Language programs for logical processing, arithmetic processing, timing delays; loops, data conversions.

#### UNIT-IV

**BASIC I/O INTERFACE:** Parallel and Serial I/O Port design and address decoding. Memory mapped I/O Vs Isolated I/O Intel's 8255 and 8251- description and interfacing with 8086. ADCs and DACs, - types, operation and interfacing with 8086. Interfacing Keyboards, alphanumeric displays, multiplexed displays, and stepper motor, optical encoder with 8086.

**INTERRRUPTS AND DMA:** 8086 Interrupt mechanism; interrupt types and interrupt vector table. Applications of interrupts, Intel's 8259. DMA operation. Intel's 8237.

#### Text Books:

- Barry B. Brey, "The Intel Microprocessor 8086/8088, 80186", Pearson Education, Eighth Edition, 2009
- D.V. Hall, Microprocessors and Interfacing, McGraw Hill 2nd ed.

#### Reference Books:

- Liu, Gibson, "Microcomputer Systems: The 8086/88 Family", 2nd Edition, PHI, 2005
- Kenneth Ayala, "The 8086 Microprocessor: Programming & Interfacing the PC", Cengage Learning, Indian Edition, 2008
- Kip Irvine, "Assembly language for IBM PC", PHI, 2nd Edition, 1993
- Peter Abel, "Assembly language programming", Pearson Edu, 5th Edition, 2002
- Uffenback, "The 8086 Family Design" PHI, 2<sup>nd</sup> Edition.
- Walter A Triebel and Avtar Singh; The 8088 and 8086 Microprocessors, Programming, Interfacing, Software, Hardware and Applications, Fourth Edition, Pearson Education.

CSE-210 N	Operating Systems					
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
3	1	-	75	25	100	3
<b>Purpose</b>	<b>To familiarize the students with the basics of Operating Systems</b>					
<b>Course Outcomes</b>						
<b>CO1</b>	To understand the structure and functions of Operating system.					
<b>CO 2</b>	To learn about processes, threads and scheduling algorithms.					
<b>CO 3</b>	To understand the principle of concurrency.					
<b>CO 4</b>	To understand the concept of deadlocks.					
<b>CO5</b>	To learn various memory management schemes.					
<b>CO6</b>	To study I/O management and file systems.					
<b>CO7</b>	To study the concept of protection and security.					

#### UNIT 1

**Introduction:** Introduction to OS. Operating system functions, Different types of O.S.: batch process, multi-programmed, time-sharing, real-time, distributed, parallel.

**System Structure:** Computer system operation, I/O structure, storage structure, storage hierarchy, different types of protections, operating system structure (simple, layered, virtual machine), O/S services, system calls.

#### UNIT II

**CPU scheduling:** scheduling criteria, preemptive & non-preemptive scheduling, scheduling algorithms, algorithm evaluation, multi-processor scheduling.

**Threads:** overview, benefits of threads, user and kernel threads.

**Process Management:** Concept of processes, process states, process control, co-operating processes, inter-process communication.

**Process Synchronization:** background, critical section problem, critical region, synchronization hardware, Classical problems of synchronization, semaphores.

#### UNIT III

**Deadlocks:** Concept of deadlock, deadlock characterization, deadlock prevention, deadlock avoidance, deadlock detection, recovery from deadlock.

**Memory Management:** background, logical vs. physical address space, contiguous memory allocation, paging, segmentation, segmentation with paging. Concept of fragmentation.

**Virtual Memory:** background, demand paging, concept of page replacement, page replacement algorithms, allocation of frames, thrashing.

#### UNIT IV

**File Systems:** file concept, file organization and access methods, allocation methods, directory structure, free-space management

**I/O Management:** I/O hardware, polling, interrupts, DMA, kernel I/O subsystem (scheduling, buffering, caching, spooling and device reservation)

**Disk Management:** disk structure, disk scheduling (FCFS, SSTF, SCAN,C-SCAN), disk reliability, disk Performance parameters

**Protection & Security:**

Goals of protection and security, security attacks, authentication, program threats, system threats, threat monitoring.

**Case studies:** UNIX file system, Windows file system

#### Text Books:

1. Operating Systems : Internals and Design Principles, William Stallings, Pearson
2. Naresh Chauhan, Principles of Operating Systems, Oxford University Press, 2014

#### Reference books:

1. Operating System Concepts", Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne, Wiley
2. Operating systems: a concept based approach", Dhananjay M. Dhamdhare, McGraw Hill.
3. Operating Systems Design and Implementation", (Prentice Hall Software Series) Andrew S Tanenbaum and Albert S Woodhull
4. Taub & Schilling, Principles of Communication Systems, TMH.
5. Mithal G K, Radio Engineering, Khanna Pub.

CSE-212 N	Object Oriented Programming Lab					
Lecture	Tutorial	Practical	Minor Test	Practical	Total	Time
0	0	3	40	60	100	3 Hour
<b>Purpose</b>	<b>To introduce the principles and paradigms of Object Oriented Programming Language for design and implement the Object Oriented System</b>					
<b>Course Outcomes (CO)</b>						
<b>CO 1</b>	To introduce the basic concepts of object oriented programming language and the its representation					
<b>CO 2</b>	To allocate dynamic memory, access private members of class and the behavior of inheritance and its implementation.					
<b>CO 3</b>	To introduce polymorphism, interface design and overloading of operator.					
<b>CO 4</b>	To handle backup system using file, general purpose template and handling of raised exception during programming					

**Q1.** Raising a number  $n$  to a power  $p$  is the same as multiplying  $n$  by itself  $p$  times. Write a function called `power()` that takes a double value for  $n$  and an int value for  $p$ , and returns the result as double value. Use a default argument of 2 for  $p$ , so that if this argument is omitted, the number will be squared. Write a `main()` function that gets values from the user to test this function.

**Q2.** A point on the two dimensional plane can be represented by two numbers: an X coordinate and a Y coordinate. For example, (4,5) represents a point 4 units to the right of the origin along the X axis and 5 units up the Y axis. The sum of two points can be defined as a new point whose X coordinate is the sum of the X coordinates of the points and whose Y coordinate is the sum of their Y coordinates. Write a program that uses a structure called `point` to model a point. Define three points, and have the user input values to two of them. Then set the third point equal to the sum of the other two, and display the value of the new point. Interaction with the program might look like this:

Enter coordinates for P1: 3 4

Enter coordinates for P2: 5 7

Coordinates of P1 + P2 are : 8, 11

**Q3.** Create the equivalent of a four function calculator. The program should request the user to enter a number, an operator, and another number. It should then carry out the specified arithmetical operation: adding, subtracting, multiplying, or dividing the two numbers. (It should use a switch statement to select the operation). Finally it should display the result. When it finishes the calculation, the program should ask if the user wants to do another calculation. The response can be 'Y' or 'N'. Some sample interaction with the program might look like this.

Enter first number, operator, and second number: 10/ 3

Answer = 3.333333

Do another (Y/ N)? Y

Enter first number, operator, second number 12 + 100

Answer = 112

Do another (Y/ N)? N

**Q4.** A phone number, such as (212) 767-8900, can be thought of as having three parts: the area code (212), the exchange (767) and the number (8900). Write a program that uses a structure to store these three parts of a phone number separately. Call the structure `phone`. Create two structure variables of type `phone`. Initialize one, and have the user input a number for the other one. Then display both numbers. The interchange might look like this:

- Enter your area code, exchange, and number: 415 555 1212

- My number is (212) 767-8900

- Your number is (415) 555-1212

**Q5.** Create two classes `DM` and `DB` which store the value of distances. `DM` stores distances in metres and centimeters and `DB` in feet and inches. Write a program that can read values for the class objects and add one object of `DM` with another object of `DB`. Use a friend function to carry out the addition operation. The object that stores the results maybe a `DM` object or `DB` objects, depending on the units in which the results are required. The display should be in the format of feet and inches or metres and centimetres depending on the object on display.

**Q6.** Create a class `rational` which represents a numerical value by two double values- `NUMERATOR` & `DENOMINATOR`. Include the following public member Functions:



- constructor with no arguments (default).
- constructor with two arguments.
- void reduce( ) that reduces the rational number by eliminating the highest common factor between the numerator and denominator.
- Overload + operator to add two rational number.
- Overload >> operator to enable input through cin.
- Overload << operator to enable output through cout.

Write a main ( ) to test all the functions in the class.

**Q7.** Consider the following class definition

```
class father {
protected : int age;
public;
father (int x) {age = x;}
virtual void iam ( )
{ cout << "I AM THE FATHER, my age is : "<< age<< endl;}
};
```

Derive the two classes son and daughter from the above class and for each, define iam ( ) to write our similar but appropriate messages. You should also define suitable constructors for these classes. Now, write a main ( ) that creates objects of the three classes and then calls iam ( ) for them. Declare pointer to father. Successively, assign addresses of objects of the two derived classes to this pointer and in each case, call iam ( ) through the pointer to demonstrate polymorphism in action.

**Q8.** Write a program that creates a binary file by reading the data for the students from the terminal. The data of each student consist of roll no., name ( a string of 30 or lesser no. of characters) and marks.

**Q9.** A hospital wants to create a database regarding its indoor patients. The information to store include

- a) Name of the patient
- b) Date of admission
- c) Disease
- d) Date of discharge

Create a structure to store the date (year, month and date as its members). Create a base class to store the above information. The member function should include functions to enter information and display a list of all the patients in the database. Create a derived class to store the age of the patients. List the information about all the to store the age of the patients. List the information about all the pediatric patients (less than twelve years in age).

**Q10.** Make a class **Employee** with a name and salary. Make a class **Manager** inherit from **Employee**. Add an instance variable, named department, of type string. Supply a method to **to String** that prints the manager's name, department and salary. Make a class **Executive** inherits from **Manager**. Supply a method **to String** that prints the string "**Executive**" followed by the information stored in the **Manager** superclass object. Supply a test program that tests these classes and methods.

**Q11.** Imagine a tollbooth with a class called toll Booth. The two data items are a type unsigned int to hold the total number of cars, and a type double to hold the total amount of money collected. A constructor initializes both these to 0. A member function called payingCar ( ) increments the car total and adds 0.50 to the cash total. Another function, called nopayCar ( ), increments the car total but adds nothing to the cash total. Finally, a member function called displays the two totals. Include a program to test this class. This program should allow the user to push one key to count a paying car, and another to count a nonpaying car. Pushing the ESC key should cause the program to print out the total cars and total cash and then exit.

**Q12.** Write a function called reversit ( ) that reverses a string (an array of char). Use a for loop that swaps the first and last characters, then the second and next to last characters and so on. The string should be passed to reversit ( ) as an argument. Write a program to exercise reversit ( ). The program should get a string from the user, call reversit ( ), and print out the result. Use an input method that allows embedded blanks. Test the program with Napoleon's famous phrase, "Able was I ere I saw Elba".

25

**Q13.** Create some objects of the string class, and put them in a Deque-some at the head of the Deque and some at the tail. Display the contents of the Deque using the forEach ( ) function and a user written display function. Then search the Deque for a particular string, using the first That ( ) function and display any strings that match. Finally remove all the items from the Deque using the getLeft ( ) function and display each item. Notice the order in which the items are displayed: Using getLeft ( ), those inserted on the left (head) of the Deque are removed in "last in first out" order while those put on the right side are removed in "first in first out" order. The opposite would be true if getRight ( ) were used.

**Q14.** Assume that a bank maintains two kinds of accounts for customers, one called as savings account and the other as current account. The savings account provides compound interest and withdrawal facilities but no

cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed. Create a class account that stores customer name, account number and type of account. From this derive the classes

cur\_acct and sav\_acct to make them more specific to their requirements. Include necessary member functions in order to achieve the following tasks:

- a) Accept deposit from a customer and update the balance.
- b) Display the balance.
- c) Compute and deposit interest.
- d) Permit withdrawal and update the balance.
- e) Check for the minimum balance, impose penalty, necessary and update the balance.
- f) Do not use any constructors. Use member functions to initialize the class members.

**Q15.** Create a base class called shape. Use this class to store two double type values that could be used to compute the area of figures. Derive two specific classes called triangle and rectangle from the base shape. Add to the base class, a member function get\_data( ) to initialize baseclass data members and another member function display\_area( ) to compute and display the area of figures. Make display\_area ( ) as a virtual function and redefine this function in the derived classes to suit their requirements. Using these three classes, design a program that will accept dimensions of a triangle or a rectangle interactively and display the area.

Remember the two values given as input will be treated as lengths of two sides in the case of rectangles and as base and height in the case of triangles and used as follows:

Area of rectangle =  $x * y$

Area of triangle =  $\frac{1}{2} * x * y$

GEETA ENGINEERING COLLEGE

CSE-214 N	Microprocessor Lab					
Lecture	Tutorial	Practical	Minor Test	Practical	Total	Time
0	0	3	40	60	100	3 Hour
Purpose	Write the efficient Assembly Language Program for different problem statements and implement different system interfacing.					
Course Outcomes						
CO 1	Understanding different steps to develop program such as Problem definition, Analysis, Design of logic, Coding, Testing, Maintenance (Modifications, error corrections, making changes etc.)					
CO 2	To be able to apply different logics to solve given problem.					
CO 3	To be able to write program using different implementations for the same problem					
CO 4	Use of programming language constructs in program implementation					

**Write an Assembly Language Program to**

1. Add / Sub two 16 bit numbers.
2. Find sum of series of numbers.
3. Multiply two 16 bit unsigned/ signed numbers.
4. Divide two unsigned/ signed numbers (32/16 , 16/8, 16/16, 8/8 )
5. Add / Sub / multiply / Divide two BCD numbers.
6. Find smallest/ largest number from array of n numbers.
7. Arrange numbers in array in ascending/ descending order.
8. Perform block transfer data using string instructions / without using string instructions.
9. Compare two strings using string instructions / without using string instructions.
10. Display string in reverse order, string length, Concatenation of two strings.
11. Convert Hex to Decimal, Decimal to Hex.
12. To find 1's and 2's complement of a number.

CSE-216 N	Internet Lab					
Lecture	Tutorial	Practical	Minor Test	Practical	Total	Time
0	0	3	40	60	100	3 Hour
Purpose	Learn the internet and design different web pages using HTML and installation of different MODEMS.					
<b>Course Outcomes</b>						
CO 1	Understanding different PC software and their applications					
CO 2	To be able to learn HTML.					
CO 3	To be able to write Web pages using HTML.					
CO 4	To be able to install modems and understand the e-mail systems.					

**PC Software:** Application of basics of MS Word 2000, MS Excel 2000, MS Power Point 2000, MS Access 2000, HTML

1. To prepare the Your Bio Data using MS Word
2. To prepare the list of marks obtained by students in different subjects and show with the help of chart/graph the average, min and max marks in each subject.
3. Prepare a presentation explaining the facilities/infrastructure available in your college/institute.
4. Design Web pages containing information of the Deptt.

**HTML Lists:**

1. Create a new document that takes the format of a business letter. Combine <P> and <BR> tags to properly separate the different parts of the documents. Such as the address, greeting, content and signature. What works best for each?
2. Create a document that uses multiple <BR> and <P> tags, and put returns between <PRE> tags to add blank lines to your document see if your browser sends them differently.
3. Create a document using the <PRE>tags to work as an invoice or bill of sale, complete with aligned dollar values and a total. Remember not to use the Tab key, and avoid using emphasis tags like <B> or <EM> within your list.
4. Create a seven-item ordered list using Roman numerals. After the fifth item, increase the next list value by 5.
5. Beginning with an ordered list, create a list that nests both an unordered list and a definition list.
6. Use the ALIGN attribute of an <IMG> tags to align another image to the top of the first image.. play with this feature, aligning images to TOP, MIDDLE and BOTTOM.
7. Create a 'table of contents' style page (using regular and section links) that loads a different document for each chapter or section of the document.

**Internet:**

1. Instilling internet & external modems, NIC and assign IP address.
2. Study of E-mail system.
3. Create your own mail-id in yahoo and indiatimes.com.
4. Add names (mail-id's) in your address book, compose and search an element.

MPC-201N		ENVIRONMENTAL STUDIES				
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
3	-	-	75	25	100	3 Hrs.
<b>Purpose</b>	To learn the multidisciplinary nature, scope and importance of Environmental Studies					
<b>Course Outcomes</b>						
<b>CO1</b>	Basic concepts of Various kinds of Microscopy and Centrifugation Techniques					
<b>CO2</b>	To learn the theoretical and practical aspects of Electrophoresis and Chromatography Techniques					
<b>CO3</b>	To learn the concepts of different kinds of Spectroscopy and Colourimetry					
<b>CO4</b>	To understand the concept of radioisotope techniques and their applications in research					

### UNIT 1

The multidisciplinary nature of environmental studies. Definition, Scope and Importance. Need for public awareness. Natural Resources: Renewable and Non-Renewable Resources: Natural resources and associated problems.

- (a) Forest Resources: Use and over-exploitation, deforestation, case studies. Timber extraction, mining, dams and their effects on forests and tribal people.
- (b) Water Resources- Use and over-utilization of surface and ground water, floods, drought, conflicts over water, dams-benefits and problems.
- (c) Mineral Resources- Use and exploitation, environmental effects of extracting and using mineral resources, case studies.
- (d) Food Resources- World Food Problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity, case studies.
- (e) Energy Resources- Growing energy needs, renewable and non-renewable energy sources, use of alternate energy sources. Case studies.
- (f) Land Resources- Land as a resource, land, degradation, man induced landslides, soil erosion and desertification.

Role of an individual in conservation of natural resources. Equitable use of resources for sustainable lifestyle.

### UNIT II

Ecosystem-Concept of an ecosystem. Structure and function of an ecosystem. Producers, consumers and 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 ecosystem-

- a. Forest Ecosystem
- b. Grassland Ecosystem
- c. Desert Ecosystem
- d. Aquatic Ecosystems(ponds, streams, lakes, rivers, oceans, estuaries)

Field Work. Visit to a local area to document Environment assets-river/forest/grassland/hill/mountain. Visit to a local polluted site- Urban /Rural Industrial/Agricultural. Study of common plants, insects and birds. Study of simple ecosystems-pond, river, hill, slopes etc. (Field work equal to 5 lecture hours).

### UNIT III

Biodiversity and its conservation. Introduction, Definition: genetic, species and ecosystem diversity. Biogeographical classification of India. Value of biodiversity: consumptive use, productive use, social, ethical, aesthetic and option values. Biodiversity of global, National and local levels. India as a mega-diversity nation Hot spots of Biodiversity. Threats to biodiversity: Habitat loss, poaching of wild life, man-wildlife conflicts. Endangered and endemic species of India. Conservation of Biodiversity- In situ and Ex-Situ conservation of biodiversity.

Environmental Pollution Definition. Cause, effects and control measures of- (a) Air Pollution (b) Water Pollution (c) Soil Pollution (d) Marine Pollution (e) Noise Pollution (f) Thermal Pollution (g) Nuclear Hazards Solid waste management- cause, effects and control measures of urban and industrial wastes. Role of an individual in prevention of pollution. Pollution case studies. Disaster management: floods, earthquake, cyclone and landslides.

### UNIT IV

Social Issues and the Environment. From unsustainable to sustainable development. Urban problems related to energy. Water conservation, rain water harvesting, watershed 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.

Wasteland Reclamation. Consumerism and waste products. Environment Protection Act. Air (Prevention and Control of Pollution) Act. Water (Prevention and Control of Pollution) Act. Wildlife Protection Act. Forest Conservation Act. Issues involved in enforcement of environmental legislation. Public Awareness. Human population and the Environment. Population growth, variation among nations. Population explosion-Family Welfare Programme. Environment and human health. Human rights. Value Education. HIV/AIDS, Women and Child Welfare. Role of Information Technology in Environment and Human Health. Case Studies.

**Text Books**

1. Environmental Studies- Deswal and Deswal. Dhanpat Rai & Co.
2. Environmental Science & Engineering Anandan, P. and Kumaravelan, R. 2009. Scitech Publications (India) Pvt. Ltd., India
3. Environmental Studies. Daniels Ranjit R. J. and Krishnaswamy. 2013. Wiley India.
4. Environmental Science- Botkin and Keller. 2012. Wiley, India

GEETA ENGINEERING COLLEGE