BACHELOR OF COMPUTER APPLICATIONS (BCA)

CHOICE BASED CREDIT SYSTEM (CBCS)

Curriculum and Syllabus

Regular (2019 – 2020)

Enable | Enlighten | Enrich
KARPAGAM
ACADEMY OF HIGHER EDUCATION
(Deemed to be University)
(Established Under Section 3 of UGC Act, 1956)

DEPARTMENT OF COMPUTER APPLICATIONS FACULTY OF ARTS, SCIENCE AND HUMANITIES

KARPAGAM ACADEMY OF HIGHER EDUCATION

(Deemed to be University)

(Established Under Section 3 of UGC Act, 1956)

Eachanari (Post), Coimbatore – 641 021.
Tamilnadu, India
Phone No. 0422-2980011 - 15
Fax No: 0422-2980022-23
E mail ID: info@karpagam.com
Web: www.kahedu.edu.in

KARPAGAM ACADEMY OF HIGHER EDUCATION

(Deemed to be University) (Established Under Section 3 of UGC Act, 1956)

FACULTY OF ARTS, SCIENCE AND HUMANITIES UNDER – GRADUATE PROGRAMMES

(REGULAR PROGRAMME)

SYLLABUS

(2019)

CHOICE BASED CREDIT SYSTEM (CBCS)

Eachanari (Post), Coimbatore – 641 021.
Tamilnadu, India
Phone No. 0422-2980011 - 15
Fax No: 0422-2980022-23
E mail ID: info@karpagam.com
Web: www.kahedu.edu.in

கற்பகம் உயர்கல்வி கலைக்கழகம்
தமிழ்த்துறை
பகுதி - I தமிழ்ப் பாடத்திட்டம் (2019 - 2020)
முதல்பருவம்
(இளநிலை அறிவியல் பட்ட வகுப்புகளுக்குரியது)
(For I-UG Science Degree Classes) 19LSU101

பாடத்திட்டப் பொதுநோக்கம்

- கற்றல் வழி சிந்தனைத் திறனையும், கருத்து வெளிப்பாட்டுத் திறனையும், மேம்படுத்துதல்.
- ஆய்வுநோக்கை மேம்படுத்துதல்.
- இலக்கியங்கள் உணர்த்தும் வாழ்வின் நுட்பமான பகுதிகளை உணர்த்துதல்.
- மனித மனத்தினைப் பக்குவப்படுத்துதலில் இலக்கியம் தரும் பங்கினை உணர்த்துதல்.
- வளர்ந்து வரும் சமூகத்தில் அறஉணர்வு, பண்பாடு போன்றவை குறித்து அறிவூட்டல்.
- அரசுத் தேர்வுகளுக்கு மாணவர்களை ஆயத்தமாக்குதல்.

பாடத்திட்டப் பயன் விளைவு

1. இந்திய குடியுரிமைப் பணி முதலான போட்டித் தேர்வுகளில், விருப்பப் பாடமாக இடம்பெறுகின்ற, 'தமிழ் இலக்கிய வரலாறு' குறித்த முழுமையான அறிமுகம் பெற்றிருத்தல்.
2. கல்வெட்டியல், ஓலைச்சுவடியியல் மற்றும் தொல்லியல் சார்ந்த ஆவணத் தேடலுக்குரிய ஆய்வுமனப்பான்மையுடன், இலக்கியங்களை அணுகுதல்.
3. தமிழின் வளர்ச்சித் துறையாகிய, 'அறிவியல் தமிழ்' ; 'இணைய தமிழ்' குறித்த பன்நோக்கு அணுகுமுறையிலான ஆய்வுச் சிந்தனை மேம்பாடு.
4. வேலைவாய்ப்புக்குரிய சுயதிறன் மேம்பாட்டுடன், படைப்பாக்கத்திறன் மேம்பாடும் பெற்றிருத்தல் .
5. சமுதாய மற்றும் வாழ்வியல் மதிப்புகளைப் பேணுவதற்குக் கருவியாக இலக்கியங்களை நாடுகின்ற மனப்பான்மை வளர்ச்சி.
6. மொழிபெயப்புத் துறைசார்ந்த வேலைவாய்புத் திறன் பெற்றிருத்தல்

Semester – I

பகுதி - I தமிழ்ப் பாடத்திட்டம் (2019-2020)

பகுதி– I, தமிழ் முதல் பருவம் 19LSU101 :

தமிழ் முதல் தாள் 4-H,4-C

(இளநிலை அறிவியல் பட்ட வகுப்புகளுக்குரியது)

(For I-UG Science Degree Classes)

அலகு – I : இக்கால இலக்கியம்: (10 மணிநேரம்)

1. மகாகவி பாரதியார் – பரம்பொருள் வாழ்த்து .
2. புரட்சிக்கவிஞன் பாரதிதாசன் – தமிழின் இனிமை.
3. கவிமணி தேசிக விநாயகம் பிள்ளை – கோவில் வழிபாடு.
4. கவிக்கோ. அப்துல்ரகுமான் – பாருக்குள்ளே நல்ல நாடு.
5. சிற்பி பாலசுப்பிரமணியன் – காலம்.
6. கவிஞர் தாமரை – தொலைந்து போனேன்.

அலகு – II : அற இலக்கியம்: (8 மணிநேரம்)

1. ஔவையார் – கொன்றை வேந்தன் (1- 50 பாடல்கள்)

   அன்னையும் பிதாவும் – புலையும் கொலையும் களவும் தவிர்

2. வேதநாயகம்பிள்ளை - நீதிநூல் – (5 பாடல்கள்)

   சின்னவோர் பொருள், கடவுளை வருந்தி, எப்புவிகளும், வைத்தவர், ஈன்றவர்

3. திருவள்ளுவர்- திருக்குறள்– பண்புடைமை, வினைத்திட்பம்

4. முன்றுறையரையனார் – பழமொழி நானூறு 5 பாடல்கள்

உணற்கு இனிய, பரந்த திறலாரை, நெடியது காண்கிலாய், இனி யாரும், உரைசான்ற

அலகு – III : சிற்றிலக்கியம்: (8 மணிநேரம்)

1. முக்கூடற் பள்ளு- 2 பாடல்கள் - சித்திரக் காலிவாலான் (நெல்வகைகள்)

   குற்றாலத் திரிகூட மால்வரை (மீன் வகைகள்)

2. நந்தி கலம்பகம்- 5 பாடல்கள்- என்னையே புகழ்ந்தேன், பதிதொறு புயல்பொழி,

   இந்தப்புவியில், அடிவிளக்கும் துகில், வானுறுமதியை

3. மதுரைச் சொக்கநாதர் தமிழ்விடு தூது –தமிழின் சிறப்பு

   பாடியருள பத்துப்பாட்டும்-விளம்பக்கேள்.

அலகு – IV : சிறுகதை: (8 மணிநேரம்)

1. மகாமசானம் – புதுமைப்பித்தன்
2. அப்பாவின் வேஷ்டி – பிரபஞ்சன்
3. அந்நியர்கள் – ஆர். சூடாமணி
4. இந்நாட்டுமன்னர் – நாஞ்சில்நாடன்

அலகு – V : மொழிப்பயிற்சி: (6 மணிநேரம்)

1. பொருத்தமான தமிழ்ச் சொற்களைப் பயன்படுத்துதல்
2. செய்யுள் பொருளுணர் திறன்
3. மொழிபெயர்ப்புப் பயிற்சிகள்
4. கடிதங்கள் மற்றும் விண்ணப்பங்கள் எழுதுதல்

பாட நூல்: கற்பகச்சோலை – தமிழ் ஏடு.

கற்பகம் உயர்கல்வி கலைக்கழகத் தமிழ்த்துறை வெளியீடு.

Semester – I

19CAU101 PROGRAMMING FUNDAMENTALS USING C / C++ 5H – 5C

Instruction Hours / week: L: 5 T: 0 P: 0 **Marks:** Internal : **40** External : **60** Total: **100**

End Semester Exam : 3 Hours

Course Objectives

Enable the student

- To impart adequate knowledge on the need of programming languages and problem solving techniques.
- To develop programming skills using the fundamentals and basics of C Language.
- To enable effective usage of arrays, structures, functions, pointers and to implement the memory management concepts.
- To teach the issues in file organization and the usage of file systems.
- To learn the characteristics of an object-oriented programming language: data abstraction and information hiding, inheritance, and dynamic binding of the messages to the methods.
- Use the basic object-oriented design principles in computer problem solving.

Course Outcomes (COs)

Upon successful completion of the course, the student will be able to:

1. Obtain the knowledge about the number systems this will be very useful for bitwise operations.
2. Develop programs using the basic elements like control statements, Arrays and Strings .
3. understand about the dynamic memory allocation using pointers which is essential for utilizing memory
4. Understand about the code reusability with the help of user defined functions.
5. Learn the basics of file handling mechanism that is essential for understanding the concepts in database management systems.
6. Understand the uses of preprocessors and various header file directives.

Unit I - INTRODUCTION TO C AND C++

History of C and C++, Overview of Procedural Programming and Object-Orientation Programming, Using main() function, Compiling and Executing Simple Programs in C++.

Data Types, Variables, Constants, Operators and Basic I/O:

Declaring, Defining and Initializing Variables, Scope of Variables, Using Named Constants, Keywords, Data Types, Casting of Data Types, Operators (Arithmetic, Logical and Bitwise), Using Comments in programs, Character I/O (getc, getchar, putc, putcharetc), Formatted and Console I/O (printf(), scanf(), cin, cout), Using Basic Header Files (stdio.h, iostream.h, conio.hetc).

Expressions, Conditional Statements and Iterative Statements:

Simple Expressions in C++ (including Unary Operator Expressions, Binary Operator Expressions), Understanding Operators Precedence in Expressions, Conditional Statements (if construct, switch-case construct), Understanding syntax and utility of Iterative Statements (while, do-while, and for loops), Use of break and continue in Loops, Using Nested Statements (Conditional as well as Iterative)

Unit II - FUNCTIONS AND ARRAYS

Utility of functions, Call by Value, Call by Reference, Functions returning value, Void functions, Inline Functions, Return data type of functions, Functions parameters, Differentiating between Declaration and Definition of Functions, Command Line Arguments/Parameters in Functions, Functions with variable number of Arguments.
Creating and Using One Dimensional Arrays (Declaring and Defining an Array, Initializing an Array, Accessing individual elements in an Array, Manipulating array elements using loops), Use Various types of arrays (integer, float and character arrays / Strings) Two-dimensional Arrays (Declaring, Defining and Initializing Two Dimensional Array, Working with Rows and Columns), Introduction to Multi-dimensional arrays.

Unit III - DERIVED DATA TYPES (STRUCTURES AND UNIONS)

Understanding utility of structures and unions, Declaring, initializing and using simple structures and unions, Manipulating individual members of structures and unions, Array of Structures, Individual data members as structures, Passing and returning structures from functions, Structure with union as members, Union with structures as members

Pointers and References in C++:

Understanding a Pointer Variable, Simple use of Pointers (Declaring and Dereferencing Pointers to simple variables), Pointers to Pointers, Pointers to structures, Problems with Pointers, Passing pointers as function arguments, Returning a pointer from a function, using arrays as pointers, Passing arrays to functions. Pointers vs. References, Declaring and initializing references, using references as function arguments and function return values

Unit IV - MEMORY ALLOCATION IN C++

Differentiating between static and dynamic memory allocation, use of malloc, calloc and free functions, use of new and delete operators, storage of variables in static and dynamic memory allocation.

File I/O, Preprocessor Directives:

Opening and closing a file (use of fstream header file, ifstream, ofstream and fstream classes), Reading and writing Text Files, Using put(), get(), read() and write() functions, Random access in files, Understanding the Preprocessor Directives (#include, #define, #error, #if, #else, #elif, #endif, #ifdef, #ifndef and #undef), Macros.

Unit V - USING CLASSES IN C++

Principles of Object-Oriented Programming, Defining & Using Classes, Class Constructors, Constructor Overloading, Function overloading in classes, Class Variables &Functions, Objects as parameters, Specifying the Protected and Private Access, Copy Constructors, Overview of Template classes and their use.

Overview of Function Overloading and Operator Overloading:

Need of Overloading functions and operators, Overloading functions by number and type of arguments, Looking at an operator as a function call, Overloading Operators (including assignment operators, unary operators).

Inheritance, Polymorphism and Exception Handling:

Introduction to Inheritance (Multi-Level Inheritance, Multiple Inheritance), Polymorphism (Virtual Functions, Pure Virtual Functions), Basics Exceptional Handling (using catch and throw, multiple catch statements), Catching all exceptions, Restricting exceptions, Rethrowing exceptions.

SUGGESTED READINGS

1. Herbtz Schildt. (2003). C++: The Complete Reference (4th ed.) McGraw Hill, New Delhi.
2. Bjarne Stroustrup. (2013). The C++ Programming Language(4th ed.). Addison-Wesley, New Delhi.
3. Bjarne Stroustroup. (2014). Programming, Principles and Practice using C++(2nd edAddison-Wesley, New Delhi.
4. Balaguruswamy, E. (2008). Object Oriented Programming with C++. Tata McGraw-Hill Education, New Delhi.
5. Paul Deitel., & Harvey Deitel. (2011). C++ How to Program (8th ed.). Prentice Hall, New Delhi.
6. John, R. Hubbard. (2000). Programming with C++- (2nd ed.). Schaum's Series.
7. Andrew Koeni., Barbara, E. Moo. (2000). Accelerated C++. Addison-Wesley.
8. Scott Meyers. (2005). Effective C++ (3rd ed.).Addison-Wesley,.
9. Harry, H. Chaudhary. (2014). Head First C++ Programming: The Definitive Beginner's Guide. LLC USA: First Create space Inc, O-D Publishing.
10. Walter Savitch.(2007) Problem Solving with C++, Pearson Education,.
11. Stanley, B. Lippman., Josee Lajoie., & Barbara, E. Moo. (2012). C++ Primer, 5th ed.). Addison-Wesley

WEB SITES

1. http://www.cs.cf.ac.uk/Dave/C/CE.html
2. http://www2.its.strath.ac.uk/courses/c/
3. http://www.iu.hio.no/~mark/CTutorial/CTutorial.html
4. http://www.cplusplus.com/doc/tutorial/
5. www.cplusplus.com/
6. www.cppreference.com/

Semester – I

19CAU102 COMPUTER SYSTEM ARCHITECTURE 4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Internal : **40** External : **60** Total: **100**
End Semester Exam : 3 Hours

Course Objectives
Enable the student

- To discuss the basic concepts of Logic gates, multiplexers and de multiplexers.
- To enable the students to gain knowledge on the architecture of modern computer.
- To understand how computer stores positive and negative numbers and to perform arithmetic operation of positive and negative numbers.
- Explain different types of addressing modes.
- To learn CISC and RISC architecture.
- To summarize the types of memory and I/O channels.

Course Outcomes (COs)
Upon successful completion of the course, the student will be able to:

1. Students will acquire a basic knowledge about computer system architecture, arithmetic, digital circuits and the low - level programming skills.
2. Define different number systems, binary addition and subtraction, 2's complement representation and operations with this representation.
3. Understand the inner workings and performance capabilities of microprocessors.
4. Understand the concept of registers and Instruction set.
5. Know the addressing modes and instruction format of various microprocessors.
6. Understand memory and I/O channels.

Unit-I - INTRODUCTION

Logic gates, Boolean algebra, circuit simplification, combinational circuits: Adders and Subtractors –Multiplexers and De multiplexers – Encoders and Decoders- sequential circuits: Flip Flop's, registers, counters and memory units.

Unit-II - DATA REPRESENTATION AND BASIC COMPUTER ARITHMETIC

Number systems, complements, fixed and floating point representation, character representation, addition, subtraction, magnitude comparison, multiplication and division algorithms for integers

Unit-III - BASIC COMPUTER ORGANIZATION AND DESIGN

Computer registers, bus system, instruction set, timing and control, instruction cycle, memory reference, input-output and interrupt, Interconnection Structures, Bus Interconnection design of basic computer.

Unit-IV - CENTRAL PROCESSING UNIT

Register organization, arithmetic and logical micro-operations, stack organization, micro programmed control. Instruction formats, addressing modes, instruction codes, machine language, assembly language, input output programming, RISC, CISC architectures, pipelining and parallel architecture.

Unit-V - MEMORY AND INPUT-OUTPUT ORGANIZATION

Cache memory, Associative memory, mapping Input / Output: External Devices, I/O Modules, Programmed I/O, Interrupt-Driven I/O, Direct Memory Access, I/O Channels.

SUGGESTED READINGS

1.M.Mano. (1992). Computer System Architecture. Pearson Education.
2. Dos Reis, A. J. (2009). Assembly Language and Computer Architecture using C++ and JAVA. Course Technology
3. Stallings, W. (2010). Computer Organization and Architecture Designing for Performance (8th ed.) Prentice Hall of India, New Delhi.
4. Mano, M.M. (2013). Digital Design. Pearson Education Asia, New Delhi.
5. Carl Hamacher. (2012). Computer Organization (5th ed.). McGrawHill, New Delhi.

Semester – I

19CAU103 COMPUTER FUNDAMENTALS 4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Internal: **40** External : **60** Total: **100**
End Semester Exam: 3 Hours

Course Objectives
Enable the student

- To identify types of computers, how they process information and how individual computers interact with other computing systems and devices.
- To identify the function of computer hardware components.
- To identify the factors that goes into an individual or organizational decision on how to purchase computer equipment.
- To identify how to maintain computer equipment and solve common problems relating to computer hardware.
- To identify how software and hardware work together to perform computing tasks and how software is developed and upgraded.
- To identify different types of software, general concepts relating to software categories, and the tasks to which each type of software is most suited or not suited.

Course Outcomes (COs)
Upon successful completion of the course, the student will be able to:

1. Understand the meaning and basic components of a computer system,
2. Define and distinguish Hardware and Software components of computer system,
3. Explain and identify different computing machines during the evolution of computer system,
4. Explain the functions of a computer,
5. Identify the various input and output units and explain their purposes
6. Understand the role of CPU and its components

Unit-I – INTRODUCTION

Introduction to computer system, uses, types. **Data Representation:** Number systems and character representation, binary arithmetic. **Human Computer Interface:** Types of software, Operating system as user interface, utility programs.

Unit-II – DEVICES

Input and output devices (with connections and practical demo), keyboard, mouse, joystick, scanner, OCR, OMR, bar code reader, web camera, monitor, printer, plotter.

Unit-III – MEMORY

Primary, secondary, auxiliary memory, RAM, ROM, cache memory, hard disks, optical disks.

Unit-IV - COMPUTER ORGANISATION AND ARCHITECTURE

C.P.U., registers, system bus, main memory unit, cache memory, Inside a computer, SMPS, Motherboard, Ports and Interfaces, expansion cards, ribbon cables, memory chips, processors.

Unit-V - OVERVIEW OF EMERGING TECHNOLOGIES

Bluetooth, cloud computing, big data, data mining, mobile computing and embedded systems.

SUGGESTED READINGS

1. Goel, A. (2010). Computer Fundamentals. Pearson Education, New Delhi.
2. Aksoy, P., & DeNardis, L. (2006).Introduction to Information Technology. Cengage Learning , New Delhi.
3. Sinha, P. K., & Sinha, P. (2007). Fundamentals of Computers. BPB Publishers, New Delhi.

Semester – I

19CAU111 PROGRAMMING FUNDAMENTALS USING C / C++ - PRACTICAL **4H – 2C**

Instruction Hours / week: L: 0 T: 0 P: 4 **Marks:** Internal : **40** External : **60** Total: **100**
End Semester Exam : 3 Hours

Course Objectives

Enable the student

- To impart adequate knowledge on the need of programming languages and problem solving techniques.
- To develop programming skills using the fundamentals and basics of C Language.
- To enable effective usage of arrays, structures, functions, pointers and to implement the memory management concepts.
- To study the concepts of Call by value and call be reference.
- To learn file organization and the usage of file systems.
- To learn the characteristics of an object-oriented programming language: data abstraction and information hiding, inheritance, and dynamic binding of the messages to the methods.

Course Outcomes (COs)

Upon successful completion of the course, the student will be able to:

1. Develop programs using the basic elements like control statements, Arrays and Strings .
2. Solve the memory access problems by using pointers
3. Learn the basics of file handling mechanism that is essential for understanding the concepts in database management systems.
4. Understand the uses of preprocessors and various header file directives.
5. Use the characteristics of an object-oriented programming language in a program.
6. Implement File handling mechanisms.

List of Programs

1. WAP to print the sum and product of digits of an integer.

2. WAP to reverse a number.

3. WAP to compute the sum of the first n terms of the following series S = 1+1/2+1/3+1/4+……

4. WAP to compute the sum of the first n terms of the following series S =1-2+3-4+5…….

5. Write a function that checks whether a given string is Palindrome or not. Use this function to find whether the string entered by user is Palindrome or not.

6. Write a function to find whether a given no. is prime or not. Use the same to generate the prime numbers less than 100.

7.WAP to compute the factors of a given number.

8. Write a macro that swaps two numbers. WAP to use it.

9. WAP to print a triangle of stars as follows (take number of lines from user):

```
*
***
*****
*******
*********
```

10. WAP to perform following actions on an array entered by the user:

i) Print the even-valued elements

ii) Print the odd-valued elements

iii) Calculate and print the sum and average of the elements of array

iv) Print the maximum and minimum element of array

v) Remove the duplicates from the array

vi) Print the array in reverse order

The program should present a menu to the user and ask for one of the options. The menu should also include options to re-enter array and to quit the program.

11. WAP that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments.

12. Write a program that swaps two numbers using pointers.

13. Write a program in which a function is passed address of two variables and then alter its contents.

14. Write a program which takes the radius of a circle as input from the user, passes it to another function that computes the area and the circumference of the circle and displays the value of area and circumference from the main() function.

15. Write a program to find sum of n elements entered by the user. To write this program, allocate memory dynamically using malloc() / calloc() functions or new operator.

16. Write a menu driven program to perform following operations on strings:

a) Show address of each character in string

b) Concatenate two strings without using strcat function.

c) Concatenate two strings using strcat function.

d) Compare two strings

e) Calculate length of the string (use pointers)

f) Convert all lowercase characters to uppercase

g) Convert all uppercase characters to lowercase

h) Calculate number of vowels

i) Reverse the string

17. Given two ordered arrays of integers, write a program to merge the two-arrays to get an ordered array.

18. WAP to display Fibonacci series (i)using recursion, (ii) using iteration

19. WAP to calculate Factorial of a number (i)using recursion, (ii) using iteration

20. WAP to calculate GCD of two numbers (i) with recursion (ii) without recursion.

21. Create Matrix class using templates. Write a menu-driven program to perform following Matrix operations (2-D array implementation):
a) Sum b) Difference c) Product d) Transpose

22. Create the Person class. Create some objects of this class (by taking information from the user). Inherit the class Person to create two classes Teacher and Student class. Maintain the respective information in the classes and create, display and delete objects of these two classes (Use Runtime Polymorphism).

23. Create a class Triangle. Include overloaded functions for calculating area. Overload assignment operator and equality operator.

24. Create a class Box containing length, breath and height. Include following methods in it:
a) Calculate surface Area
b) Calculate Volume
c) Increment, Overload ++ operator (both prefix & postfix)
d) Decrement, Overload -- operator (both prefix & postfix)
e) Overload operator == (to check equality of two boxes), as a friend function
f) Overload Assignment operator
g) Check if it is a Cube or cuboid

Write a program which takes input from the user for length, breath and height to test the above class.

25. Create a structure Student containing fields for Roll No., Name, Class, Year and Total Marks. Create 10 students and store them in a file.

26. Write a program to retrieve the student information from file created in previous question and print it in following format:

Roll No. Name Marks

27. Copy the contents of one text file to another file, after removing all whitespaces.

28. Write a function that reverses the elements of an array in place. The function must accept only one pointer value and return void.

29. Write a program that will read 10 integers from user and store them in an array. Implement array using pointers. The program will print the array elements in ascending and descending order.

SUGGESTED READINGS

1. Bjarne Stroustrup. (2013). The C++ Programming Language(4th ed.). Addison-Wesley, New Delhi.
2. Bjarne Stroustroup. (2014). Programming, Principles and Practice using C++(2nd edAddison-Wesley, New Delhi.
3. Balaguruswamy, E. (2008). Object Oriented Programming with C++. Tata McGraw-Hill Education, New Delhi.
4. Paul Deitel., & Harvey Deitel. (2011). C++ How to Program (8th ed.). Prentice Hall, New Delhi.
5. Harry, H. Chaudhary. (2014). Head First C++ Programming: The Definitive Beginner's Guide. LLC USA: First Create space Inc, O-D Publishing.

WEB SITES

1. http://www2.its.strath.ac.uk/courses/c/
2. http://www.iu.hio.no/~mark/CTutorial/CTutorial.html
3. http://www.cplusplus.com/doc/tutorial/
4. www.cplusplus.com/
5. www.cppreference.com/

Semester – I

19CAU112 COMPUTER SYSTEM ARCHITECTURE - PRACTICAL **3H – 2C**

Instruction Hours / week: L: 0 T: 0 P: 3 **Marks:** Internal : **40** External : **60** Total: **100**
End Semester Exam : 3 Hours

Course Objectives

Enable the student

- To discuss the basic concepts of Logic gates, multiplexers and de multiplexers.
- To enable the students to gain knowledge on the architecture of modern computer.
- To understand how computer stores positive and negative numbers and to perform arithmetic operation of positive and negative numbers.
- Explain different types of addressing modes.
- To learn CISC and RISC architecture.
- To summarize the types of memory and I/O channels.

Course Outcomes (COs)
Upon successful completion of the course, the student will be able to:

1. Students will acquire a basic knowledge about computer system architecture, arithmetic, digital circuits and the low - level programming skills.
2. Understand the inner workings and performance capabilities of advanced microprocessors.
3. analyze the use of encoder and decoder.
4. Understand the working of half adder and full adder.
5. Design and analyze memory, I/O devices and cache structures for processor.
6. Evaluate the performance of computer systems

List of Experiments (Any 8 Experiments)

1. Verification of Logic Gates
2. Code converters
3. Realization of Multiplexer using basic gates
4. Encoder and Decoder
5. Realization Half and Full adders
6. Realization of Subtractor
7. Realization of Parity generator
8. Flip-Flop Circuits
9. Digital to analog Converters
10. Demonstrate a Basic Arithmetic Computing operations

SUGGESTED READINGS

1.M.Mano. (1992). Computer System Architecture. Pearson Education.
2. Dos Reis, A. J. (2009). Assembly Language and Computer Architecture using C++ and JAVA. Course Technology
3. Stallings, W. (2010). Computer Organization and Architecture Designing for Performance (8th ed.) Prentice Hall of India, New Delhi.
4. Mano, M.M. (2013). Digital Design. Pearson Education Asia, New Delhi.
5. Carl Hamacher. (2012). Computer Organization (5th ed.). McGrawHill, New Delhi.

Semester – I

19CAU113 COMPUTER FUNDAMENTALS - PRACTICAL 3H – 2C

Instruction Hours / week: L: 0 T: 0 P: 3 Marks: Internal : **40** External : **60** Total: **100**
End Semester Exam : 3 Hours

Course Objectives
Enable the student

- Create a document in Microsoft Word with formatting options.
- Add a graphic to a document.
- Write functions in Microsoft Excel to perform basic calculations and to convert number to text and text to number.
- Indicate the names and functions of the Excel interface components.
- Construct formulas, including the use of built-in functions, and relative and absolute references.
- Create and modify charts.

Course Outcomes (COs)
Upon successful completion of the course, the student will be able to:

1. Modify text using various formatting options from the editing tools under the Home tab
2. Demonstrate the mechanics and uses of Word tables to organize and present data.
3. Demonstrate working knowledge of using Word's themes and clip art to create a variety of visual effects.
4. Create and design a spreadsheet for general office use.
5. Demonstrate the basic mechanics and navigation of an Excel spreadsheet.
6. Demonstrate formatting techniques and presentation styles.

Demonstrate the use of basic functions and formulas

Practical exercises based on MS Office/ Open Office tools using document preparation and spreadsheet handling packages.

MS Word

1. Prepare **a grocery list** having four columns (Serial number, The name of the product, quantity and price) for the month of April, 06.

• Font specifications for Title (Grocery List): 14-point Arial font in bold and italics.
• The headings of the columns should be in 12-point and bold.
• The rest of the document should be in 10-point Times New Roman.
• Leave a gap of 12-points after the title.

2. Create a **telephone directory**.

• The heading should be 16-point Arial Font in bold

• The rest of the document should use 10-point font size
• Other headings should use 10-point Courier New Font.
• The footer should show the page number as well as the date last updated.

3. Design a **time-table form** for your college.

- The first line should mention the name of the college in 16-point Arial Font and should be bold.
- The second line should give the course name/teacher‘s name and the department in 14-point Arial.
- Leave a gap of 12-points.
- The rest of the document should use 10-point Times New Roman font.
- The footer should contain your specifications as the designer and date of creation.

4. BPB Publications plans to release a new book designed as per your syllabus. Design the **first page of the book** as per the given specifications.

- The title of the book should appear in bold using 20-point Arial font.
- The name of the author and his qualifications should be in the center of the page in 16-point Arial font.
- At the bottom of the document should be the name of the publisher and address in 16-point Times New Roman.
- The details of the offices of the publisher (only location) should appear in the footer.

5. Create the following one page documents.
   a. Compose a note inviting friends to a get-together at your house, Including a list of things to bring with them.
   b. Design a certificate in landscape orientation with a border around the document.
   c. Design a Garage Sale sign.
   d. Make a sign outlining your rules for your bedroom at home, using a numbered list.

6. Create the following documents:
   (a) A newsletter with a headline and 2 columns in portrait orientation, including at least one image surrounded by text.
   (b) Use a newsletter format to promote upcoming projects or events in your classroom or college.

7. Convert following text to a table, using comma as delimiter
   Type the following as shown (do not bold).
   Color, Style, Item
   Blue, A980, Van
   Red, X023, Car
   Green, YL724, Truck
   Name, Age, Sex
   Bob, 23, M
   Linda, 46, F
   Tom, 29, M

8. Enter the following data into a table given on the next page.

Salesperson	Dolls	Trucks	Puzzles
Kennedy, Sally	1327	1423	1193
White, Pete	1421	3863	2934
Pillar, James	5214	3247	5467
York, George	2190	1278	1928
Banks, Jennifer	1201	2528	1203
Atwater, Kelly	4098	3079	2067
Pillar, James	5214	3247	5467
York, George	2190	1278	1928
Banks, Jennifer	1201	2528	1203
Atwater, Kelly	4098	3079	2067

Add a column Region (values: S, N, N,S,S,S) between the Salesperson and Dolls columns to the given table Sort your table data by Region and within Region by Salesperson in ascending order: In this exercise, you will add a new row to your table, place the word Total at the bottom of the Salesperson column, and sum the Dolls, Trucks, and Puzzles columns.

9. Wrapping of text around the image.

10. Following features of menu option must be covered

FILE Complete menu
EDIT Complete menu
VIEW Complete menu
INSERT Complete menu
FORMAT Complete menu
TABLE Complete menu
WINDOW Complete menu
HELP Complete menu
TOOLS All options except Online collaboration, Tools on Macro, Templates

MS Excel

1. Enter the Following data in Excel Sheet

REGIONAL SALES PROJECTION						
State	**Qtr1**	**Qtr2**	**Qtr3**	**QTR4**	**Qtr Total**	**Rate Amount**
Delhi	2020	2400	2100	3000	15	
Punjab	1100	1300	1500	1400	20	
U.P.	3000	3200	2600	2800	17	
Haryana	1800	2000	2200	2700	15	
Rajasthan	2100	2000	1800	2200	20	

TOTAL
AVERAGE

(a) Apply Formatting as follow:

i. Title in TIMES NEW ROMAN

ii. Font Size - 14

iii. Remaining text - ARIAL, Font Size -10

iv. State names and Qtr. Heading Bold, Italic with Gray Fill Color.

v. Numbers in two decimal places.

vi. Qtr. Heading in center Alignment.

vii. Apply Border to whole data.

(b) Calculate State and Qtr. Total

(c) Calculate Average for each quarter

(d) Calculate Amount = Rate * Total .

2. Given the following worksheet

	A	**B**	**C**	**D**
1	Roll No.	Name	Marks	Grade
2	1001	Sachin	99	
3	1002	Sehwag	65	
4	1003	Rahul	41	
5	1004	Sourav	89	
6	1005	Har Bhajan	56	

Calculate the grade of these students on the basis of following guidelines:

If Marks	Then Grade
>= 80	A+
>= 60 < 80	A
>= 50 < 60	B
< 50	F

3. Given the following worksheet

	A	B	C	D	E	F
1	**Salesman**		**Sales in (Rs.)**			
2	No.	Qtr1	Qtr2	Qtr3	Qtr4	Total	Commission
3	S001	5000	8500	12000	9000		
4	S002	7000	4000	7500	11000		
5	S003	4000	9000	6500	8200		
6	S004	5500	6900	4500	10500		
7	S005	7400	8500	9200	8300		
8	S006	5300	7600	9800	6100		

Calculate the commission earned by the salesmen on the basis of following Candidates:

If Total Sales	**Commission**
< 20000	0% of sales
> 20000 and < 25000	4% of sales
> 25000 and < 30000	5.5% of sales
> 30000 and < 35000	8% of sales
>= 35000	11% of sales

The total sales is sum of sales of all the four quarters.

4. A company XYZ Ltd. pays a monthly salary to its employees which consists of basic salary, allowances & deductions. The details of allowances and deductions are as follows:

Allowances

- HRA Dependent on Basic
  - 30% of Basic if Basic <=1000
  - 25% of Basic if Basic>1000 & Basic<=3000
  - 20% of Basic if Basic >3000
- DA Fixed for all employees, 30% of Basic
- Conveyance Allowance Rs. 50/- if Basic is <=1000

Rs. 75/- if Basic >1000 & Basic<=2000

Rs. 100 if Basic >2000

- Entertainment Allowance NIL if Basic is

<=1000 Rs. 100/- if Basic > 1000

Deductions

- Provident Fund 6% of Basic
- Group Insurance Premium Rs. 40/- if Basic is <=1500
  - Rs. 60/- if Basic > 1500 & Basic<=3000
  - Rs. 80/- if Basic >3000

Calculate the following:

Gross Salary = Basic + HRA + DA + Conveyance + Entertainment

Total deduction = Provident Fund + Group Insurance Premium
Net Salary = Gross Salary – Total Deduction.

5. Create Payment Table for a fixed Principal amount, variable rate of interests and time in the format below:

No. of Installments	**5%**	**6%**	**7%**	**8%**	**9%**
3	XX	XX	XX	XX	XX
4	XX	XX	XX	XX	XX
5	XX	XX	XX	XX	XX
6	XX	XX	XX	XX	XX

6. Use an array formula to calculate Simple Interest for given principal amounts given the rate of Interest and time

Rate of Interest	8%
Time	5 Years
Principal	Simple Interest
1000	?
18000	?
5200	?

7. The following table gives year wise sale figure of five salesmen in Rs.

Salesman	2000	2001	2002	2003
S1	10000	12000	20000	50000
S2	15000	18000	50000	60000
S3	20000	22000	70000	70000
S4	30000	30000	100000	80000
S5	40000	45000	125000	90000

(*a*) Calculate total sale year wise.
(*b*) Calculate the net sale made by each salesman
(*c*) Calculate the maximum sale made by the salesman
(d) Calculate the commission for each salesman under the condition.
(*i*) If total sales >4,00,000 give 5% commission on total sale made by the salesman.
(*ii*) Otherwise give 2% commission.
(*e*) Draw a bar graph representing the sale made by each salesman.
(*f*) Draw a pie graph representing the sale made by salesman in 2000.

8. Enter the following data in Excel Sheet

PERSONAL BUDGET FOR FIRST QUARTER

Monthly Income (Net): 1,475

EXPENSES	**JAN**	**FEB**	**MARCH**	**QUARTER TOTAL**	**QUARTER AVERAGE**
Rent	600.00	600.00	600.00		
Telephone	48.25	43.50	60.00		
Utilities	67.27	110.00	70.00		
Credit Card	200.00	110.00	70.00		
Oil	100.00	150.00	90.00		
AV to Insurance	150.00				
Cable TV	40.75	40.75	40.75		
Monthly Total					

Calculate Quarter total and Quarter average.

(a) Calculate Monthly total.

(b) Surplus = Monthly income - Monthly total.

(c) What would be total surplus if monthly income is 1500.

(d) How much does telephone expense for March differ from quarter average.

(e) Create a 3D column graph for telephone and utilities. (f) Create a pie chart for monthly expenses.

9. Enter the following data in Excel Sheet

TOTAL REVENUE EARNED FOR SAM'S BOOKSTALL

Publisher name	**1997**	**1998**	**1999**	**2000**	**total**
A	Rs.1000.00	Rs.1100.00	Rs.1300.00	Rs.800.00	
B	Rs.1500.00	Rs.700.00	Rs.1000.00	Rs.2000.00	
C	Rs.700.00	Rs.900.00	Rs.1500.00	Rs.600.00	
D	Rs.1200.00	Rs.500.00	Rs.200.00	Rs.1100.00	
E	Rs.800.00	Rs.1000.00	Rs.3000.00	Rs.560.00	

(a) Compute the total revenue earned.

(b) Plot the line chart to compare the revenue of all publisher for 4 years.

(c) Chart Title should be _Total Revenue of sam's Bookstall (1997-2000)'

(d) Give appropriate categories and value axis title.

10. Generate 25 random numbers between 0 & 100 and find their sum, average and count. How many no. are in range 50-60.

SUGGESTED READINGS

1. Bittu Kumar (2015). Microsoft Office 2010. VS Publishers, New Delhi

2. Ramesh Bangia (2015). Learning Microsoft Office 2010, UBS Publishers.
3. Peter Weverka (2010) Office 2010 All-in-One For Dummies, Wiley Publishing Inc.

Semester – I

19AEC101 ENVIRONMENTAL STUDIES 3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**
End Semester Exam: 3 Hours

Course Objectives

Enable the student

- To create the awareness about environmental problems among people.
- To develop an attitude of concern for the environment.
- To motivate public to participate in environment protection and improvement.
- To encourage to learn ecosystems and biodiversity.
- To learn environment pollution and control measures of pollution.
- To create system concepts and methodologies and analyze interactions.

Course Outcomes (COs)

Upon successful completion of the course, the student will be able to:

1. Master core concepts and methods from ecological and physical sciences and their application in environmental problem solving.
2. Master core concepts and methods from economic, political, and social analysis as they pertain to the design and evaluation of environmental policies and institutions.
3. Appreciate the ethical, cross-cultural, and historical context of environmental issues and the links between human and natural systems.
4. Understand the transnational character of environmental problems and ways of addressing them, including interactions across local to global scales.
5. Apply systems concepts and methodologies to analyze and understand interactions between social and environmental processes.
6. Reflect critically about their roles and identities as citizens, consumers and environmental actors in a complex, interconnected world.

Unit I – INTRODUCTION - ENVIRONMENTAL STUDIES & ECOSYSTEMS

Environment Definition, Scope and importance; Ecosystem, Structure and functions of ecosystem. Energy flow, Food chains and food webs, Ecological succession. Classification of ecosystem. Forest ecosystem, Grassland Ecosystem, Desert ecosystem, Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries).

Unit II - NATURAL RESOURCES - RENEWABLE AND NON-RENEWABLE RESOURCES

Natural resources - Renewable and Non – Renewable resources. Land resources and land use change, Land degradation, soil erosion and desertification. Forest resources - Deforestation: Causes and impacts due to mining, dam building on environment, forests, biodiversity and tribal populations. Water resources - Use and over-exploitation of surface and ground water, floods,

droughts, conflicts over water. Use of alternate energy sources, growing energy needs, case studies. Role of an individual in conservation of natural resources. Equitable use of resources for sustainable lifestyles.

Unit III - BIODIVERSITY AND ITS CONSERVATION

Levels of biological diversity - genetic, species and ecosystem diversity. Ecosystem and biodiversity services: Ecological, economic, social, ethical, aesthetic and Informational value. Bio-geographical classification of India. Biodiversity patterns (global, National and local levels). Hot-spots of biodiversity. India as a mega-diversity nation. Endangered and endemic species of India. Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts. Conservation of biodiversity: in-situ and ex-situ conservation of biodiversity.

Unit IV - ENVIRONMENTAL POLLUTION

Definition, causes, effects and control measures of Air pollution, Water pollution, Soil pollution, Noise pollution. Nuclear hazards and human health risks. Solid waste management and control measures of urban and industrial wastes. Role of an individual in prevention of pollution. Case studies.

Unit V - SOCIAL ISSUES AND THE ENVIRONMENT

Concept of sustainability and sustainable development. Water conservation - Rain water harvesting, watershed management. Climate change, global warming, ozone layer depletion, acid rain and its impacts on human communities and agriculture. Environment Laws (Environment Protection Act, Air Act, Water Act, Wildlife Protection Act, Forest Conservation Act). International agreements (Montreal and Kyoto protocols). Resettlement and rehabilitation of project affected persons. Disaster management (floods, earthquake, cyclones and landslides). Environmental Movements (Chipko, Silent valley, Bishnois of Rajasthan). Environmental ethics: Role of Indian and other religions and cultures in environmental conservation. Environmental communication and public awareness, case studies (e.g., CNG vehicles in Delhi). Human population growth: Impacts on environment, human health and welfare.

Suggested Readings

1. Anonymous. 2004. A text book for Environmental Studies, University Grants Commission and Bharat Vidypeeth Institute of Environmental Education Research, New Delhi.
2. Anubha Kaushik., and Kaushik, C.P. 2004. Perspectives in Environmental Studies. New Age International Pvt. Ltd. Publications, New Delhi.
3. Arvind Kumar. 2004. A Textbook of Environmental Science. APH Publishing Corporation, New Delhi.
4. Daniel, B. Botkin., and Edward, A. Keller. 1995. Environmental Science John Wiley and Sons, Inc., New York.
5. Mishra, D.D. 2010. Fundamental Concepts in Environmental Studies. S.Chand & Company Pvt. Ltd., New Delhi.

6. Odum, E.P., Odum, H.T. and Andrews, J. 1971. Fundamentals of Ecology. Philadelphia: Saunders.
7. Rajagopalan, R. 2016. Environmental Studies: From Crisis to Cure, Oxford University Press**.**
8. Sing, J.S., Sing. S.P. and Gupta, S.R. 2014. Ecology, Environmental Science and Conservation. S. Chand & Publishing Company, New Delhi.
9. Singh, M.P., Singh, B.S., and Soma, S. Dey. 2004. Conservation of Biodiversity and Natural Resources. Daya Publishing House, New Delhi.
10. Tripathy. S.N., and Sunakar Panda. (2004). Fundamentals of Environmental Studies (2nd ed.). Vrianda Publications Private Ltd, New Delhi.
11. Verma, P.S., and Agarwal V.K. 2001. Environmental Biology (Principles of Ecology). S. Chand and Company Ltd, New Delhi.
12. Uberoi, N.K. 2005. Environmental Studies. Excel Books Publications, New Delhi.

கற்பகம் உயர்கல்வி கலைக்கழகம்
தமிழ்த்துறை
பகுதி - I தமிழ்ப் பாடத்திட்டம் (2019 - 2020)
இரண்டாம் பருவம்
(இளநிலை அறிவியல் பட்ட வகுப்புகளுக்குரியது)
(For I-UG Science Degree Classes) 19LSU201

பாடத்திட்டப் பொதுநோக்கம்

- கற்றல் வழி சிந்தனைத் திறனையும், கருத்து வெளிப்பாட்டுத் திறனையும், மேம்படுத்துதல்.
- ஆய்வுநோக்கை மேம்படுத்துதல்.
- இலக்கியங்கள் உணர்த்தும் வாழ்வின் நுட்பமான பகுதிகளை உணர்த்துதல்.
- மனித மனத்தினைப் பக்குவப்படுத்துதலில் இலக்கியம் தரும் பங்கினை உணர்த்துதல்.
- வளர்ந்து வரும் சமூகத்தில் அறஉணர்வு, பண்பாடு போன்றவை குறித்து அறிவூட்டல்.
- அரசுத் தேர்வுகளுக்கு மாணவர்களை ஆயத்தமாக்குதல்.

பாடத்திட்டப் பயன் விளைவு

1. இந்திய குடியுரிமைப் பணி முதலான போட்டித் தேர்வுகளில், விருப்பப் பாடமாக இடம்பெறுகின்ற, 'தமிழ் இலக்கிய வரலாறு' குறித்த முழுமையான அறிமுகம் பெற்றிருத்தல்.
2. கல்வெட்டியல், ஓலைச்சுவடியியல் மற்றும் தொல்லியல் சார்ந்த ஆவணத் தேடலுக்குரிய ஆய்வுமனப்பான்மையுடன், இலக்கியங்களை அணுகுதல்.
3. தமிழின் வளர்ச்சித் துறையாகிய, 'அறிவியல் தமிழ்' ; 'இணைய தமிழ்' குறித்த பன்நோக்கு அணுகுமுறையிலான ஆய்வுச் சிந்தனை மேம்பாடு.
4. வேலைவாய்ப்புக்குரிய சுயதிறன் மேம்பாட்டுடன், படைப்பாக்கத்திறன் மேம்பாடும் பெற்றிருத்தல் .
5. சமுதாய மற்றும் வாழ்வியல் மதிப்புகளைப் பேணுவதற்குக் கருவியாக இலக்கியங்களை நாடுகின்ற மனப்பான்மை வளர்ச்சி.
6. மொழிபெயப்புத் துறைசார்ந்த வேலைவாய்புத் திறன் பெற்றிருத்தல்

Semester – II

பகுதி – I, தமிழ் இரண்டாம் பருவம் 19LSU201

தமிழ் இரண்டாம் தாள் 4-H,4-C

(இளநிலை அறிவியல் பட்ட வகுப்புகளுக்குரியது)

(For I-UG Science Degree Classes)

அலகு – I : பக்தி இலக்கியம் (7 மணிநேரம்)

சைவ, வைணவ இலக்கியங்கள் - தோற்றம் ,வளர்ச்சி, வரலாறு.

1. **சைவம் - (15 பாடல்கள்)** பெரியபுராணம் – திருமூலநாயனார் புராணம் -அந்தி இளம்பிறைக் கண்ணி, மற்று அவர்தாம் அணிமா, காவிரி நீர் பெருந்தீர்த்தம், அந்நிலைமைத் தானத்தை, அந்தணர்தம் சாத்தனூர், மற்றுஅதன் தன் உடம்பினை, இவன் உயிர்பெற் றெழில், பாய்த்திய பின் திருமூலராய், வெய்ய சுடர் கதிரவனும், அங்கவளும், பித்து உற்ற மையல் அன்று, இந்த நிலைமையில், ஆவடு தண்துறை, ஊன்உடம்பில், முன்னிய அப்பொருள்.
2. **வைணவம் – ஆண்டாள் நாச்சியார் திருப்பாவை: (11 பாடல்கள்):** மார்கழித்திங்கள், வையத்து வாழ்வீர்காள், ஓங்கி உலகளந்த, ஆழி மழைக்கண்ணா, மாயனை மன்னுவட மதுரை, சிற்றம் சிறுகாலே, ஒருத்தி மகனாய், மாலே மணிவண்ணா, கூடாரை வெல்லும், கறவைகள் பின்சென்று, வங்கக்கடல் கடைந்த.

அலகு – II : சங்க இலக்கியம் (14 மணிநேரம்)

சங்க இலக்கியங்கள் - அறிமுகம்

அ). எட்டுத்தொகை

நற்றிணை: கொண்டல் மாமழை – குறிஞ்சி – தலைவன் - 140

குறுந்தொகை: வாரார் ஆயினும், வரினும் –முல்லை– தலைவி - 110

ஐங்குறுநூறு : மருதம் –தோழி -வேட்கைப்பத்து: வாழிஆதன் வாழி அவனி - 6

பதிற்றுப்பத்து : சிதைந்தது மன்ற - 27

பரிபாடல்: பரிபாடல் திரட்டு-வையை –10: மலைவரை மாலை- கரை நண்ணி 10:1-25, ஆங்க அணிநிலை மாடத்து- நீ கானும் போன்ம் 10: 41-55, இலம்படு புலவர் - ஏத்தினர் தொழுவே 10: 126-130.

கலித்தொகை : பாலைக்கலி- செவிலி – எறித்தரு கதிர்தாங்கி-9

அகநானூறு : அன்னை அறியினும் அறிக – தோழி - நெய்தல் - 110

புறநானூறு : யாதும் ஊரே யாவருங் கேளிர் –பொதுவியல்- 192

ஆ). பத்துப்பாட்டு: நெடுநல்வாடை - கார்காலச் சிறப்பு : வையகம் பனிப்ப -1-70

அலகு - III : காப்பியம் (6 மணிநேரம்)

(அ). சிலப்பதிகாரம்:

மங்கல வாழ்த்துப் பாடல்: (21-29)- நாக நீள் நகரொடு-கண்ணகி என்பாண் மன்னோ .

வழக்குரை காதை, (48-56) - நீர்வார் கண்ணை-புகா ரென்பதியே .

வஞ்சின மாலை: (5-34) - வன்னிமரமும் – பிறந்த பதிப் பிறந்தேன்.

நடுகற் காதை: (207-234) - அருத்திற லரசர் – மன்னவ ரேறென்,.

வாழ்த்துக்காதை: (9) - என்னேயிஃ தென்னே – மீவிசும்பிற் றோன்றுமால்.

(ஆ). மணிமேகலை:

பசியின் கொடுமை: பாத்திரம் பெற்ற காதை:

'போதி நீழல்' - 'பெருகியதன்றோ' , 'ஆற்றுநர்க்களிப்போர்' - 'நல்லறம் கண்டனை' (73-98).

சிறைக்கோட்டம் அறக்கோட்டமாக்கிய காதை: மாவண் கிள்ளிக்கு காவலன் உரைத்தவை:

'பைஞ்சேறு மெழுகாப் பசும்பொன் மண்டபத்து -

அரவோர்க் காக்கினன் அரசாள் வேந்தன்' (116-163).

அலகு – IV : கட்டுரைகள் (8 மணிநேரம்)

1. **தமிழிசை இயக்கம்** – பேரா. க. வெள்ளைவாரணர்
2. **தனிப்பாடல்கள்** - பேரா. வ. சுப. மாணிக்கனார்
3. **கலைச்சிறப்பு** - முத்தமிழ்க் காவலர் கி.ஆ.பெ. விசுவநாதம்
4. **காளத்திவேடனும் கங்கைவேடனும்** –சொல்லின்செல்வர் ரா.பி.சேதுப்பிள்ளை
5. **ஏட்டில் இல்லாத இலக்கியம்** – ஔவை துரைசாமி

அலகு- V : மொழிப்பயிற்சி (5 மணிநேரம்)

(பேச்சுப்பயிற்சி & படைப்பாக்கப் பயிற்சி)

படைப்பிலக்கியப் பயிற்சிகள் (கதை, கவிதை, கட்டுரை, உரைநடை எழுதுதல், நேர்காணல்) மொழிபெயர்ப்பு

பாட நூல்: கற்பகச்சோலை – தமிழ் ஏடு.

கற்பகம் உயர்கல்வி கலைக்கழகத் தமிழ்த்துறை வெளியீடு.

Semester – II

19ENU201 ENGLISH 4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Internal : **40** External : **60** Total: **100**
End Semester Exam : 3 Hours

Course Objectives:

- To train students to acquire proficiency in English.
- To explore different genres of literature and learning grammar.
- To provide aesthetic pleasure through literature.
- To inculcate moral values through literature.
- To develop ethical values.
- To give basic grammar knowledge.

Course Outcome:

1. Develop the knowledge of interpersonal skills.
2. Establish and maintain social relationships.
3. Genres of literature will give moral values of life.
4. Develop communication skills in business environment
5. Communication skills will get developed.
6. Develop to have language competence.

Unit - I : PROSE

1. Morals in the Indian Context - Francis Nicholas Chelliah
2. How Comic Books help us to relive our Childhood - Benoit Peeters
3. Let's Do What India Needs From Us -Dr.A.P.J. Abdul Kalam

Unit - II : POEM

1. The Stolen Boat - William Wordsworth
2. Telephone Conversation- Wole Soyinka
3. A River - A.K. Ramanujan

Unit - III : SHORT STORIES

1. Rapunzel - Brothers Grimm
2. The Ant and The Grasshopper- W. Somerset Maugham
3. The Nightingale and the Rose - Oscar Wilde.

Unit – IV : Drama

1. The Merchant of Venice- Act 4-Scence 1
2. The Death Trap- Saki

Unit – V : Grammar and Composition

GRAMMAR:

1. Tenses
2. Articles
3. Auxiliaries (Primary and Modal)
4. Tag Questions

Composition:

1. Reading to Comprehend
2. Letter Writing
3. Resume Writing
4. General Essay

Prescribed Text:

Reminisce, Published by the Department of English,KarpagamAcademy of Higher Education.

Suggested Reading:

Hewings Martin,1999 Advanced English Grammar, Cambridge University Press

Semester – II

19CAU201 PROGRAMMING IN JAVA 4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Internal : **40** External : **60** Total: **100**
End Semester Exam : 3 Hours

Course Objectives

Enable the Student

- To understand the fundamentals of programming such as variables, conditional and iterative execution, methods, etc.
- To understand fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries, etc.
- To use the Java SDK environment to create, debug and run simple Java programs.
- To use Java in various technologies in different platforms.
- To understand the fundamental of Packages and access modifiers and interface in java.
- To understand the fundamental of Exception Handling and AWT component and AWT classes.

Course Outcomes (COs)

Upon successful completion of the course, the student will be able to:

1. Student will obtain knowledge of the structure and model of the Java programming language.
2. Use looping and decision constructs to solve problems.
3. Develop software in the Java programming language (application)
4. Evaluate user requirements for software functionality required to decide whether the Java programming language can meet user requirements (analysis)
5. propose the use of certain technologies by implementing them in the Java programming language to solve the given problem (synthesis)
6. choose an engineering approach to solving problems, starting from the acquired knowledge of programming and knowledge of operating systems. (evaluation)

Unit I - INTRODUCTION TO JAVA

Java Architecture and Features, Understanding the semantic and syntax differences between C++ and Java, Compiling and Executing a Java Program, Variables, Constants, Keywords Data Types, Operators (Arithmetic, Logical and Bitwise) and Expressions, Comments, Doing Basic Program Output, Decision Making Constructs (conditional statements and loops) and Nesting, Java Methods (Defining, Scope, Passing and Returning Arguments, Type Conversion and Type and Checking, Built-in Java Class Methods)

Unit II - ARRAYS, STRINGS AND I/O

Creating & Using Arrays (One Dimension and Multi-dimensional), Referencing Arrays Dynamically, Java Strings: The Java String class, Creating & Using String Objects, Manipulating Strings, String Immutability & Equality, Passing Strings To & From Methods, String Buffer Classes. Simple I/O using System.out and the Scanner class, Byte and Character streams, Reading/Writing from console and files. **Object-Oriented Programming Overview** Principles of Object-Oriented Programming, Defining & Using Classes, Controlling Access to Class Members, Class Constructors, Method Overloading, Class Variables & Methods, Objects as parameters, final classes, Object class, Garbage Collection.

Unit III - INHERITANCE

Inheritance: (Single Level and Multilevel, Method Overriding, Dynamic Method Dispatch, Abstract Classes), Interfaces and Packages, Extending interfaces and packages, Package and Class Visibility, Using Standard Java Packages (util, lang, io, net), Wrapper Classes, Autoboxing/Unboxing, Enumerations and Metadata.

Unit IV - EXCEPTION HANDLING

Exception types, uncaught exceptions, throw, built-in exceptions, Creating your own exceptions; Multi-threading: The Thread class and Runnable interface, creating single and multiple threads, Thread prioritization, synchronization and communication, suspending/resuming threads.

Unit V - DATABASE CONNECTIVITY AND JAVA APPLETS

Using java.net package, Overview of TCP/IP and Datagram programming. Accessing and manipulating databases using JDBC. Introduction to Applets, Writing Java Applets, Working with Graphics, Incorporating Images & Sounds. Event Handling Mechanisms, Listener Interfaces, Adapter and Inner Classes.

SUGGESTED READINGS

1. Ken Arnold., James Gosling.,& David Homes. (2005). The Java Programming Language (4th ed.).
2. James Gosling., Bill Joy., Guy, L. Steele Jr., Gilad Bracha., & Alex Buckley. (2014).The Java Language Specification, Java SE (8 ed.). Addison Wesley.
3. Joshua Bloch. (2008). Effective Java (2nd ed.). Addison-Wesley.
4. Cay, S. Horstmann,. GaryCornell. (2012).Core Java 2 Volume 1 (9th ed.). . Prentice Hall, New Delhi.
5. Cay, S. Horstmann., Gary Cornell. (2013). Core Java 2 Volume 2 - Advanced Features(9th ed.). Printice Hall, New Delhi.
6. Bruce Eckel. (2002). Thinking in Java (3rd ed.). PHI, New Delhi
7. Balaguruswamy, E. (2009). Programming with Java (4th ed.). McGraw Hill, New Delhi.
8. Paul Deitel., & Harvey Deitel. (2011). Java: How to Program (10th ed.). Prentice Hall, New Delhi.

9. David, J. Eck. (2009). Introduction to Programming Using Java. CreateSpace Independent Publishing Platform, New Delhi.
10. John , R. Hubbard. (2004). Programming with JAVA, Schaum's Series, (2nd ed.).

WEB SITES

1. java.sun.com/docs/books/tutorial/
2. www.java.net/

Semester – II

19CAU202 DISCRETE STRUCTURES 4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Internal : **40** External : **60** Total: **100**
End Semester Exam : 3 Hours

Course Objectives
Enable the student

- To develop logical thinking and its application to computer science
- To write and evaluate a proof or outline the basic structure of and give examples of each proof technique described
- To know the concepts of sets and functions.
- to transform a given problem into a combination of several simpler statements and arrive at a solution to the problem and be able to prove it logically
- To solve problems with the help of tools of mathematical analysis.
- To reason and ability to present a coherent and mathematically accurate argument.

Course Outcomes (COs)
Upon successful completion of the course, the student will be able to:

1. Familiar with elementary algebraic set theory.
2. Demonstrate the ability to write and evaluate a proof or outline the basic structure of and give examples of each proof technique described.
3. Understand the basic principles of sets and operations in sets.
4. Demonstrate an understanding of relations and functions and be able to determine their properties.
5. Demonstrate different traversal methods for trees and graphs.
6. Initiate to knowledge from inference theory.

Unit I

Sets: Introduction, Sets , finite and infinite sets, uncountably infinite sets, functions, relations, properties of binary relations, closure, partial ordering relations.

Unit II

Pigeonhole principle, Permutation and Combination, Mathematical Induction, Principle of inclusion and Exclusion.

Unit III

Recurrences: Recurrence relations, generating functions, linear recurrence relations with constant coefficients and their solution.

Unit IV

Graph Theory: Basic terminology, models and types, multigraphs and weighted graphs, graph representation, graph isomorphism, connectivity, Euler and Hamiltonian Paths and circuits, Planar graphs, graph coloring, trees, basic terminology and properties of trees, introduction to Spanning trees

Unit V

Prepositional Logic: Logical Connectives, Well-formed Formulas, Tautologies, Equivalences,
Inference Theory.

SUGGESTED READINGS

1. Kenneth Rosen.(2011). Discrete Mathematics and Its Applications (7th ed.), McGraw Hill , Company, New Delhi.

2. Tremblay ,J .P., and Manohar R., (2001). Discrete Mathematical Structures with Applications to
Computer Science(1st ed.), McGraw-Hill Book Company, New Delhi.

3. Coremen,T.H.,Leiserson C.E. , & R. L. Rivest.(2009). Introduction to algorithms, (3rd ed.), Prentice Hall on India, New Delhi.

4. Albertson, M.O.,& Hutchinson,J.P., (1988). Discrete Mathematics with Algorithms, John Wiley
Publication, New Delhi.

5. Hein,J.L., (2010). Discrete Structures, Logic, and Computability (3rd ed.), Jones and Bartlett Publishers, New Delhi.
.
6. Hunter,D.J.(2016). Essentials of Discrete Mathematics(3rd ed.). Jones and Bartlett Publishers, New Delhi.

Semester – II

19CAU203 COMPUTER NETWORKS AND INTERNET TECHNOLOGIES **4H – 4C**

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Internal : **40** External : **60** Total: **100**
End Semester Exam : 3 Hours

Course Objectives

Enable the student

- To study the basics of Computer Networks.
- To study and compare various Network architectures and fundamental protocols.
- To learn about networking protocol and OSI model.
- To understand the topologies of networks, layered architecture (OSI and TCP/IP) and protocol suites.
- To understand the principles of creating an effective web page.
- To develop skills in analyzing the usability of a website.

Course Outcomes (COs)

Upon successful completion of the course, the student will be able to:

1. Independently understand basic computer network technology.
2. Understand and explain Data Communications System and its components.
3. Enumerate the layers of the OSI model and TCP/IP.
4. Gain the skills and project-based experience needed for entry into web design and development careers.
5. Develop awareness and appreciation of the many ways that people access the web, and will be able to create standards-based websites that can be accessed by the full spectrum of web access technologies
6. Select and apply markup languages for processing, identifying, and presenting of information in web

Unit I - COMPUTER NETWORKS

Introduction to computer network, data communication, components of data communication, data transmission mode, data communication measurement, LAN, MAN, WAN, wireless LAN, internet, intranet, extranet. **Network Models**: Client/ server network and Peer-to-peer network, OSI, TCP/IP, 8L layers and functionalities.

Unit II - TRANSMISSION MEDIA AND LAN TOPOLOGIES

Introduction, Guided Media: Twisted pair, Coaxial cable, 4L Optical fiber. Unguided media: Microwave, Radio frequency propagation, Satellite. **LAN Topologies**: Ring, bus, star, mesh and tree topologies. Network Devices: NIC, repeaters, hub, bridge, switch, gateway and router.

Unit III- INTERNET TERMS AND APPLICATIONS

Internet Terms: Web page, Home page, website, internet browsers, URL, Hypertext, 2L ISP, Web server, download and upload, online and offline. **Internet Applications:** www, telnet, ftp, e-mail, social networks, search engines, 6L Video Conferencing, e-Commerce, m-Commerce, VOIP, blogs.

Unit IV - INTRODUCTION TO WEB DESIGN

Introduction to hypertext markup language (html) 16L Document type definition, creating web pages, lists, hyperlinks, tables, web forms, inserting images, frames, hosting options and domain name registration. **Customized Features**: Cascading style sheet (css) for text formatting and other manipulations.

Unit V - JAVASCRIPT FUNDAMENTALS

Data types and variables, functions, methods and events, 14L controlling program flow, JavaScript object model, built-in objects and operators.

SUGGESTED READINGS

1. Larry L.Peterson & Bruce S.Davie (2011). Computer Networks A System Approach, Morgan Kaufmann Publishers.
2. Andrew S. Tanenbaum, David J. Wetherall Computer Networks (5th Edition),PHI, 2010
3. B. A. Forouzan, Data Communication and Networking , TMH,2003.
4. D.R. Brooks, An Introduction to HTML and Javascript for Scientists and Engineers, Springer W. Willard,2009
5. HTML A Beginner's Guide, Tata McGraw-Hill Education, 2009.
6. J. A. Ramalho, Learn Advanced HTML 4.0 wit

WEBSITES

1. https://developer.mozilla.org/en-US/docs/Web
2. https://www.w3schools.com
3. http://en.wikipedia.org/wiki/script_language
4. https://css-tricks.com

Semester – II

19CAU211 PROGRAMMING IN JAVA - PRACTICAL 4H – 2C

Instruction Hours / week: L: 0 T: 0 P: 4 Marks: Internal : **40** External : **60** Total: **100**
End Semester Exam : 3 Hours

Course Objectives

Enable the student

- To understand the fundamentals of programming such as variables, conditional and iterative execution, methods, etc.
- To understand fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries, etc.
- To use the Java SDK environment to create, debug and run simple Java programs.
- To use Java in various technologies in different platforms.
- To understand the fundamental of Packages and access modifiers and interface in java.
- To implement the Exception Handling and AWT component and AWT classes.

Course Outcomes (COs)

Upon successful completion of the course, the student will be able to:

1. Student will obtain knowledge of the structure and model of the Java programming language.
2. How to use the Java programming language for various programming technologies (understanding)
3. Develop software in the Java programming language (application)
4. Evaluate user requirements for software functionality required to decide whether the Java programming language can meet user requirements (analysis)
5. propose the use of certain technologies by implementing them in the Java programming language to solve the given problem (synthesis)
6. choose an engineering approach to solving problems, starting from the acquired knowledge of programming and knowledge of operating systems. (evaluation)

List of Programs

1. To find the sum of any number of integers entered as command line arguments
2. To find the factorial of a given number
3. To learn use of single dimensional array by defining the array dynamically.
4. To learn use of lenth in case of a two dimensional array
5. To convert a decimal to binary number
6. To check if a number is prime or not, by taking the number as input from the keyboard
7. To find the sum of any number of integers interactively, i.e., entering every number from the keyboard, whereas the total number of integers is given as a command line argument
8. Write a program that show working of different functions of String and StringBufferclasss like setCharAt(, setLength(), append(), insert(), concat()and equals().
9. Write a program to create a ―distance‖ class with methods where distance is computed in terms of feet and inches, how to create objects of a class and to see the use of this pointer

10. Modify the ―distance‖ class by creating constructor for assigning values (feet and inches) to the distance object. Create another object and assign second object as reference variable to another object reference variable. Further create a third object which is a clone of the first object.
11. Write a program to show that during function overloading, if no matching argument is found, then java will apply automatic type conversions(from lower to higher data type)
12. Write a program to show the difference between public and private access specifiers. The program should also show that primitive data types are passed by value and objects are passed by reference and to learn use of final keyword
13. Write a program to show the use of static functions and to pass variable length arguments in a function.
14. Write a program to demonstrate the concept of boxing and unboxing.
15. Create a multi-file program where in one file a string message is taken as input from the user and the function to display the message on the screen is given in another file (make use of Scanner package in this program).
16. Write a program to create a multilevel package and also creates a reusable class to generate Fibonacci series, where the function to generate fibonacii series is given in a different file belonging to the same package.
17. Write a program that creates illustrates different levels of protection in classes/subclasses belonging to same package or different packages
18. Write a program ―DivideByZero‖ that takes two numbers a and b as input, computes a/b, and invokes Arithmetic Exception to generate a message when the denominator is zero.
19. Write a program to show the use of nested try statements that emphasizes the sequence of checking for catch handler statements.
20. Write a program to create your own exception types to handle situation specific to your application (Hint: Define a subclass of Exception which itself is a subclass of Throwable).
21. Write a program to demonstrate priorities among multiple threads.
22. Write a program to demonstrate multithread communication by implementing synchronization among threads (Hint: you can implement a simple producer and consumer problem).
23. Write a program to create URL object, create a URLConnection using the openConnection() method and then use it examine the different components of the URL and content.
24. Write a program to implement a simple datagram client and server in which a message that is typed into the server window is sent to the client side where it is displayed.
25. Write a program that creates a Banner and then creates a thread to scrolls the message in the banner from left to right across the applet‘s window.
26. Write a program to get the URL/location of code (i.e. java code) and document(i.e. html file).
27. Write a program to demonstrate different mouse handling events like mouseClicked(), mouseEntered(), mouseExited(), mousePressed, mouseReleased() and mouseDragged().
28. Write a program to demonstrate different keyboard handling events.
29. Write a program to generate a window without an applet window using main() function.
30. Write a program to demonstrate the use of push buttons.

SUGGESTED READINGS

1. James Gosling., Bill Joy., Guy, L. Steele Jr., Gilad Bracha., & Alex Buckley. (2014).The Java Language Specification, Java SE (8 ed.). Addison Wesley.
2. Joshua Bloch. (2008). Effective Java (2nd ed.). Addison-Wesley.
3. Cay, S. Horstmann,. GaryCornell. (2012).Core Java 2 Volume 1 (9th ed.). . Prentice Hall, New Delhi.
4. Cay, S. Horstmann., Gary Cornell. (2013). Core Java 2 Volume 2 - Advanced Features(9th ed.). Printice Hall, New Delhi.
5. Balaguruswamy, E. (2009). Programming with Java (4th ed.). McGraw Hill, New Delhi.
6. Paul Deitel., & Harvey Deitel. (2011). Java: How to Program (10th ed.). Prentice Hall, New Delhi.

WEB SITES

1. java.sun.com/docs/books/tutorial/
2. www.en.wikipedia.org/wiki/Java
3. www.java.net/

Semester – II

19CAU212 DISCRETE STRUCTURES - PRACTICAL 3H – 2C

Instruction Hours / week: L: 0 T: 0 P: 3 Marks: Internal : **40** External : **60** Total: **100**
End Semester Exam : 3 Hours

Course Objectives

Enable the student

- To develop logical thinking and its application to computer science
- To write and evaluate a proof or outline the basic structure of and give examples of each proof technique described
- To know the concepts of sets and functions.
- to transform a given problem into a combination of several simpler statements and arrive at a solution to the problem and be able to prove it logically
- To solve problems with the help of tools of mathematical analysis.
- To reason and ability to present a coherent and mathematically accurate argument.

Course Outcomes(COs)

Upon successful completion of the course, the student will be able to:

1. Familiar with elementary algebraic set theory.
2. Demonstrate the ability to write and evaluate a proof or outline the basic structure of and give examples of each proof technique described.
3. Understand the basic principles of sets and operations in sets.
4. Demonstrate an understanding of relations and functions and be able to determine their properties.
5. Demonstrate different traversal methods for trees and graphs.
6. Initiate to knowledge from inference theory.

List of Programs

1. Write a C Program to find the number of subsets of a set contains n elements.
2. Write a C Program to find transitive closure of a relation.
3. Write a C Program to prove
1/ (1*2) + 1/(2*3) ……+1/(n(n+1)) = n/(n+1)

4. Write a C Program to to perform the sum = 1+ (1+2) + (1+2+3) + … + (1+2…+n)

5. Write a C program to print Fibonacci series till Nth term using recursion

6. Write a Cprogram in c to calculate factorial of a number using recursion

7. Write a C Program to find a minimum spanning tree using Prim's algorithm

8. Write a C program to find the shortest path with the lower cost in a graph using Dijkstra's Algorithm

9. Write a C Program to construct the truth table for the following formula.

(i) P∧ Q∧⌉R (ii) P∧⌉Q∧ R (iii) P∧ Q∧⌉R

10. Write a C Program to prove De – Morgan's law.

SUGGESTED READINGS

1. Kenneth Rosen. (2006). Discrete Mathematics and Its Applications (6th ed.). McGraw Hill, New Delhi.

2. Tremblay , J .P. , & Manohar, R. (1997). Discrete Mathematical Structures with Applications to Computer Science. McGraw-Hill Book Company, New Delhi.

3. Coremen, T.H., Leiserson, C.E. , & R. L. Rivest. (2009). Introduction to algorithms, (3rd ed.). Prentice Hall on India, New Delhi.

4. Albertson, M. O.,& Hutchinson, J. P. (1988). Discrete Mathematics with Algorithms .: John wiley Publication, New Delhi.

5. Hein, J. L. (2009). Discrete Structures, Logic, and Computability(3rd ed.). Jones and Bartlett Publishers, New Delhi.

6. Hunter, D.J. (2008). Essentials of Discrete Mathematics. Jones and Bartlett Publishers, New Delhi.

Semester – II

19CAU213 COMPUTER NETWORKS AND INTERNET TECHNOLOGIES - PRACTICAL **3H – 2C**

Instruction Hours / week: L: 0 T: 0 P: 3 Marks: Internal : **40** External : **60** Total: **100**
End Semester Exam : 3 Hours

Course Objectives

Enable the student

- To understand the principles of creating an effective web page.
- To develop skills in analyzing the usability of a website.
- To learn the language of HTML and JavaScript.
- To learn to design web pages using Forms, Tables, Frames, TextBox
- To study the basics of JavaScript
- To learn to validate input in web pages using JavaScript

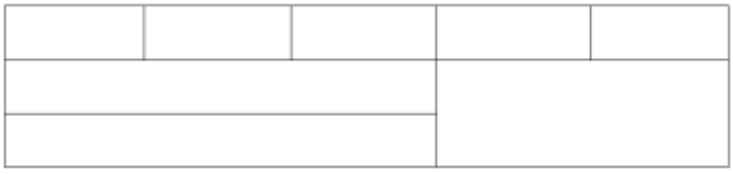
Course Outcomes (COs)

Upon successful completion of the course, the student will be able to:

1. Gain the skills and project-based experience needed for entry into web design and development careers.
2. Develop awareness and appreciation of the many ways that people access the web, and will be able to create standards-based websites that can be accessed by the full spectrum of web access technologies
3. Select and apply markup languages for processing, identifying, and presenting of information in web pages.
4. Create and manipulate web media objects using editing software.
5. Create web pages and validate input using JavaScript.
6. Develop web pages and perform computation using JavaScript.

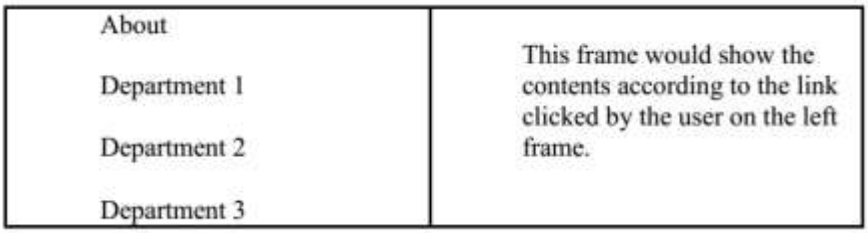
List of Programs

1. Create HTML document with following formatting – Bold, Italics, Underline, Colors, Headings, Title, Font and Font Width, Background, Paragraph, Line Brakes, Horizontal Line, Blinking text as well as marquee text.
2. Create HTML document with Ordered and Unordered lists, Inserting Images, Internal and External linking
3. Create HTML document with Table

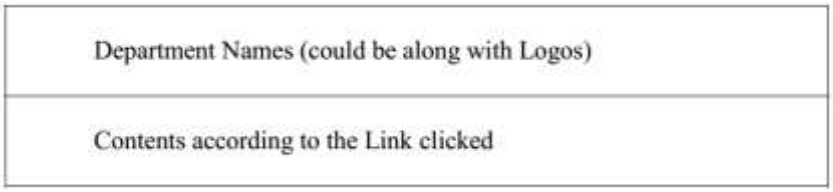


4. Create Form with Input Type, Select and Text Area in HTML.
5. Create an HTML containing Roll No., student‘s name and Grades in a tabular form.

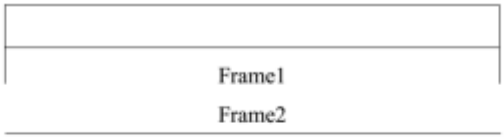
6. Create an HTML document (having two frames) which will appear as follows

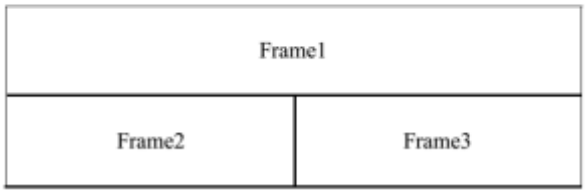


7. Create an HTML document containing horizontal frames as follows



8. Create a website of 6 – 7 pages with different effects as mentioned in above problems.
9. Create HTML documents (having multiple frames) in the following three formats





10. Create a form using HTML which has the following types of controls:
    V. Text Box
    VI. Option/radio buttons
    VII. Check boxes
    VIII. Reset and Submit buttons

List of Practicals using Javascript : Create event driven program for following:

11. Print a table of numbers from 5 to 15 and their squares and cubes using alert.
12. Print the largest of three numbers.
13. Find the factorial of a number n.
14. Enter a list of positive numbers terminated by Zero. Find the sum and average of these numbers.
15. A person deposits Rs 1000 in a fixed account yielding 5% interest. Compute the amount in the account at the end of each year for n years.
16. Read n numbers. Count the number of negative numbers, positive numbers and zeros in the list.

SUGGESTED READINGS

1. Andrew S. Tanenbaum, David J. Wetherall Computer Networks (5th Edition),PHI, 2010
2. B. A. Forouzan, Data Communication and Networking , TMH,2003.
3. D.R. Brooks, An Introduction to HTML and Javascript for Scientists and Engineers, Springer W. Willard,2009
4. HTML A Beginner's Guide, Tata McGraw-Hill Education, 2009.
5. J. A. Ramalho, Learn Advanced HTML 4.0 wit

Semester – III

19CAU301 DATA STRUCTURES 4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives:

Enable the student

- To understand the fundamental concepts of data structures
- To Learn linear data structures – lists, stacks, and queues
- To apply Tree and Graph structures
- To understand sorting, searching and hashing algorithms
- To develop application using data structures
- Choose appropriate data structures and algorithms for problem solving.

Course Outcomes (COs)

Upon successful completion of the course, the student will be able to:

1. Implement abstract data types for linear data structures.
2. Apply the different linear and non-linear data structures to problem solutions.
3. Analyze the applications of tree.
4. Implement graph theory over various data structures.
5. Critically analyze the various sorting algorithms.
6. Able to use linear and non-linear data structures like stacks, queues , linked list etc.

Unit-I

Arrays-Single and Multi-dimensional Arrays, Sparse Matrices (Array and Linked Representation).Stacks Implementing single / multiple stack/s in an Array; Prefix, Infix and Postfix expressions, Utility and conversion of these expressions from one to another; Applications of stack; Limitations of Array representation of stack

Unit –II

Linked Lists Singly, Doubly and Circular Lists (Array and Linked representation); Normal and Circular, representation of Stack in Lists; Self Organizing Lists; Skip Lists Queues, Array and Linked representation of Queue, De-queue, Priority Queues

Unit –III

Trees - Introduction to Tree as a data structure; Binary Trees (Insertion, Deletion , Recursive and Iterative Traversals on Binary Search Trees); Threaded Binary Trees (Insertion, Deletion, Traversals); Height-Balanced Trees (Various operations on AVL Trees).

Unit -IV

Searching and Sorting, Linear Search, Binary Search, Comparison of Linear and Binary Search, Selection Sort, Insertion Sort, Insertion Sort, Shell Sort, Comparison of Sorting Techniques

Unit –V

Hashing - Introduction to Hashing, Deleting from Hash Table, Efficiency of Rehash Methods, Hash Table Reordering, Resolving collusion by Open Addressing, Coalesced Hashing, Separate Chaining, Dynamic and Extendible Hashing, Choosing a Hash Function, Perfect Hashing, Function

Suggested readings

1. 1. Aaron M. Tenenbaum, Moshe J. Augenstein, Yedidyah Langsam,(2003). *Data Structures Using Java*.
2. Aaron M. Tenenbaum, Moshe J. Augenstein, YedidyahLangsam, (2009). *Data Structures Using C and C++*, (2nd ed.), PHI.
3. Adam Drozdek, (2012). *Data Structures and algorithm in C++*, (3rd ed.), Cengage Learning.
4. Goodrich, M. and Tamassia, R.,(2013). *Data Structures and Algorithms Analysis in Java*, (4th ed.), Wiley. Herbert Schildt, (2014). "Java The Complete Reference (English) 9th Edition Paperback", Tata McGraw Hill.
5. John Hubbard,(2009). *Data Structures with JAVA*, (2nd ed.), India, McGraw Hill Education.
6. Malik, D. S., Nair, P.S., (2003). *Data Structures Using Java*, Course Technology.
7. Malik, D.S., (2010). *Data Structure using C++*,(2nd ed.), Cengage Learning.
8. Mark Allen Weiss, (2011). *Data Structures and Algorithms Analysis in Java*, Pearson Education, (3rd ed.).
9. Robert L. Kruse, (1999). *Data Structures and Program Design in C++*, Pearson.
10. Robert Lafore, (2003). *Data Structures and Algorithms in Java*, (2nd ed.), Pearson Macmillan Computer Publications.
11. Sartaj Sahni, (2011). *Data Structures, Algorithms and applications in C+*, (2nd ed.) Universities Press.

Websites

1. http://en.wikipedia.org/wiki/Data_structure
2. http://www.cs.sunysb.edu/~skiena/214/lectures/
3. www.amazon.com/Teach-Yourself-Structures-Algorithms

Semester – III

19CAU302 OPERATING SYSTEMS 4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**
End Semester Exam: 3 Hours

Course Objectives:
Enable the student

- To Study the basic concepts and functions of operating systems.
- To understand the structure and functions of OS.
- To Learn about Processes, Threads and Scheduling algorithms.
- To Understand the principles of concurrency, Deadlocks and Memory Management
- To Learn about the Protection and Security Concepts.
- Understand basic resource management techniques.

Course Outcomes (COs)
Upon completion of this course the students will be able to:

1. Design various Scheduling algorithms.
2. Apply the principles of concurrency.
3. Design deadlock, prevention and avoidance algorithms.
4. Compare and contrast various memory management schemes.
5. Apply the Security Concepts based on Authentication.
6. Appreciate the role of operating system as System software.

Unit –I

Introduction to Operating System: Basic OS Functions-Resource Abstraction-Types of Operating Systems–Multiprogramming Systems-Batch Systems-Time Sharing Systems-Operating Systems for Personal Computers & Workstations-Process Control & Real Time Systems.

Unit –II

Operating System Organization: Processor and user modes-Kernels-System Calls and System Programs. **Process Management:** System view of the process and resources- Process abstraction-Process hierarchy-Threads-Threading issues-Thread libraries-Process Scheduling-Non pre-emptive and Preemptive scheduling algorithms-Concurrent and processes-Critical Section-Semaphores-Methods for inter-process communication- Deadlocks.

Unit –III

Memory Management: Physical and Virtual address space-Memory Allocation strategies –Fixed and Variable partitions-Paging-Segmentation-Virtual memory.

Unit –IV

File and I/O Management: Directory structure-File operations-File Allocation methods-Device management.

Unit -V

Protection and Security: Policy mechanism-Authentication-Internal aCSUess Authorization.

Suggested readings

1. Silberschatz,A., Galvin, P.B., Gagne,G.,(2008). *Operating Systems Concepts*, (8th ed.), John Wiley Publications.
2. Stallings,W., (2008). *Operating Systems, Internals & Design Principles*, (5th Edition), Prentice Hall of India.
3. Tanenbaum, A.S., (2007). *Modern Operating Systems*, (3rd ed.), Pearson Education.

Websites

1. www.cs.columbia.edu/~nieh/teaching/e6118_s00/
2. www.clarkson.edu/~jnm/cs644pages.cs.wisc.edu/~remzi/Classes/736/Fall2002/

Semester – III

19CAU303 ADVANCED NETWORKING 4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives:

Enable the student

- To study the basic taxonomy and terminology of the computer networking and enumerate the layers of OSI model and TCP/IP model.
- To read the fundamentals and basics concepts of Physical layer with real time examples
- To study data link layer concepts, design issues, and protocols.
- To learn the functions of network layer and the various routing protocols.
- To familiarize the functions and protocols of the Transport layer and Application layer.
- Channel error detection and correction, MAC protocols, Ethernet and WLAN.

Course Outcomes (COs)

Upon successful completion of the course, the student will be able to:

1. Describe the functions of each layer in OSI and TCP/IP model.
2. Explain the functions of Application layer and Presentation layer paradigms and Protocols.
3. Describe the Session layer design issues and Transport layer services.
4. Classify the routing protocols and analyze how to assign the IP addresses for the given network.
5. Describe the functions of data link layer and explain the protocols.
6. Explain the types of transmission media with real time applications

Unit – I

Introduction to Computer Networks : Network definition; network topologies; network classifications; network protocol; layered network architecture; overview of OSI reference model; overview of TCP/IP protocol suite. **Data Communication Fundamentals and Techniques**: Analog and digital signal; data-rate limits; digital to digital line encoding schemes; pulse code modulation; parallel and serial transmission;

Unit –II

Digital to analog modulation-; multiplexing techniques- FDM, TDM; transmission media. **Networks Switching Techniques and Access mechanisms:** Circuit switching; packetswitching - connectionless datagram switching, connection-oriented virtual circuit switching; dial-up modems; digital subscriber line; cable TV for data transfer.

UNIT –III

Data Link Layer Functions and Protocol: Error detection and error correction techniques; data-link control- framing and flow control; error recovery protocols- stop and wait ARQ, go-back-n ARQ; Point to Point Protocol on Internet.

Unit –IV

Multiple Access Protocol and Networks: CSMA/CD protocols; Ethernet LANS; connecting LAN and back-bone networks- repeaters, hubs, switches, bridges, router and gateways;

Networks Layer Functions and Protocols: Routing; routing algorithms; network layer protocol of Internet- IP protocol, Internet control protocols.

Unit –V

Transport Layer Functions and Protocols: Transport services- error and flow control, Connection establishment and release- three way handshake;

Overview of Application layer protocol: Overview of DNS protocol; overview of WWW &HTTP protocol.

Suggested readings

1. Forouzan B. A., (2007). *Data Communications and Networking*, (4th ed.), THM.
2. Tanenbaum, A. S. , (2002). Computer Networks**,** (4th ed.), PHI.

Websites

1. en.wikipedia.org/wiki/Internet_protocol_suite
2. http://docwiki.cisco.com/wiki/Introduction_to_WAN_Technologies
3. www.yale.edu/pclt/COMM/**TCPIP**.HTM
4. www.w3schools.com/**tcpip**/default.asp
5. https://www.cse.iitb.ac.in/

Semester – III

19CAU304A ANDROID PROGRAMMING 3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives:

Enable the student

- To compare the differences between Android and other mobile development environments.
- To learn the Object-oriented features of Kotlin and APIs for Android Development.
- To describe the working of Android applications, life cycle, manifest, and Intents
- To demonstrate the implementation of Form widgets for Android App development.
- To learn the SQLite database connectivity and database operations with android
- To design and develop useful Android applications with compelling user. interfaces .by using, extending, and creating your own layouts and Views and using Menus.

Course Outcomes (COs)

Upon successful completion of the course, the student will be able to:

1. Analyze the Architecture and features of Android with another Mobile Operating System.
2. Evaluate the standard of Kotlin language for developing Android Applications
3. Apply knowledge for creating user Interface and develop activity for Android App.
4. Evaluate the user interface architecture of Android for developing Android Apps
5. Understand the implementation of SQLite database operations with Android.
6. Design and implement Database Application and Content providers.

Unit -I

Introduction: History of Android, Introduction to Android Operating Systems, Android Development Tools, Android Architecture.

Unit –II

Overview of object oriented programming using Java: OOPs Concepts: Inheritance, Polymorphism, Interfaces, Abstract class, Threads, Overloading and Overriding, Java Virtual Machine.

Unit –III

Development Tools: Installing and using Eclipse with ADT plug-in, Installing Virtual machine for Android sandwich/Jelly bean (Emulator), configuring the installed tools, creating a android project– Hello Word, run on emulator, Deploy it on USB-connected Android device.

Unit –IV

User Interface Architecture: Application context, intents, Activity life cycle, multiple screen size **User Interface Design:** Form widgets, Text Fields, Layouts, Button control, toggle buttons, Spinners(Combo boxes),Images, Menu, Dialog

Unit –V

Database: Understanding of SQL database, connecting with the database.

Suggested readings

1. James C.Sheusi,(2013). *Android application development for Java for Java programmers*, Cengage Learning.

Websites

1. http://www.developer.android.com
2. http://developer.android.com/about/versions/index.html
3. http://developer.android.com/training/basics/firstapp/index.html
4. http://developer.android.com/guide/components/activities.html
5. http://developer.android.com/guide/components/fundamentals.html
6. http://developer.android.com/guide/components/intents-filters.html.
7. http://developer.android.com/training/multiscreen/screensizes.html
8. http://developer.android.com/guide/topics/ui/controls.html
9. http://developer.android.com/guide/topics/ui/declaring-layout.html
10. http://developer.android.com/training/basics/data-storage/databases.html

Semester – III

19CAU304B STRUTS FRAMEWORK 3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives:

Enable the students to

- Understand the Model, View, Controller (MVC) design pattern and how it is applied by Struts Framework
- Know the components of Struts Application and database connectivity
- Implement JSP functions using Struts
- Perform client and server side validation using Struts Validator Framework
- Develop web applications using Struts
- Develop EJB programs and get familiar with Struts framework

Course Outcome (COs):

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

1. Know about MVC and overview of JSP
2. Identify the components of a Struts Application and how to connect database in web based applications
3. Know about the struts Sub elements and Control Tags
4. Develop programs with Data Tags and Bean Tags
5. Develop programs with HTML Tags and Logic Tags and able to construct enterprise-level web based applications
6. Construct web based applications and Identify where data structures are appearing in them.

Unit –I

Introduction, Understanding the MVC Design Pattern, The Struts Implementation of the MVC, Directory Structure, Web Application Deployment Descriptor, The Tomcat JSP/Servlet Container, Installing and Configuring Tomcat, Testing Your Tomcat Installation, An Overview of the Java Servlet and JavaServer Pages, The GenericServlet and HttpServlet Classes, Life Cycle of a Servlet. Struts OverView, Life Cycle of Struts.

Unit -II

Components of a Struts Application, The Controller, The View, DynaActionForm & LazyDynaBean, ActionServlet, RequestProcessor, ActionForm, IncludeAction, Forward Action, LocaleAction, DispatchAction, LookupDispatchAction, MappingDispatchAction, EventDispatchAction, SwitchAction, Interceptors, Implementing Custom interceptors, Struts Validation, Exception Handling, Managing Errors, Struts Error Management - ActionError,

ActionErrors, Creating Custom ActionMappings, Struts JDBC Connection, Using a DataSource in Struts Application, Debugging Struts Applications.

Unit -III

The struts-config.xml, The Struts Subelements, The icon Tag Subelement, display-name Tag Subelement, description Tag Subelement, set-property Tag Subelement, Adding a Struts DataSource, Adding FormBean Definitions, Adding Global Forwards, Adding Actions, Adding a RequestProcessor, Adding Message Resources, Adding a Plug-in. The Bean Tag Library, Installing the Bean Tags, bean:cookie Tag, bean:define Tag, bean:header Tag, bean:include Tag, bean:message Tag, bean:page Tag, bean:parameter Tag ,bean:resource Tag,<bean:size Tag, bean:struts Tag, bean:write Tag

Unit -IV

HTML Tag Library, Base Tag, Button Tag, Cancel Tag, Checkbox Tag, Errors Tag, Form Tag, Hidden Tag, Html Tag, Image Tag, Img Tag, Link Tag, Multibox Tag, Select Tag, Option Tag, Options Tag, Password Tag, Radio Tag, Reset Tag, Rewrite Tag, Submit Tag, Text Tag, Textarea Tag

Unit -V

The Logic Tag Library , Empty Tag, notEmpty Tag ,equal Tag, notEqual Tag, forward Tag, redirect Tag, greaterEqual Tag, greaterThan Tag, iterate Tag, lessEqual Tag, lessThan Tag, match Tag, notMatch Tag, present Tag, notPresent Tag

Suggested readings

1. James Goodwill,(2002). *Mastering Jakarta Struts*, Wiley Publishing, Inc.

Semester – III

19CAU311 DATA STRUCTURES - PRACTICAL 4H – 2C

Instruction Hours / week: L: 0 T: 0 P: 4 Marks: Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objective

- To understand the fundamental concepts of data structures
- To Learn linear data structures – lists, stacks, and queues
- To apply Tree and Graph structures
- To understand sorting, searching and hashing algorithms
- To develop application using data structures
- Choose appropriate data structures and algorithms for problem solving.

Course Outcomes (COs)

Upon successful completion of the course, the student will be able to:

1. Implement abstract data types for linear data structures.
2. Apply the different linear and non-linear data structures to problem solutions.
3. Analyze the applications of tree.
4. Implement graph theory over various data structures.
5. Critically analyze the various sorting algorithms.
6. Able to use linear and non-linear data structures like stacks, queues , linked list etc.

LIST OF PROGRAMS:

1. Write a program to search an element from a list. Give user the option to perform Linear or Binary search. Use Template functions.
2. WAP using templates to sort a list of elements. Give user the option to perform sorting using Insertion sort, Bubble sort or Selection sort.
3. Implement Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list and concatenate two linked lists (include a function and also overload operator +).
4. Implement Doubly Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.
5. Implement Circular Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.
6. Perform Stack operations using Linked List implementation.
7. Perform Stack operations using Array implementation. Use Templates.
8. Perform Queues operations using Circular Array implementation. Use Templates.

9. Create and perform different operations on Double-ended Queues using Linked List implementation.
10. Write a program to scan a polynomial using linked list and add two polynomial.
11. Write a program to calculate factorial and to compute the factors of a given no. (i)using recursion, (ii) using iteration
12. Write a program to display Fibonacci series (i)using recursion, (ii) using iteration
13. Write a program to calculate GCD of 2 number (i) with recursion (ii) without recursion
14. Write a program to create a Binary Search Tree and include following operations in tree:

    (a) Insertion (Recursive and Iterative Implementation)
    (b) Deletion by copying
    (c) Deletion by Merging
    (d) Search a no. in BST
    (e) Display its preorder, postorder and inorder traversals Recursively
    (f) Display its preorder, postorder and inorder traversals Iteratively
    (g) Display its level-by-level traversals
    (h) Count the non-leaf nodes and leaf nodes
    (i) Display height of tree
    (j) Create a mirror image of tree
    (k) Check whether two BSTs are equal or not

15. Write a program to convert the Sparse Matrix into non-zero form and vice-versa.
16. Write a program to reverse the order of the elements in the stack using additional stack.
17. Write a program to reverse the order of the elements in the stack using additional Queue.
18. Write a program to implement Diagonal Matrix using one-dimensional array.
19. Write a program to implement Lower Triangular Matrix using one-dimensional array.
20. Write a program to implement Upper Triangular Matrix using one-dimensional array.
21. Write a program to implement Symmetric Matrix using one-dimensional array.
22. Write a program to create a Threaded Binary Tree as per inorder traversal, and implement operations like finding the suCSUessor / predecessor of an element, insert an element, inorder traversal.
23. Write a program to implement various operations on AVL Tree.

Suggested readings

1. Aaron M. Tenenbaum, Moshe J. Augenstein, Yedidyah Langsam,(2003). *Data Structures Using Java*.
2. Aaron M. Tenenbaum, Moshe J. Augenstein, YedidyahLangsam, (2009). *Data Structures Using C and C++*, (2nd ed.), PHI.
3. Adam Drozdek, (2012). *Data Structures and algorithm in C++*, (3rd ed.), Cengage Learning.
4. Goodrich, M. and Tamassia, R.,(2013). *Data Structures and Algorithms Analysis in Java*, (4th ed.), Wiley. Herbert Schildt, (2014). "Java The Complete Reference (English) 9th Edition Paperback", Tata McGraw Hill.
5. John Hubbard,(2009). *Data Structures with JAVA*, (2nd ed.), India, McGraw Hill Education.

Websites

1. http://en.wikipedia.org/wiki/Data_structure
2. http://www.cs.sunysb.edu/~skiena/214/lectures/
3. www.amazon.com/Teach-Yourself-Structures-Algorithms

Semester – III

19CAU312 OPERATING SYSTEMS - PRACTICAL 4H – 2C

Instruction Hours / week: L: 0 T: 0 P: 4 Marks: Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objective

- To Study the basic concepts and functions of operating systems.
- To understand the structure and functions of OS.
- To Learn about Processes, Threads and Scheduling algorithms.
- To Understand the principles of concurrency, Deadlocks and Memory Management
- To Learn about the Protection and Security Concepts.
- Understand basic resource management techniques.

Course Outcomes (COs)

Upon successful completion of the course, the student will be able to:

1. Design various Scheduling algorithms.
2. Apply the principles of concurrency.
3. Design deadlock, prevention and avoidance algorithms.
4. Compare and contrast various memory management schemes.
5. Apply the Security Concepts based on Authentication.
6. Appreciate the role of operating system as System software.

LIST OF PROGRAMS:

1. Write a program (using *fork()* and/or *exec()* commands) where parent and child execute:
   a) same program, same code.
   b) same program, different code.
   c) before terminating, the parent waits for the child to finish its task.
2. Write a program to report behavior of Linux kernel including CPU information and. memory information
3. Write a program to print file details including owner a access permissions, file access time, where file name is given as argument.
4. Write a program to copy files using system calls.
5. Write a program to implement first-fit, best-fit and worst-fit allocation strategies.
6. Write program to implement FCFS scheduling algorithm.
7. Write program to implement Round Robin scheduling algorithm.

8. Write program to implement SJF scheduling algorithm.
9. Write program to implement non-preemptive priority based scheduling algorithm.
10. Write program to implement preemptive priority based scheduling algorithm.
11. Write program to implement SRJF scheduling algorithm.
12. Write program to calculate sum of n numbers using *thread* library.

Suggested readings

1. Silberschatz,A., Galvin, P.B., Gagne,G.,(2008). *Operating Systems Concepts*, (8th ed.), John Wiley Publications.
2. Stallings,W., (2008). *Operating Systems, Internals & Design Principles*, (5th Edition), Prentice Hall of India.
3. Tanenbaum, A.S., (2007). *Modern Operating Systems*, (3rd ed.), Pearson Education.

Websites

1.www.cs.columbia.edu/~nieh/teaching/e6118_s00/

2.www.clarkson.edu/~jnm/cs644pages.cs.wisc.edu/~remzi/Classes/736/Fall2002/

Semester – III

19CAU313 ADVANCED NETWORKING - PRACTICAL 4H – 2C

Instruction Hours / week: L: 0 T: 0 P: 4 Marks: Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objective

- To study the basic taxonomy and terminology of the computer networking and enumerate the layers of OSI model and TCP/IP model.
- To read the fundamentals and basics concepts of Physical layer with real time examples
- To study data link layer concepts, design issues, and protocols.
- To learn the functions of network layer and the various routing protocols.
- To familiarize the functions and protocols of the Transport layer and Application layer.
- Channel error detection and correction, MAC protocols, Ethernet and WLAN.

Course Outcomes (COs)

Upon completion of this course, the student will be able to:

1. Describe the functions of each layer in OSI and TCP/IP model.
2. Explain the functions of Application layer and Presentation layer paradigms and Protocols.
3. Describe the Session layer design issues and Transport layer services.
4. Classify the routing protocols and analyze how to assign the IP addresses for the given network.
5. Describe the functions of data link layer and explain the protocols.
6. Explain the types of transmission media with real time applications

LIST OF PROGRAMS:

1. Simulate Cyclic Redundancy Check (CRC) error detection algorithm for noisy channel.
2. Simulate and implement stop and wait protocol for noisy channel.
3. Simulate and implement go back n sliding window protocol.
4. Simulate and implement selective repeat sliding window protocol.
5. Simulate and implement distance vector routing algorithm
6. Simulate and implement Dijkstra algorithm for shortest path routing.

Suggested readings

1. Forouzan B. A., (2007). *Data Communications and Networking*, (4th ed.), THM.
2. Tanenbaum, A. S. , (2002). Computer Networks**,** (4th ed.), PHI.

Websites

1. en.wikipedia.org/wiki/Internet_protocol_suite
2. http://docwiki.cisco.com/wiki/Introduction_to_WAN_Technologies

Semester – III

19CAU314A ANDROID PROGRAMMING PRACTICAL 3H – 1C

Instruction Hours / week: L: 0 T: 0 P: 3 Marks: Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objective

Enable the student

- To compare the differences between Android and other mobile development environments.
- To learn the Object-oriented features of Kotlin and APIs for Android Development.
- To describe the working of Android applications, life cycle, manifest, and Intents
- To demonstrate the implementation of Form widgets for Android App development.
- To learn the SQLite database connectivity and database operations with android
- To design and develop useful Android applications with compelling user. interfaces .by using, extending, and creating your own layouts and Views and using Menus.

Course Outcomes (COs)

Upon completion of this course, the student will be able to:

1. Analyze the Architecture and features of Android with another Mobile Operating System.
2. Evaluate the standard of Kotlin language for developing Android Applications
3. Apply knowledge for creating user Interface and develop activity for Android App.
4. Evaluate the user interface architecture of Android for developing Android Apps
5. Understand the implementation of SQLite database operations with Android.
6. Design and implement Database Application and Content providers.

LIST OF PROGRAMS:

1. Create ―Hello World‖ application. That will display ―Hello World‖ in the middle of the screen in the emulator. Also display ―Hello World in the middle of the screen in the Android Phone.
2. Create an application with login module. (Check username and password).
3. Create spinner with strings taken from resource folder (res >> value folder) and on changing the spinner value, Image will change.
4. Create a menu with 5 options and and selected option should appear in text box.
5. Create a list of all courses in your college and on selecting a particular course teacher-in-charge of that course should appear at the bottom of the screen.
6. Create an application with three option buttons, on selecting a button colour of the screen will change.

7. Create and Login application as above. On successful login, pop up the message.
8. Create an application to Create, Insert, update, Delete and retrieve operation on the database.

Suggested readings

1. James C.Sheusi,(2013). *Android application development for Java for Java programmers*, Cengage Learning.

Websites

1. http://www.developer.android.com
2. http://developer.android.com/about/versions/index.html
3. http://developer.android.com/training/basics/firstapp/index.html
4. http://developer.android.com/guide/components/activities.html
5. http://developer.android.com/guide/components/fundamentals.html
6. http://developer.android.com/guide/components/intents-filters.html.
7. http://developer.android.com/training/multiscreen/screensizes.html
8. http://developer.android.com/guide/topics/ui/controls.html
9. http://developer.android.com/guide/topics/ui/declaring-layout.html
10. http://developer.android.com/training/basics/data-storage/databases.html

Semester – III

19CAU314B STRUTS FRAMEWORK - PRACTICAL 3H – 1C

Instruction Hours / week: L: 0 T: 0 P: 3 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives:

Enable the students to

- Understand the Model, View, Controller (MVC) design pattern and how it is applied by Struts Framework
- Know the components of Struts Application and database connectivity
- Implement JSP functions using Struts
- Perform client and server side validation using Struts Validator Framework
- Develop web applications using Struts
- Develop EJB programs and get familiar with Struts framework

Course Outcome (COs):

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

1. Know about MVC and overview of JSP
2. Identify the components of a Struts Application and how to connect database in web based applications
3. Know about the struts Sub elements and Control Tags
4. Develop programs with Data Tags and Bean Tags
5. Develop programs with HTML Tags and Logic Tags and able to construct enterprise-level web based applications
6. Construct web based applications and Identify where data structures are appearing in them.

LIST OF PROGRAMS:

1. Create a Simple Struts Human Resource (HR) Application
2. Create Struts Login Application Using Action form
3. Create a Strut Global Action Forwards and Action Mappings
4. Create and implement Multiple Struts Configuration File
5. Create Struts Application to implement struts Lookup Dispatch Action
6. Create Struts Application to mapping the struts Action Servlet
7. Create a Feedback form to implement struts HTML Tag
8. Create Struts Application to use of action forms for validating user input.

9. Create Struts Applications to implement Struts Logic Tag
10. Create a Struts LOGIC tag library provides tags that are useful in managing conditional generation of output text

Suggested readings

1. James Goodwill,(2002). *Mastering Jakarta Struts*, Wiley Publishing, Inc.

Semester – IV

19CAU401 RELATIONAL DATABASE MANAGEMENT SYSTEMS 4H-4C

Instruction Hours / week: L:4 T:0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

Enable the student

- To describe a good introduction to the discipline of database management systems.
- To give a good formal foundation on the data models and E-R model.
- To demonstrate the principles database constraints behind systematic database design by covering normalization concept.
- To introduce the concepts of basic SQL as a universal Database language.
- To have an introductory knowledge about the PL/SQL concept
- To train the student to translate business requirements into relational database schemas and manipulate databases using the SQL Data Manipulation Language.

Course Outcomes (COs)

Upon completion of this course, students will be able to:

1. Demonstrate an understanding of the elementary features of RDBMS
2. Design conceptual models of a database using ER modeling for real life applications
3. Analyze the existing design of a database schema and apply concepts of normalization to design an optimal database
4. Able to develop structured query language (SQL) queries to create, read, update, and delete relational database
5. Design efficient PL/SQL programs to access Oracle databases
6. Familiar with basic database storage structures and access techniques: file and page organizations, indexing methods including B tree, and hashing.

Unit I

Databases and Database Users --Characteristics of the Database Approach - Actors on the Scene- Workers behind the Scene- Advantages of Using the DBMS Approach- A Brief History of Database Applications- Database System Concepts and Architecture : Data Models, Schemas, and Instances - Three-Schema Architecture and Data Independence- Database Languages and Interfaces- The Database System Environment- Centralized and Client/Server Architectures for DBMSs- Classification of Database Management Systems.

Unit II

Data Modeling Using the Entity-Relationship (ER) :Conceptual Data Models for Database Design -A Sample Database Application -Entity Types, Entity Sets, Attributes, and Keys - Relationship Types, Relationship Sets, Roles,and Structural Constraints -Weak Entity Types -ER Diagrams, Naming Conventions, and Design Issues. The Enhanced Entity-Relationship: Subclasses, Superclasses, and Inheritance - Specialization and Generalization - Constraints and

Characteristics of Specialization and Generalization Hierarchies -Modeling of UNION Types Using Categories- Data Abstraction, Knowledge Representation.

Unit III

Relational Algebra: Relational Algebraic Operations-Aggregate Function-Update Operations.SQL: Characteristics-Advantages& Action-Data types and Literals-Types of SQL Commands:DDL,DML,DQL,DCL,DAS,TCS-SQL Operators-Arithmetic, Comparsion, Logical& Set Operator-Operator Preceedence. Tables, view and Indexes: Introduction-View-Indexes. Queries and subqueries: Introduction-Subqueries-Aggregate Function-Insert, update and Delete Operations.

Unit IV

Overview – declaration section – executable command section: conditional logic, loops, CASE statements – exception handling section. Triggers: definition – types: row level, statement level, before and after, instead of – syntax – enabling and disabling triggers - replacing and dropping triggers. Cursors – definition – open – fetch – close – cursor attributes- select for update – types : implicit, explicit. Procedures, Functions, and Packages: Local and global – procedures vs. functions – stored procedures, functions – create procedure syntax - create function syntax – package header – package body – calling procedures, functions, package members. Replacing and dropping procedures, functions, packages.

Unit V

Database Design Theory and Normalization: Basics of Functional Dependencies and Normalization for Relational Databases-Informal Design Guidelines for Relation Schemas-Functional Dependencies-Normal Forms Based on Primary Keys-General Definitions of Second and Third Normal Forms-Boyce-Codd Normal Form-Multivalued Dependency and Fourth Normal Form-Join Dependencies and Fifth Normal Form.

TEXT BOOKS

1. Ramez Elmasri and Shamkant B. Navathe.2011. Fundamental of Database Systems.6th edition.Pearson India.[Unit I to V]
2. Kevin Loney and George Koch. 2009. Oracle 11g The Complete Reference, 1st Edition, Tata Mcgraw-Hill, New Delhi.[Unit IV]

REFERENCE BOOKS

1. Ramez Elmasri.2013. Fundamentals of Database Systems: Models, Languages, Design and Application Programming , 6th edition Pearson India.
2. Bipin C. Desai. 2008. An Introduction to Database Systems, Galgotia Publications, New Delhi.
3. Gerald V. Post. Database Management Systems Designing and Building Business Applications, 2nd Edition, Tata McGraw-Hill, New Delhi.
4. Rajesh Narang. 2006. Database Management Systems. Prentice Hall of India, New Delhi

5. Alexix and Mathews Leon, 2006. Fundamentals of Database Management System, 1st Edition Vijay Nicole imprints private Limited

WEB SITES

1. http://en.wikipedia.org/wiki/RDBMS
2. http://aspalliance.com/1211_Relational_Database_Management_Systems__Concepts_and_Terminologies
3. www.compinfo-center.com/apps/rdbms.html

4. **Semester – IV**

19CAU402 SOFTWARE ENGINEERING 4H –4C

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

- To introduce the fundamental concepts of software engineering.
- To Analyze, specify and document software requirements for a software system.
- To Design, select and apply the most appropriate software engineering process for a given project, plan for a software project, identify its scope and risks, and estimate its cost and time.
- Expose the criteria for test cases.
- Be familiar with test management and test automation techniques
- Implement a given software design using sound development practices.

Course Outcomes (COs)

Upon completion of this course the students will be able to:

1. Identify suitable life cycle models to be used and translate a requirement specification to a design using an appropriate software engineering methodology.
2. Apply systematic procedure for software design and deployment.
3. Analyze a problem and identify and define the computing requirements to the problem.
4. Formulate appropriate testing strategy for the given software system.
5. Develop software projects based on current technology, and test the software using testing tools.
6. An ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives

Unit -I

Introduction: The Evolving Role of Software, Software Characteristics, Changing Nature of Software, Software Engineering as a Layered Technology, Software Process Framework, Framework and Umbrella Activities, Process Models, Capability Maturity Model Integration (CMMI).

Unit-II

Requirement Analysis; Initiating Requirement EngineeringProcess- Requirement Analysis and Modeling Techniques- FlowOriented Modeling- Need for SRS- Characteristics and Components of SRS- Software Project Management: Estimation in Project Planning Process, Project Scheduling.

Unit-III

Risk Management: Software Risks, Risk Identification Risk Projection and Risk Refinement, RMMM plan, **Quality Management-** Quality Concepts, Software Quality Assurance, Software Reviews, Metrics for Process and Projects

Unit-IV

Design Engineering-Design Concepts, Architectural Design Elements, Software Architecture, Data Design at the Architectural Level and Component Level, Mapping of Data Flow into Software Architecture, Modeling Component Level Design

Unit-V

Testing Strategies & Tactics: Software Testing Fundamentals, Strategic Approach to Software Testing, Test Strategies for Conventional Software, Validation Testing, System testing Black-Box Testing, White-Box Testing and their type, Basis Path Testing

Suggested Readings

1. Aggarwal K.K., Singh,Y., (2008). *Software Engineering*, (2nd ed.), New Age International Publishers.
2. Bell,D., (2005). *Software Engineering for Students*, (4th ed.), Addison-Wesley.
3. Jalote,P., (2008). *An Integrated Approach to Software Engineering* (2nd ed.), New Age International Publishers.
4. Mall,R.,(2004). *Fundamentals of Software Engineering*, (2nd ed.), Prentice-Hall of India.
5. Pressman, R.S.,(2009). *Software Engineering: A Practitioner's Approach,* (7th Edition), McGraw-Hill.
6. Sommerville, I.,(2006), *Software Engineering*, (8th ed.), Addison Wesley.

Websites

1. http://en.wikipedia.org/wiki/Software_engineering
2. http://www.onesmartclick.com/engineering/software-engineering.html
3. http://www.CSU.gatech.edu/classes/AY2000/cs3802_fall/

Semester – IV

19CAU403 WEB PROGRAMMING 4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

Enable the student

- To understand the concepts and architecture of the World Wide Web.
- To understand and practice markup languages
- Understand the technologies used in Web Programming.
- Know the importance of object-oriented aspects of Scripting.
- To understand and practice embedded dynamic scripting on client-side Internet Programming
- To develop dynamic web applications, create and consume web services

Course Outcomes (COs)

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

1. Design web pages.
2. Use technologies of Web Programming.
3. Apply object-oriented aspects to Scripting.
4. Create a basic website using HTML and Cascading Style Sheets.
5. Design and implement dynamic web page with validation using JavaScript objects and by applying different event handling mechanisms.
6. Use scripting languages and web services to transfer data and add interactive components to web pages.

Unit-I

Creating an HTML Document – creating list – creating links between document – linking to resources on the internet – working with hypertext attributes – working with fonts and text styles – tables – creating frameset – working with forms – working with cascading style sheets.

Unit-II

JavaScript: Introduction to javascript – Programming fundamentals – Functions and objects – Navigator object model

Unit-III

JavaScript: Form and form elements – Scripting frames and multiple windows – Event object – Functions and custom objects.

Unit-IV

ASP: Client side scripting vs. Server side scripting- Variables & Constants- Procedures – Forms – Cookies – Application - #include – Global.asa - Functions-ASP object model: Response- Request- Application- Session – Server – Error – Array

Unit-V

ASP: Collections & Control Structure-File system object: File System – Text Stream-Drive – File – Folder – Directory – ADO - sql & Databases for data driven applications-ASP Components: Ad Rotator – Browser Cap. – Content Linking – Content Rotator .

Suggested Readings

1. Danny Goodman, (2000). *Javascript Bible*, (3rd ed.), IDG Books India Pvt Ltd. **(Unit-III).**
2. David Flanagan, (2006). *JavaScript: The Definitive Guide*, O'Reilly,
3. Nicholas C. Zakas, Inc Ebrary and Ebrary,(2005). *Professional JavaScript for Web Developers*, New Delhi , John Wiley & Sons Inc.
4. Patrick Carey, (2005). *New Perspectives on HTML and XHTML*, (1st ed.), Thomson Course Technology Publishing. **(Unit- I).**
5. Rohit Khurana's, (2002). *Javascript Professional edition*, (2nd ed.), A.P.H. Publishing company, NewDelhi**.(Unit -II)**
6. Russell Jones, A. (2000). *Mastering ActiveServerPages 3*, **(**1st ed.), BPB Publishing, New Delhi.(**Unit- IV & Unit -V).**

Web Sites:

1. www.w3schools.com/
2. www.**javascript**kit.com
3. www.**asp**free.com
4. www.aspnettutorials.com

Semester – IV

19CAU404A R PROGRAMMING 3H – 3C

Instruction Hours / week: L:3 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives :

Enable the student

- To acquire the computing tasks such as using conditional processing statements, loops, and writing one's own functions.
- Performing advanced graphing of data and statistical modeling of data.
- Use statistical distribution functions in R
- Read Structured Data into R from various sources
- Understand split-apply-combine (group-wise operations) in R
- Perform basic statistical modeling of data

Course Outcomes (COs)

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

1. Learn how to install and configure software necessary for a statistical programming environment.
2. Discuss generic programming language concepts as they are implemented in a high-level statistical language.
3. The course covers practical issues in statistical computing which includes programming in R, reading data into R, accessing R packages, writing R functions, debugging, and organizing and commenting R code.
4. Import external data into R for data processing and statistical analysis
5. Learn the main R data structures – vector and data frame
6. Learn the file processing in R.

Unit-I

History and Overview of R : The S Philosophy - Back to R -Basic Features of R - Free Software -Design of the R System - Limitations of R- R Resources .Getting Started with R :Installation - Getting started with the R interface -.R Nuts and Bolts :Entering Input - Evaluation -R Objects - Numbers - Attributes - Creating Vectors - Mixing Objects - Explicit Coercion - Matrices -Lists -Factors - Missing Values - Data Frames - Names .

Unit-II

Getting Data In and Out of R :Reading and Writing Data - Reading Data Files with read.table() - Reading in Larger Datasets with read.table - Calculating Memory Requirements for R Objects . Using the readr Package .Using Textual and Binary Formats for Storing Data :Using

dput() and dump() – Binary Formats - Interfaces to the Outside World : File Connections - Reading Lines of a Text File - Reading From a URL Connection - Subsetting R Objects :Subsetting a Vector - Subsetting a Matrix - Subsetting Lists - Subsetting Nested Elements of a List - Extracting Multiple Elements of a List - Partial Matching -Removing NA Values .

Unit-III

Vectorized Operations :Vectorized Matrix Operations .Dates and Times :Dates in R - Times in R - Operations on Dates and Times .Managing Data Frames with the dplyr package :Data Frames -The dplyr Package - dplyr Grammar - Installing the dplyr package - select() - filter() -arrange() - rename() - mutate() - group_by()-%>%.Control Structures :if-else - for Loops - Nested for loops - while Loops - repeat Loops - next, break .

Unit-IV

Functions: Functions in R - Your First Function - Argument Matching - Lazy Evaluation – The Argument - Arguments Coming After the Argument .Scoping Rules of R : A Diversion on Binding Values to Symbol - Scoping Rules - Lexical Scoping: Why Does It Matter? -Lexical vs. Dynamic Scoping -- Application: Optimization - Plotting the Likelihood. Coding Standards for R .Loop Functions : Looping on the Command Line - lapply() - sapply() - split() - Splitting a Data Frame - tapply - apply() - Col/Row Sums and Means -Other Ways to Apply - mapply()- Vectorizing a Function .

Unit-V

Debugging -:Something's Wrong! - Figuring Out What's Wrong - Debugging Tools in R . Using traceback() - Using debug() - Using recover().Profiling R Code: Using system.time() . Timing Longer Expressions - The R Profiler - Using summaryRprof().Simulation :Generating Random Numbers - Setting the random number seed -Simulating a Linear Model - Random Sampling .

Suggested Readings

1. Daniel Navarro, (2013). *Learning Statistics with R*. University of Adelaide Publications.
2. Hadley Wickham, (2014). *Advanced R Programming*, (1st ed.)
3. Jeffrey Stanton, (2013). *Introduction to Data Science, with Introduction to R,* Version 3 ,
4. Roger.D.Peng, (2015). *R Programming for Data Science*

Semester – IV

19CAU404B OPEN SOURCE TECHNOLOGIES 3H-3C

Instruction Hours / week: L: 3 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives :

Enable the student

- To introduce open source technology for development of web applications.
- For Study the problems with traditional commercial software.
- To understand open source scripting language for programming in web environment i.e. PHP.
- To study the open source management system and connection with database.
- To learn open source web server, software tools.
- To learn the open source ideals in order to apply those principles

Course Outcomes (COs)

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

1. Leaned the need of open source technology, open source development model, application of open sources, aspects of open source movement
2. The students will be aware about the problems with traditional commercial software.
3. The student will be familiar with basis syntax of PHP, common PHP scripts elements.
4. The student will be familiar with creating of the server side scripting using PHP, implement PHP database connectivity, perform operation on database and open source database management system.
5. The students will be familiar with Working of different web Servers.
6. The students will be aware about the software tool and process like Eclipse IDE, Selenium ID

Unit-I

Why open source, what is Open source, open source principles, open standards requirements for software, open source successes, free software, some example of free software, free software license provider, free software Vs Open source software, Public Domain , FOSS DOES not Mean any cost, proprietary Vs Open Source Licensing Model.

Principles and Open Source Methodology: History, open source initiatives, open standards principles-methodologies, philosophy, software freedom, open source software, development, Licenses, copyright.

Unit-II

Open source projects: Starting and maintaining an open source project, open source hardware- open source design-open source teaching (OST).Open Source Ethics: Open Source Vs Closed Source-Open source Government-The ethics of open source-social and financial impacts of open source technology-shared software, shared source.

Unit-III

Apache Web Server: Introduction-Starting, Stopping, and Restarting Apache-Configuration-Securing Apache Create the Web Site-Apache Log Files

Unit-IV

MySQL: Introduction-Tutorial-Database Independent Interface-Table Joins-Loading and Dumping a Database

Unit-V

Perl: Introduction-Perl Documentation-Perl Syntax Rules-A Quick Introduction To Object-Oriented Programming-What We Didn't Talk About

Suggested Readings

1. Andrew M. St. Laurent, (2004). *Understanding Open Source and Free Software Licensing*, O'Reilly Media.
2. Dan Woods, Gautam Guliani, (2005). Open Source for the Enterprise, O'Reilly Media.
3. Fadi P. Deek and James A. M. McHugh,(2007). *Open Source Technology and Policy*, Cambridge University Press.
4. James Lee, Brent Ware,2002. *Open Source Web Development with LAMP: Using Linux, Apache, MySQL, Perl, and PHP*, , Publisher: Addison Wesley Date [UNIT IV – V]
5. Kailashvadera, Vhavyesh Gandhi, (2009). *Open source Technologies, Lakshmi Publications*,(1st ed.). [UNIT I – III]
6. Nick Wells,(2012). *The Complete Guide to Linux System Administration*, Delmar Cengage Learning.

Semester – IV

19CAU411 RELATIONAL DATABASE MANAGEMENT SYSTEMS - PRACTICAL

4H– 2C

Instruction Hours / week: L: 0 T: 0 P:4 C:2 **Marks:** Internal: **40** External: **60** Total: **100**
End Semester Exam: 3 Hours

Course objective
Enable the student

- To describe a good introduction to the discipline of database management systems.
- To give a good formal foundation on the data models and E-R model.
- To demonstrate the principles database constraints behind systematic database design by covering normalization concept.
- To introduce the concepts of basic SQL as a universal Database language.
- To have an introductory knowledge about the PL/SQL concept
- To train the student to translate business requirements into relational database schemas and manipulate databases using the SQL Data Manipulation Language.

Course Outcomes (COs)
Upon completion of this course, students will be able to:

1. Demonstrate an understanding of the elementary features of RDBMS
2. Design conceptual models of a database using ER modeling for real life applications
3. Analyze the existing design of a database schema and apply concepts of normalization to design an optimal database
4. Able to develop structured query language (SQL) queries to create, read, update, and delete relational database
5. Design efficient PL/SQL programs to access Oracle databases
6. Familiar with basic database storage structures and access techniques: file and page organizations, indexing methods including B tree, and hashing.

List Of Programs:

1. Create a table with following fields:
   Staff table:

Field name	Constraint	Type	Size
Staff_no	Primary key	Character	6
Staff_name	Not Null	Character	30
Gender	Check	Character	1
Dob		Date	
Dept_code	Foreign key	Character	4
Designation		Character	15

Basic		Number	7,2

Department table:

Field name	Constraint	Type	Size
Dept_code	Primary key	Character	4
Dept_name		Character	30

Execute the following queries:

- Insert records for all the staff.
- Select the records for male staff.
- Select the staff whose name start with S.
- Update the records.
- To list the staff who joined 2 years back.
- To list the staff in computer science dept.
- To list the staff_name and the dept_name in which he/she works.
- To list the maximum and minimum salary in each dept.
- To list the dept along with the total amount spent on salary
- To list the name of the staff who draw the salary more than the average salary.

2. Perform queries using string functions.
3. Perform queries using mathematical functions.
4. Perform queries using date functions.
5. Perform queries using aggregate and set functions.
6. Creation the following
   - Views
   - Objects
   - Sequence
   - Synonyms.
7. Create a table with the following fields:
   Book table:

Field name	Constraint	Type	Size
Access_no	Primary key	Character	6
Title		Character	30
Author		Character	30
Publisher		Character	30
Subject		Character	10
Price		Number	6,2

Execute the following queries:

- List the C and C++ books.
- List the books written by a particular author.
- List the books which costs between Rs.300/- and Rs.500/-

- List the no of books available in each subject.
- List the books in the decreasing order of the cost.
- Calculate the total cost and average cost of the available books
- List the books of minimum cost and maximum cost.

8. Perform different types of join operations between the two tables.

9. Create a table to store the salary details of the employees in a company. Use Cursor to update the employee salary.

10. Write a PL/SQL program to calculate the Fibonacci series, factorial, Palindrome string on user choice.

11. Create a table master book to contain the information of magazine_code, magazine_name, and publisher,Weekly/biweekly/monthly, price. Write PL/SQL block to perform insert, update and delete operations on the above table.

12. Create a table to contain phone number, user name, address of the phone user. Write a function to search for a address using phone numbers.

13. Create a table with the following fields:

Account table:

Field name	Constraint	Type	Size
Acc_no	Primary key	Character	4
Cust_name		Character	30
Branch_name		Character	30
Cust_city		Character	30

Borrower table:

Field name	Constraint	Type	Size
Acc_no	Foreign key	Character	30
Branch_name		Character	30
Amount		Number	8,2

Write the procedure to update the records of the tables.

14. Write a package to perform arithmetic operations.

15. Creating triggers.

TEXT BOOKS

1. Ramez Elmasri and Shamkant B. Navathe.2011. Fundamental of Database Systems.6th edition.Pearson India.[Unit I to V]

REFERENCE BOOKS

1. Ramez Elmasri.2013. Fundamentals of Database Systems: Models, Languages, Design and Application Programming , 6th edition Pearson India.

WEB SITES

http://en.wikipedia.org/wiki/RDBMS
http://aspalliance.com/1211_Relational_Database_Management_Systems__Concepts_and_Terminologies
www.compinfo-center.com/apps/rdbms.html

Semester – IV

19CAU412 SOFTWARE ENGINEERING - PRACTICAL 4H – 2C

Instruction Hours / week: L: 0 T: 0 P: 4 Marks: Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

Enable the student

- To introduce the fundamental concepts of software engineering.
- To Analyze, specify and document software requirements for a software system.
- To Design, select and apply the most appropriate software engineering process for a given project, plan for a software project, identify its scope and risks, and estimate its cost and time.
- Expose the criteria for test cases.
- Be familiar with test management and test automation techniques
- Implement a given software design using sound development practices.

Course Outcomes (COs)

Upon completion of this course the students will be able to:

1. Identify suitable life cycle models to be used and translate a requirement specification to a design using an appropriate software engineering methodology.
2. Apply systematic procedure for software design and deployment.
3. Analyze a problem and identify and define the computing requirements to the problem.
4. Formulate appropriate testing strategy for the given software system.
5. Develop software projects based on current technology, and test the software using testing tools.
6. An ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives.

Sample Projects:

1. **Criminal Record Management**: Implement a criminal record management system for jailers, police officers and CBI officers
2. **DTC Route Information**: Online information about the bus routes and their frequency and fares
3. **Car Pooling: To maintain a web based intranet application that enables the** corporate employees within an organization to avail the facility of carpooling effectively.
4. Patient Appointment and Prescription Management System
5. Organized Retail Shopping Management Software
6. Online Hotel Reservation Service System
7. Examination and Result computation system
8. Automatic Internal Assessment System

9. Parking Allocation System
10. Wholesale Management System

Suggested Readings

1. Bell,D., (2005). *Software Engineering for Students*, (4th ed.), Addison-Wesley.
2. Jalote,P., (2008). *An Integrated Approach to Software Engineering* (2nd ed.), New Age International Publishers.
3. Mall,R.,(2004). *Fundamentals of Software Engineering*, (2nd ed.), Prentice-Hall of India.
4. Pressman, R.S.,(2009). *Software Engineering: A Practitioner's Approach,* (7th Edition), McGraw-Hill.
5. Sommerville, I.,(2006), *Software Engineering*, (8th ed.), Addison Wesley.

Websites

1. http://en.wikipedia.org/wiki/Software_engineering
2. http://www.onesmartclick.com/engineering/software-engineering.html

http://www.CSU.gatech.edu/classes/AY2000/cs3802_fall

Semester – IV

19CAU413 WEB PROGRAMMING - PRACTICAL 4H – 2C

Instruction Hours / week: L: 0 T: 0 P: 4 Marks: Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objective

Enable the student

- To understand the concepts and architecture of the World Wide Web.
- To understand and practice markup languages
- Understand the technologies used in Web Programming.
- Know the importance of object-oriented aspects of Scripting.
- To understand and practice embedded dynamic scripting on client-side Internet Programming
- To develop dynamic web applications, create and consume web services

Course Outcomes (COs)

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

1. Design web pages.
2. Use technologies of Web Programming.
3. Apply object-oriented aspects to Scripting.
4. Create a basic website using HTML and Cascading Style Sheets.
5. Design and implement dynamic web page with validation using JavaScript objects and by applying different event handling mechanisms.
6. Use scripting languages and web services to transfer data and add interactive components to web pages.

List of Programs:

1. Develop a website for Karpagam University using HTML.
2. Write Online Quiz program (Include Style Sheets)
3. Create a simple animation using DHTML.
4. Write a program to apply Mask into an Image Using Filters in DHTML.
5. Generate web page that represents clock-every 60 see the page updated with server current time Using JavaScript.
6. Design a form and validate it using JavaScript.
7. Show the demo of AD Rotator Component
8. Write Database Access program using ASP.
9. Program to retrieve Cookies information using ASP
10. Program to count web page hits using ASP

11. Program to create Date & Time, String Manipulation using ASP
12. Write a program to find the visitor's Browser Type, IP Address and More Information

Suggested Readings

1. Danny Goodman, (2000). *Javascript Bible*, (3rd ed.), IDG Books India Pvt Ltd. **(Unit-III).**
2. David Flanagan, (2006). *JavaScript: The Definitive Guide*, O'Reilly,
3. Nicholas C. Zakas, Inc Ebrary and Ebrary,(2005). *Professional JavaScript for Web Developers*, New Delhi , John Wiley & Sons Inc.
4. Patrick Carey, (2005). *New Perspectives on HTML and XHTML*, (1st ed.), Thomson Course Technology Publishing. **(Unit- I).**
5. Rohit Khurana's, (2002). *Javascript Professional edition*, (2nd ed.), A.P.H. Publishing company, NewDelhi**.(Unit -II)**
6. Russell Jones, A. (2000). *Mastering ActiveServerPages 3*, **(**1st ed.), BPB Publishing, New Delhi.(**Unit- IV & Unit -V).**

Web Sites:

1. www.w3schools.com/
2. www.**javascript**kit.com
3. www.**asp**free.com
4. www.aspnettutorials.com

Semester – IV

19CAU414A R PROGRAMMING - PRACTICAL 3H – 1C

Instruction Hours / week: L: 0 T: 0 P: 3 Marks: Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objective

Enable the student

- To acquire the computing tasks such as using conditional processing statements, loops, and writing one's own functions.
- Performing advanced graphing of data and statistical modeling of data.
- Use statistical distribution functions in R
- Read Structured Data into R from various sources
- Understand split-apply-combine (group-wise operations) in R
- Perform basic statistical modeling of data

Course Outcomes (COs)

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

1. Learn how to install and configure software necessary for a statistical programming environment.
2. Discuss generic programming language concepts as they are implemented in a high-level statistical language.
3. The course covers practical issues in statistical computing which includes programming in R, reading data into R, accessing R packages, writing R functions, debugging, and organizing and commenting R code.
4. Import external data into R for data processing and statistical analysis
5. Learn the main R data structures – vector and data frame
6. Learn the file processing in R

List of Programs:

1. Write a program to demonstrate functions and operators
2. **Vectors:** Grouping values into vectors, then doing arithmetic and graphs with them
3. **Matrices:** Creating and graphing two-dimensional data sets
4. **Summary Statistics:** Calculating and plotting some basic statistics: mean, median, and standard deviation
5. **Factors:** Creating and plotting categorized data
6. **Data Frames:** Organizing values into data frames, loading frames from files and merging them
7. Write a program to design R as a calculator

7. Write a program to demonstrate Probability distributions
8. Write a program to demonstrate Importing and exporting data
9. Write a program to Establish a Regression

Suggested Readings

1. Daniel Navarro, (2013). *Learning Statistics with R*. University of Adelaide Publications.
2. Hadley Wickham, (2014). *Advanced R Programming*, (1st ed.)
3. Jeffrey Stanton, (2013). *Introduction to Data Science, with Introduction to R*, Version 3 ,
4. Roger.D.Peng, (2015). *R Programming for Data Science*

Semester –IV

19CAU414B OPEN SOURCE TECHNOLOGIES - PRACTICAL 3H – 1C

Instruction Hours / week: L: 0 T: 0 P: 3 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives:

Enable the student

- To introduce open source technology for development of web applications.
- For Study the problems with traditional commercial software.
- To understand open source scripting language for programming in web environment i.e. PHP.
- To study the open source management system and connection with database.
- To learn open source web server, software tools.
- To learn the open source ideals in order to apply those principles

Course Outcomes (COs)

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

1. Leaned the need of open source technology, open source development model, application of open sources, aspects of open source movement
2. The students will be aware about the problems with traditional commercial software.
3. The student will be familiar with basis syntax of PHP, common PHP scripts elements.
4. The student will be familiar with creating of the server side scripting using PHP, implement PHP database connectivity, perform operation on database and open source database management system.
5. The students will be familiar with Working of different web Servers.
6. The students will be aware about the software tool and process like Eclipse IDE, Selenium ID.

List of Programs:

1. Create ―Hello World application. That will display ―Hello World in the middle of the screen in the emulator. Also display ―Hello World in the middle of the screen in the Android Phone.
2. Create an application with login module. (Check username and password).
3. Create spinner with strings taken from resource folder and on changing the spinner value, Image will change.
4. Create a menu with 5 options and selected option should appear in text box.
5. Create a list of all courses in your college and on selecting a particular course teacher-in-charge of that course should appear at the bottom of the screen.

6. Create an application with three option buttons, on selecting a button colour of the screen will change.
7. Create and Login application as above. On successful login, pop up the message.
8. Create an application to Create, Insert, update, Delete and retrieve operation on the database.

Suggested Readings

1. Andrew M. St. Laurent, (2004). *Understanding Open Source and Free Software Licensing*, O'Reilly Media.
2. Dan Woods, Gautam Guliani, (2005). Open Source for the Enterprise, O'Reilly Media.
3. Fadi P. Deek and James A. M. McHugh,(2007). *Open Source Technology and Policy*, Cambridge University Press.
4. James Lee, Brent Ware,2002. *Open Source Web Development with LAMP: Using Linux, Apache, MySQL, Perl, and PHP*, , Publisher: Addison Wesley Date [UNIT IV – V]
5. Kailashvadera, Vhavyesh Gandhi, (2009). *Open source Technologies, Lakshmi Publications*,(1st ed.). [UNIT I – III]
6. Nick Wells,(2012). *The Complete Guide to Linux System Administration*, Delmar Cengage Learning.

Semester – V

19CAU501A ARTIFICIAL INTELLIGENCE 4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**
End Semester Exam: 3 Hours

Course Objectives

Enable the student

- To understand the various characteristics of Intelligent agents and different search strategies in AI
- To learn to represent knowledge in solving AI problems
- To learn about knowledge inference and planning strategies of AI.
- To know about various Expert System tools and applications
- To learn programing in logic language to implement various AI algorithms
- To read and analyzed important historical and current trends addressing artificial intelligence**.**

Course Outcomes (Cos)

Upon completion of this course the students will be able to:

1. Compare AI with human intelligence and traditional information processing, discuss its strengths, limitations and its application to complex and human centered problems.
2. Formalize a given problem in the language/framework of AI knowledge representation.
3. Analyze and formalize the problem through knowledge inference and planning strategies.
4. Demonstrate awareness and a fundamental understanding of various applications of AI techniques in machine learning and expert systems,
5. Apply the concept of AI using programming language PROLOG
6. Develop a basic understanding of the building blocks of AI as presented in terms of intelligent agents: Search, Knowledge representation, inference, logic, and learning.

Unit I – INTRODUCTION

Introduction to Artificial Intelligence, Background and Applications, Turing Test and Rational Agent approaches to AI, Introduction to Intelligent Agents, their structure, behavior and environment.

Unit II - PROBLEM SOLVING AND SEARCHING TECHNIQUES

Problem Characteristics, Production Systems, Control Strategies, Breadth First Search, Depth First Search, Hill climbing and its Variations, Heuristics Search Techniques: Best First Search, A* algorithm, Constraint Satisfaction Problem, Means-End Analysis, Introduction to Game Playing, Min-Max and Alpha-Beta pruning algorithms.

Unit III - KNOWLEDGE REPRESENTATION

Introduction to First Order Predicate Logic, Resolution Principle, Unification, Semantic Nets, Conceptual Dependencies, Frames, and Scripts, Production Rules, Conceptual Graphs. Programming in Logic (PROLOG)

Unit IV- DEALING WITH UNCERTAINTY AND INCONSISTENCIES

Truth Maintenance System, Default Reasoning, Probabilistic Reasoning, Bayesian Probabilistic Inference, Possible World Representations.

Unit V- UNDERSTANDING NATURAL LANGUAGES

Parsing Techniques, Context-Free and Transformational Grammars, Recursive and Augmented Transition Nets.

SUGGESTED READINGS

1. DAN.W. Patterson. Introduction to A.I and Expert Systems. PHI.2007.
2. Russell & Norvig. Artificial Intelligence-A Modern Approach. LPE. Pearson Prentice Hall.2nd edition. 2005.
3. Rich & Knight. Artificial Intelligence. Tata McGraw Hill. 2nd edition. 1991.
4. W.F. Clocksin and Mellish. Programming in PROLOG. Narosa Publishing House. 3rd edition. 2001.
5. Ivan Bratko. Prolog Programming for Artificial Intelligence. Addison-Wesley. Pearson Education. 3rd edition. 2000.

WEBSITES

1. https://artint.info/html/ArtInt_350.html
2. https://www.cleverism.com/artificial-intelligence-complete-guide/
3. https://search.credoreference.com/content/topic/artificial_intelligence

Semester – V

19CAU501B SOFTWARE TESTING 4H – 4C

Instruction Hours / week: L:3 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

Enable the student

- To introduce the fundamental concepts of software engineering.
- To Analyze, specify and document software requirements for a software system.
- To Design, select and apply the most appropriate software engineering process for a given project, plan for a software project, identify its scope and risks, and estimate its cost and time.
- Expose the criteria for test cases.
- Be familiar with test management and test automation techniques
- To gain confidence in and providing information about the level of quality of the software

Course Outcomes (COs)

Upon completion of this course the students will be able to:

1. Identify suitable life cycle models to be used and translate a requirement specification to a design using an appropriate software engineering methodology.
2. Apply systematic procedure for software design and deployment.
3. Analyze a problem and identify and define the computing requirements to the problem.
4. Formulate appropriate testing strategy for the given software system.
5. Develop software projects based on current technology, and test the software using testing tools.
6. Apply the software testing techniques in commercial environment

Unit- I

Assessing Testing Capabilities and Competencies

Who is associated with testing? – The multiple Roles of Testing – Defect – The business perspective for testing – test process and testers ; Building a software testing strategy: Computer System Strategic risks – Economics of Testing – Common Computer Problems – Economics of System Development Life Cycle (SDLC) Testing – Testing-an Organizational Issue – Establishing a testing policy – Structured approach to testing – Test Strategy; Software Testing Methodology: Reduce the cost of testing – Functional and Structural testing – Workbench concept – Eight considerations in developing testing methodologies

Unit -II

Software Testing Fundamentals

Examining the specification: Getting started – Performing a high-level review of the specification – Low-level specification test techniques; Testing the software with blinders on:

Dynamic Black-Box Testing- Test-to-Pass and Test-to-Fail- Equivalence Partitioning- Data testing – State testing – Other Black-box test techniques; Examining the code: Static White-Box testing- Formal reviews – Coding Standards and Guidelines- Generic Code Review Checklist; Testing the software with X-Ray glasses: Dynamic White-Box testing- Dynamic White-Box testing versus Debugging- Testing the Pieces- Data Coverage- Code Coverage.

Unit- III

Software Testing Techniques

Determining your software testing techniques: Testing Techniques/Tools selection process – Selecting Techniques/tools – Structural system testing techniques- Functional

system testing techniques – UNIT testing technique – Functional testing and analysis – Functional testing – Test factor/Test technique matrix

Testing process

The Cost of Computer Testing – Life Cycle Testing concept – Verification and validation in the software. Assess Project Management Development Estimate and Status - Develop Test Plan - Requirements Phase Testing -Design Phase Testing -Program Phase Testing

Unit -IV

Testing process

Execute Test and Record results- Acceptance Test- Report Test Results- Testing Software Installation- Test Software Changes - Evaluate Test Effectiveness.

Unit- V

Testing Specialized Systems and Applications

Testing client/server systems - Testing web-based systems - Testing security - Building tests documentation.

Suggested Readings

1. Ron Patton, (2004). *Software Testing*, (2nd ed.), New Delhi, Pearson Education. (2nd unit)

2. William E.Perry, (2001). *Effective methods for Software Testing*, (2nd ed.), New Delhi: John Wiley & Sons, Inc., (1, 3, 4 & 5 units)

Semester – V

19CAU502A COMPUTER GRAPHICS 4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**
End Semester Exam: 3 Hours

Course Objectives

Enable the student

- This course presents an introduction to computer graphics designed to give the student an overview of fundamental principles.
- The course makes the student to understand about the video and raster scan displays and their storage
- Methods for modeling objects as polygonal meshes or smooth surfaces, and as rendering such as hidden-surface removal, shading, illumination, and shadows will be investigated.
- To make the student to understand the usage of input devices and its working
- The course objective relies on the student to understand the line algorithm and 2D,3D Geometrical transformation.
- To Provide an understanding of mapping from a world coordinates to device coordinates, clipping, and projections.

Course Outcomes (COs)

Upon completion of the course, students will be able to

1. Have a knowledge and understanding of the structure of an interactive computer graphics system, and the separation of system components.
2. Have a knowledge and understanding of geometrical transformations and 2D viewing.
3. Be able to create interactive graphics applications.
4. Have a knowledge and understanding of techniques for representing 3D geometrical objects.
5. Have a knowledge and understanding of the various clipping algorithms and visible surface detection algorithm.
6. Implement various algorithms to scan, convert the basic geometrical primitives, transformations, Area filling, clipping.

Unit I - INTRODUCTION

Basic elements of Computer graphics, Applications of Computer Graphics.

Unit II - GRAPHICS HARDWARE

Architecture of Raster and Random scan display devices, input/output devices.

Unit III - FUNDAMENTAL TECHNIQUES IN GRAPHICS

Raster scan line, circle and ellipse drawing, thick primitives, Polygon filling, line and polygon clipping algorithms, 2D and 3D Geometric Transformations, 2D and 3D Viewing Transformations (Projections- Parallel and Perspective), Vanishing points.

Unit IV - GEOMETRIC MODELING

Representing curves & Surfaces.

Unit V - Visible Surface determination

Hidden surface elimination. **Surface rendering-** Illumination and shading models. Basic color models and Computer Animation.

SUGGESTED READINGS

1. J.D.Foley, A.Van Dan, Feiner, Hughes Computer Graphics Principles & Practice. (1990). 2nd Edition. Publication Addison Wesley.
2. D.Hearn, Baker. (2008). Computer Graphics. Prentice Hall of India. 2008.
3. D.F.Rogers.(1997).Procedural Elements for Computer Graphics. McGraw Hill.
4. D.F.Rogers. (1989). Adams Mathematical Elements for Computer Graphics. McGraw Hill 2nd Edition.

WEBSITES

1. https://w3.cs.jmu.edu/bernstdh/web/common/references/graphics.php
2. www.cs.kent.edu/~farrell/cg02/reference/
3. www.cs.brandeis.edu/~cs155/Intro_6.pdf

Semester – V

19CAU502B INFORMATION SECURITY AND CYBER LAWS 4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**
End Semester Exam: 3 Hours

Course Objectives

Enable the student

- To provides an overview of Information Security and Assurance.
- To provide an exposure to the spectrum of security activities methods methodologies and procedures with emphasis on practical aspects of Information Security.
- Gain knowledge about securing both clean and corrupted systems, protect personal data, and secure computer networks.
- Understand key terms and concepts in cyber law, intellectual property and cyber crimes, trademarks and domain theft.
- To examine secure software development practices.
- To incorporate approaches for incident analysis and response.
- To incorporate approaches for risk management and best practices.

Course Outcomes(COs)

A student who successfully completes this course should at a minimum be able to:

1. State the basic concepts in information security
2. Explain concepts related to applied cryptography including the four techniques for crypto-analysis symmetric and asymmetric cryptography, digital signature, message authentication code, hash functions and modes of encryption operations.
3. Explain common vulnerabilities in computer programs including buffer overflow Vulnerabilities time-of-check to time-of-use flaws incomplete mediation.
4. The learner will gain an understanding of cryptography, how it has evolved, and some key encryption techniques used today.
5. The learner will develop an understanding of security policies (such as confidentiality, integrity, and availability), as well as protocols to implement such policies.
6. The learner will gain familiarity with prevalent network and distributed system attacks, defenses against them, and forensics to investigate the aftermath.

Unit I - COURSE INTRODUCTION

Computer network as a threat, hardware vulnerability, software vulnerability, importance of data security.
Digital Crime: Overview of digital crime, criminology of computer crime.

Unit II - INFORMATION GATHERING TECHNIQUES

Tools of the attacker, information and cyber warfare, scanning and spoofing, password cracking, malicious software, session hijacking

Unit III - RISK ANALYSIS AND THREAT

Risk analysis, process, key principles of conventional computer security, security policies, authentication, data protection, access control, internal vs external threat, security assurance, passwords, authentication, and access control, computer forensics and incident response

Unit IV- INTRODUCTION TO CRYPTOGRAPHY AND APPLICATIONS

Important terms, Threat, Flaw, Vulnerability, Exploit, Attack, Ciphers, Codes, Caeser Cipher, Rail-Fence Cipher, Public key cryptography (Definitions only), Private key cryptography (Definition and Example)
Safety Tools and Issues : Firewalls, logging and intrusion detection systems, Windows and windows XP / NT security, Unix/Linux security, ethics of hacking and cracking

Unit V- CYBER LAWS

CYBER LAWS to be covered as per IT 2008:
• Chapter 1: Definitions 88
• Chapter 2: Digital Signature And Electronic Signature
• [Section 43] Penalty and Compensation for damage to computer, computer system, etc.
• [Section 65] Tampering with Computer Source Documents
• [Section 66 A] Punishment for sending offensive messages through communication service, etc.
• [Section 66 B] Punishments for dishonestly receiving stolen computer resource or communication device
• [Section 66C] Punishment for identity theft
• [Section 66D] Punishment for cheating by personation by using computer resource
• [Section 66E] Punishment for violation of privacy
• [Section 66F] Punishment for cyber terrorism
• [Section 67] Punishment for publishing or transmitting obscene material in electronic form
• [Section 67A] Punishment for publishing or transmitting of material containing sexually explicit act, etc. in electronic form[Section 67B] Punishment for publishing or transmitting of material depicting children in sexually explicit act, etc. in electronic form
• [Section 72] Breach of confidentiality and privacy

SUGGESTED READINGS

1. M. Merkow, J. Breithaupt. 2005. Information Security Principles and Practices. Pearson Education. 2005.
2. G.R.F. Snyder, T. Pardoe. 2010. Network Security. Cengage Learning.
3. A. Basta, W.Halton. 2008. Computer Security: Concepts, Issues and Implementation. Cengage Learning India.

WEB SITES

1. http://www.csc.ncsu.edu/faculty/ning
2. csrc.nist.gov/publications/nistpubs/800-12/handbook.pdf
3. www2.warwick.ac.uk/fac/sci/dcs/teaching/modules/cs134/

Semester – V

19CAU503A DATA MINING 4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

Enable the student

- To Understand Data Mining fundamentals and Characterize the kinds of patterns that can be discovered by association rule mining
- To Compare and evaluate different data mining techniques like classification, prediction.
- To Cluster the high dimensional data for better organization of the data
- To describe complex data types with respect to spatial and web mining
- To Design data warehouse with dimensional modelling and apply OLAP operations.
- To program using available data mining tools and general-purpose languages.

Course Outcomes(COs)

Upon completion of this course the students will be able to:

1. Extract knowledge using data mining techniques and Implement Preprocess the data for mining applications and apply the association rules for mining the data
2. Design and deploy appropriate classification techniques and decision trees.
3. Understand the concept of clustering and its real time applications
4. Explore recent trends in data mining such as web mining, spatial-temporal mining
5. Able to know the basic concepts of data warehouse and OLAP operations
6. Organize and Prepare the data needed for data mining using pre preprocessing techniques

Unit- I

Introduction : Fundamentals of data mining – Data Mining Functionalities – Classification of Data Mining systems – Major issues in Data Mining.

Data Warehouse and OLAP Technology: An Overview – Data Warehouse – Multidimensional Data Model – Data Warehouse Architecture

Unit-II

Data Preprocessing: Needs Preprocessing the Data – Data Cleaning – Data Integration and Transformation – Data Reduction – Discretization and Concept Hierarchy Generation – Online Data Storage.

Preparing Data for Mining: Variable Measures.

Unit-III

Mining Frequent Patterns, Associations and Correlations: Basic Concepts – Efficient and Scalable Frequent item set Mining Methods – From Association Mining to Correlation Analysis.

Unit-IV

Predictive and descriptive data mining techniques, supervised and unsupervised learning techniques, process of knowledge discovery in databases, pre-processing methods

Unit-V

Data Mining Techniques: Association Rule Mining, classification and regression techniques, clustering, Scalability and data management issues in data mining algorithms, measures of interestingness

Suggested Readings

1. Gupta, G.K., (2006). *Introduction to Data Mining with Case Studies*, PHI.
2. Pang-Ning Tan, Michael Steinbach, Vipin Kumar, (2005). *Introduction to Data Mining*, Pearson Education.
3. Richard Roiger, Michael Geatz, (2003). *Data Mining: A Tutorial Based Primer*, Pearson Education.
4. Soman,K.P., Diwakar Shyam, Ajay,V.,(2006*). Insight Into Data Mining: Theory And Practice*, PHI.

Semester – V

19CAU503B PROGRAMMING IN PYTHON 4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0 Marks: Internal: 40 Ext : 60 Total: 100

End Semester Exam: 3 Hours

Course Objectives

Enable the student

- To Learn Syntax and Semantics and create Functions in Python.
- To Understand the basic logic statements in Python
- To Handle Strings in Python.
- To Understand Lists, Dictionaries in Python.
- To Build GUI applications
- To Use Python interactively

Course Outcomes (COs)

Upon completion of this the course students will be able to:

1. Develop algorithmic solutions to simple computational problems
2. Structure simple Python programs for solving problems.
3. Decompose a Python program into functions.
4. Represent compound data using Python lists, tuples, dictionaries.
5. Read and write data from/to files in Python Programs.
6. Implement database and GUI applications

Unit-I

Planning the Computer Program: Concept of problem solving-Problem definition- Program design-Debugging-Types of errors in programming-Documentation.

Unit-II

Techniques of Problem Solving: Flowcharting-decision table-algorithms-Structured programming concepts-Programming methodologies: top-down and bottom-up Programming.

Unit-III

Overview of Programming: Structure of a Python Program-Elements of Python.

Unit-IV

Introduction to Python: Python Interpreter-Using Python as calculator-Python shell- Indentation. Atoms-Identifiers and keywords-Literals-Strings-Operators(Arithmetic operator, Relational operator, Logical or Boolean operator, Assignment, Operator, Ternary operator, Bit wise operator, Increment or Decrement operator).

Unit-V

Creating Python Programs: Input and Output Statements-Control statements(Branching, Looping, Conditional Statement, Exit function, Difference between break, continue and pass.). Defining Functions-Default arguments.

Suggested Readings

1. Allen Downey, Jeffrey Elkner, Chris Meyers, (2012). How to think like a computer scientist : learning with Python , Freely available online.
2. Budd,T.,(2011). *Exploring Python*, (1st ed.) TMH

Websites

1. http://docs.python.org/3/tutorial/index.html.
2. http://interactivepython.org/courselib/static/pythonds.
3. http://www.ibiblio.org/g2swap/byteofpython/read/.

Semester – V

19CAU504A DIGITAL IMAGE PROCESSING 3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

Enable the student

- To study the image fundamentals and mathematical transforms necessary for image processing.
- To study the image enhancement techniques
- To study image restoration procedures.
- To study the image compression procedures.
- To apply knowledge of mathematics, science, and engineering
- To design and conduct experiments, as well as to analyze and interpret data

Course Outcomes (COs)

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

1. Review the fundamental concepts of a digital image processing system.
2. Analyze images in the frequency domain using various transforms.
3. Evaluate the techniques for image enhancement and image restoration.
4. Categorize various compression techniques.
5. Interpret Image compression standards.
6. Interpret image segmentation and representation techniques.

Unit-I

Introduction: Light, Brightness adaption and discrimination, Pixels, coordinate conventions, Imaging Geometry, Perspective Projection, Spatial Domain Filtering, sampling and quantization. Spatial Domain Filtering: Intensity transformations, contrast stretching, histogram equalization, Correlation and convolution, Smoothing filters, sharpening filters, gradient and Laplacian.

Unit-II

Hotelling Transform, Fourier Transforms and properties, FFT (Decimation in Frequency and Decimation in Time Techniques), Convolution, Correlation, 2-D sampling, Discrete Cosine Transform, Frequency domain filtering.

Unit-III

Image Restoration, Basic Framework, Interactive Restoration, Image deformation and geometric transformations, image morphing, Restoration techniques, Noise characterization, Noise restoration filters, Adaptive filters, Linear, Position invariant degradations, Estimation of Degradation functions, Restoration from projections, Image Compression-Encoder-Decoder model, Types of redundancies, Lossy and Lossless compression, Entropy of an information source, Shannon's 1st Theorem, Huffman Coding, Arithmetic Coding, Golomb Coding, LZW

coding, Transform Coding, Sub-image size selection, blocking artifacts, DCT implementation using FFT, Run length coding.

Unit – IV

FAX compression (CSUITT Group-3 and Group-4), Symbol-based coding, JBIG-2, Bit-plane encoding, Bit-allocation, Zonal Coding, Threshold Coding, JPEG, Lossless predictive coding, Lossy predictive coding, Motion Compensation
Wavelet based Image Compression: Expansion of functions, Multi-resolution analysis, Scaling functions, MRA refinement equation, Wavelet series expansion, Discrete Wavelet Transform (DWT),Continuous Wavelet Transform, Fast Wavelet Transform, 2-D wavelet Transform, JPEG-2000 encoding, Digital Image Watermarking

Unit-V

Morphological Image Processing: Basics, SE, Erosion, Dilation, Opening, Closing, Hit-or-Miss Transform, Boundary Detection, Hole filling, Connected components, convex hull, thinning, thickening, skeletons, pruning, Geodesic Dilation, Erosion, Reconstruction by dilation and erosion. Image Segmentation: Boundary detection based techniques, Point, line detection, Edge detection, Edge linking, local processing, regional processing, Hough transform, Thresholding, Iterative thresholding, Otsu's method, Moving averages, Multivariable thresholding, Region-based segmentation, Watershed algorithm, Use of motion in segmentation

Suggested Readings

1. Castleman, K R., (1996). *Digital Image Processing*, Pearson Education.
2. Gonzalez, R C., Woods, R E.,(2008). *Digital Image Processing*, (3rd ed.)Pearson Education.
3. Jain,A.K.,(1989). *Fundamentals of Digital image Processing*, Prentice Hall of India.
4. Rafael C. Gonzalez, Richard E. Woods, Steven Eddins, (2004). *Digital Image Processing using* MATLAB', Pearson Education, Inc..
5. Schalkoff, (1989). *Digital Image Processing and Computer Vision*, John Wiley and Sons.

Semester – V

19CAU504B MONGO DB 3H – 3C

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Internal:**40** External:**60** Total: **100**

End Semester Exam : 3 Hours

Course Objectives

Enable the student

- To get knowledge and skills to master the NoSQL database mongoDB.
- To write programs using MongoDB
- To design an e-commerce data model
- To gain the knowledge on replication
- To know about Query language
- To know about query optimization

Course Outcomes(COs)

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

1. Gain the right skills and knowledge needed to develop Applications on mongoDB
2. Run Applications on MongoDB
3. Do the query operation in MongoDB
4. Manipulate aggregate function
5. Design an e-commerce data model
6. Gain the knowledge on replication

Unit I - GETTING STARTED

A database for the modern web – MongoDB through the JavaScript shell – Writing programs using MongoDB.

Unit II - APPLICATION DEVELOPMENT

Document-oriented data – Principles of schema design – Designing an e-commerce data model – Nuts and bolts on databases, collections, and documents. Queries and aggregation – E-commerce queries – MongoDB"s query language – Aggregating orders – Aggregation in detail.

Unit III - UPDATES, ATOMIC OPERATIONS, AND DELETES

A brief tour of document updates – E-commerce updates – Atomic document processing – MongoDB updates and deletes. Indexing and query optimization: Indexing theory – Indexing in practice – Query optimization.

Unit IV – REPLICATION

Overview – Replica sets – Master-slave replication – Drivers and replication. Shading: Overview – A sample shard cluster – Querying and indexing a shard cluster – Choosing a shard key – sharding in production.

Unit V - DEPLOYMENT AND ADMINISTRATION

Deployment – Monitoring and diagnostics – Maintenance – Performance troubleshooting

SUGGESTED READINGS

1. Kyle Banker. (2012). MongoDB in Action. Manning Publications Co.
2. Rick Copeland. (2013). MongoDB Applied Design Patterns, 1st Edition, O"Reilly Media Inc.
3. Gautam Rege, (2012). Ruby and MongoDB Web Development Beginner's Guide. Packt Publishing Ltd
4. Mike Wilson.. (2013). Building Node Applications with MongoDB and Backbone, O"Reilly Media Inc.
5. David Hows. (2009). The definitive guide to MongoDB, 2nd edition, Apress Publication, 8132230485
6. Shakuntala Gupta Edward. 2016. Practical Mongo DB , 2nd edition, Apress Publications, 2016, ISBN 1484206487

WEBSITES

1. http://www.mongodb.org/about/production-deployments/
2. http://docs.mongodb.org/ecosystem/drivers/
3. http://www.mongodb.org/about/applications/
4. http://www.mongodb.org/

Semester – V

19CAU511A ARTIFICIAL INTELLIGENCE – PRACTICAL 4H – 2C

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**
End Semester Exam: 3 Hours

Course Objectives

- To understand the various characteristics of Intelligent agents and different search strategies in AI
- To learn to represent knowledge in solving AI problems
- To learn about knowledge inference and planning strategies of AI.
- To know about various Expert System tools and applications
- To learn programing in logic language to implement various AI algorithms
- To read and analyzed important historical and current trends addressing artificial intelligence**.**

Course Outcomes (Cos)

Upon completion of this course the students will be able to:

1. Compare AI with human intelligence and traditional information processing, discuss its strengths, limitations and its application to complex and human centered problems.
2. Formalize a given problem in the language/framework of AI knowledge representation.
3. Analyze and formalize the problem through knowledge inference and planning strategies.
4. Demonstrate awareness and a fundamental understanding of various applications of AI techniques in machine learning and expert systems,
5. Apply the concept of AI using programming language PROLOG
6. Develop a basic understanding of the building blocks of AI as presented in terms of intelligent agents: Search, Knowledge representation, inference, logic, and learning.

List of Programs:

1. Write a prolog program to calculate the sum of two numbers.
2. Write a prolog program to find the maximum of two numbers.
3. Write a prolog program to calculate the factorial of a given number.
4. Write a prolog program to calculate the nth Fibonacci number.
5. Write a prolog program, insert_nth(item, n, into_list, result) that asserts that result is the list into_list with item inserted as the n‘th element into every list at all levels.
6. Write a Prolog program to remove the Nth item from a list.
7. Write a Prolog program, remove-nth(Before, After) that asserts the After list is the Before list with the removal of every n‘th item from every list at all levels.
8. Write a Prolog program to implement append for two lists.
9. Write a Prolog program to implement palindrome(List)
10. Write a Prolog program to implement max(X,Y,Max) so that Max is the greater of two numbers X and Y.
11. Write a Prolog program to implement maxlist(List,Max) so that Max is the greatest number in the list of numbers List.

12. Write a Prolog program to implement sumlist(List,Sum) so that Sum is the sum of a given list of numbers List.
13. Write a Prolog program to implement two predicates evenlength(List) and oddlength(List) so that they are true if their argument is a list of even or odd length respectively.
14. Write a Prolog program to implement reverse(List,ReversedList) that reverses lists.
15. Write a Prolog program to implement maxlist(List,Max) so that Max is the greatest number in the list of numbers List using cut predicate.
16. Write a Prolog program to implement GCD of two numbers.
17. Write a prolog program that implements Semantic Networks/Frame Structures.

SUGGESTED READINGS

1. DAN.W. Patterson. Introduction to A.I and Expert Systems. PHI.2007.
2. Russell & Norvig. Artificial Intelligence-A Modern Approach. LPE. Pearson Prentice Hall.2nd edition. 2005.
3. Rich & Knight. Artificial Intelligence. Tata McGraw Hill. 2nd edition. 1991.
4. W.F. Clocksin and Mellish. Programming in PROLOG. Narosa Publishing House. 3rd edition. 2001.
5. Ivan Bratko. Prolog Programming for Artificial Intelligence. Addison-Wesley. Pearson Education. 3rd edition. 2000.

WEBSITES

1. https://artint.info/html/ArtInt_350.html
2. https://www.cleverism.com/artificial-intelligence-complete-guide/
3. https://search.credoreference.com/content/topic/artificial_intelligence

Semester –V

19CAU511B SOFTWARE TESTING - PRACTICAL 4H – 2C

Instruction Hours / week: L: 0 T: 0 P: 3 Marks: Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

Enable the student

- To introduce the fundamental concepts of software engineering.
- To Analyze, specify and document software requirements for a software system.
- To Design, select and apply the most appropriate software engineering process for a given project, plan for a software project, identify its scope and risks, and estimate its cost and time.
- Expose the criteria for test cases.
- Be familiar with test management and test automation techniques
- To gain confidence in and providing information about the level of quality of the software

Course Outcomes (COs)

Upon completion of this course the students will be able to:

1. Identify suitable life cycle models to be used and translate a requirement specification to a design using an appropriate software engineering methodology.
2. Apply systematic procedure for software design and deployment.
3. Analyze a problem and identify and define the computing requirements to the problem.
4. Formulate appropriate testing strategy for the given software system.
5. Develop software projects based on current technology, and test the software using testing tools.
6. Apply the software testing techniques in commercial environment

List of Programs:

1. Create a VB form with the following fields and create the database also for them. Insert 3 records. Using Win Runner tool record the above 3 transaction and test them and produce the Report. (Blackbox Testing).
2. Create a VB form and then add login dialog form. Using Win Runner tool check the Username and Password and produce the Report. (Security testing).
3. Create a VB form with the following fields and check the calculation is correct or not by using the test toll Win Runner. (Functional Testing) Fields – Name, Designation, Department, Basic, HRA, DA, PF and netsal.
4. using Win Runner test tool check the database values after changing. Using Flight database. (Regression testing).
5. Write a C program for Boundary Testing.
6. Write a C program for Loop Testing.
7. Write a C program for Integration Testing.
8. Write a C program for Interface Testing.
9. Write a C program for Unit testing.

Suggested Readings

1. Ron Patton, (2004). *Software Testing*, (2nd ed.), New Delhi, Pearson Education. (2nd unit)

2. William E.Perry, (2001). *Effective methods for Software Testing*, (2nd ed.), New Delhi: John Wiley & Sons, Inc., (1, 3, 4 & 5 units)

Semester – V

19CAU512A COMPUTER GRAPHICS – PRACTICAL 4H – 2C

Instruction Hours / week: L: 0 T: 0 P: 4 Marks: Internal: **40** External: **60** Total: **100**
End Semester Exam: 3 Hours

Course Objectives

- This course presents an introduction to computer graphics designed to give the student an overview of fundamental principles.
- The course makes the student to understand about the video and raster scan displays and their storage
- Methods for modeling objects as polygonal meshes or smooth surfaces, and as rendering such as hidden-surface removal, shading, illumination, and shadows will be investigated.
- To make the student to understand the usage of input devices and its working
- The course objective relies on the student to understand the line algorithm and 2D,3D Geometrical transformation.
- To Provide an understanding of mapping from a world coordinates to device coordinates, clipping, and projections.

Course Outcomes (COs)

Upon completion of the course, students will be able to

1. Have a knowledge and understanding of the structure of an interactive computer graphics system, and the separation of system components.
2. Have a knowledge and understanding of geometrical transformations and 2D viewing.
3. Be able to create interactive graphics applications.
4. Have a knowledge and understanding of techniques for representing 3D geometrical objects.
5. Have a knowledge and understanding of the various clipping algorithms and visible surface detection algorithm.
6. Implement various algorithms to scan, convert the basic geometrical primitives, transformations, Area filling, clipping.

List of programs:

1. Write a program to implement Bresenham‘s line drawing algorithm.
2. Write a program to implement mid-point circle drawing algorithm.
3. Write a program to clip a line using Cohen and Sutherland line clipping algorithm.
4. Write a program to clip a polygon using Sutherland Hodgeman algorithm.
5. Write a program to apply various 2D transformations on a 2D object (use homogenous coordinates).
6. Write a program to apply various 3D transformations on a 3D object and then apply parallel and perspective projection on it.
7. Write a program to draw Hermite/Bezier curve.

SUGGESTED READINGS

1. J.D.Foley, A.Van Dan, Feiner, Hughes Computer Graphics Principles & Practice. (1990). 2nd Edition. Publication Addison Wesley.
2. D.Hearn, Baker. (2008). Computer Graphics. Prentice Hall of India. 2008.
3. D.F.Rogers.(1997).Procedural Elements for Computer Graphics. McGraw Hill.
4. D.F.Rogers. (1989). Adams Mathematical Elements for Computer Graphics. McGraw Hill 2nd Edition.

WEBSITES

1. https://w3.cs.jmu.edu/bernstdh/web/common/references/graphics.php
2. www.cs.brandeis.edu/~cs155/Intro_6.pdf

Semester – V

19CAU512B INFORMATION SECURITY AND CYBER LAWS - PRACTICAL 4H – 2C

Instruction Hours / week: L: 0 T: 0 P: 4 Marks: Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

- To provides an overview of Information Security and Assurance.
- To provide an exposure to the spectrum of security activities methods methodologies and procedures with emphasis on practical aspects of Information Security.
- Gain knowledge about securing both clean and corrupted systems, protect personal data, and secure computer networks.
- Understand key terms and concepts in cyber law, intellectual property and cyber crimes, trademarks and domain theft.
- To examine secure software development practices.
- To incorporate approaches for incident analysis and response.
- To incorporate approaches for risk management and best practices.

Course Outcomes(COs)

A student who successfully completes this course should at a minimum be able to:

1. State the basic concepts in information security
2. Explain concepts related to applied cryptography including the four techniques for crypto-analysis symmetric and asymmetric cryptography, digital signature, message authentication code, hash functions and modes of encryption operations.
3. Explain common vulnerabilities in computer programs including buffer overflow Vulnerabilities time-of-check to time-of-use flaws incomplete mediation.
4. The learner will gain an understanding of cryptography, how it has evolved, and some key encryption techniques used today.
5. The learner will develop an understanding of security policies (such as confidentiality, integrity, and availability), as well as protocols to implement such policies.
6. The learner will gain familiarity with prevalent network and distributed system attacks, defenses against them, and forensics to investigate the aftermath.

List of Programs:

1. Demonstrate the use of Network tools: ping, ipconfig, ifconfig, tracert, arp, netstat, whois
2. Use of Password cracking tools : John the Ripper, Ophcrack. Verify the strength of passwords using these tools.
3. Perform encryption and decryption of Caesar cipher. Write a script for performing these operations.
4. Perform encryption and decryption of a Rail fence cipher. Write a script for performing these operations.
5. Use nmap/zenmap to analyse a remote machine.
6. Use Burp proxy to capture and modify the message.

7. Demonstrate sending of a protected word document.
8. Demonstrate sending of a digitally signed document.
9. Demonstrate sending of a protected worksheet.
10. Demonstrate use of steganography tools.
11. Demonstrate use of gpg utility for signing and encrypting purposes.

SUGGESTED READINGS

1. M. Merkow, J. Breithaupt. 2005. Information Security Principles and Practices. Pearson Education. 2005.
2. G.R.F. Snyder, T. Pardoe. 2010. Network Security. Cengage Learning.
3. A. Basta, W.Halton. 2008. Computer Security: Concepts, Issues and Implementation. Cengage Learning India.

WEB SITES

1. http://www.csc.ncsu.edu/faculty/ning
2. csrc.nist.gov/publications/nistpubs/800-12/handbook.pdf

Semester – V

19CAU513A DATA MINING - PRACTICAL 4H – 2C

Instruction Hours / week: L: 0 T: 0 P: 4 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

- To Understand Data Mining fundamentals and Characterize the kinds of patterns that can be discovered by association rule mining
- To Compare and evaluate different data mining techniques like classification, prediction.
- To Cluster the high dimensional data for better organization of the data
- To describe complex data types with respect to spatial and web mining
- To Design data warehouse with dimensional modelling and apply OLAP operations.
- To program using available data mining tools and general-purpose languages.

Course Outcomes(COs)

Upon completion of this course the students will be able to:

1. Extract knowledge using data mining techniques and Implement Preprocess the data for mining applications and apply the association rules for mining the data
2. Design and deploy appropriate classification techniques and decision trees.
3. Understand the concept of clustering and its real time applications
4. Explore recent trends in data mining such as web mining, spatial-temporal mining
5. Able to know the basic concepts of data warehouse and OLAP operations
6. Organize and Prepare the data needed for data mining using pre preprocessing techniques

List of Programs:

1. Use the following learning schemes, with the default settings to analyze the weather data (in weather.arff). For test options, first choose "Use training set", then choose "Percentage Split" using default 66% percentage split. Report model percent error rate
2. Using iris dataset preprocess and classify it with J4.8 and Naïve Bayes Classifier. examine the tree in the Classifier output panel
3. Using the datasets *ReutersCorn-Train* and *ReutersGrain-Train*. Classify articles using binary attributes and word count attributes.
4. Apply any two association rule based algorithm for the supermarket analysis
5. Using weka Experimenter perform comparison analysis of J48, oneR and ID3 for vote dataset
6. Using Weka Experimenter perform comparison analysis of Naïve Bayes with different datasets
7. Apply ZeroR, OneR, and J48, to classify the Iris data in an experiment using 10 train and test runs, with 66% of the data used for training and 34% used for testing.
8. Using Weka Knowledge flow Sett up a flow to load an ARFF file (batch mode) and perform a cross-validation using J48 (WEKA's C4.5 implementation).

Suggested Readings

1. Gupta, G.K., (2006). *Introduction to Data Mining with Case Studies*, PHI.
2. Pang-Ning Tan, Michael Steinbach, Vipin Kumar, (2005). *Introduction to Data Mining*, Pearson Education.
3. Richard Roiger, Michael Geatz, (2003). *Data Mining: A Tutorial Based Primer*, Pearson Education.
4. Soman,K.P., Diwakar Shyam, Ajay,V.,(2006*). Insight Into Data Mining: Theory And Practice*, PHI.

Semester – V

19CAU513B PROGRAMMING IN PYTHON - PRACTICAL 4H – 2C

Instruction Hours / week: L: 0 T: 0 P: 4 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

- To Learn Syntax and Semantics and create Functions in Python.
- To Understand the basic logic statements in Python
- To Handle Strings in Python.
- To Understand Lists, Dictionaries in Python.
- To Build GUI applications
- To Use Python interactively

Course Outcomes (COs)

Upon completion of this the course students will be able to:

1. Develop algorithmic solutions to simple computational problems
2. Structure simple Python programs for solving problems.
3. Decompose a Python program into functions.
4. Represent compound data using Python lists, tuples, dictionaries.
5. Read and write data from/to files in Python Programs.
6. Implement database and GUI applications

List of Programs:

1. Write a menu driven program to convert the given temperature from Fahrenheit to Celsius and vice versa depending upon user's choice.
2. WAP to calculate total marks, percentage and grade of a student. Marks obtained in each of the three subjects are to be input by the user. Assign grades according to the following criteria :
   Grade A: Percentage >=80
   Grade B: Percentage>=70 and <80
   Grade C: Percentage>=60 and <70
   Grade D: Percentage>=40 and <60
   Grade E: Percentage<40
3. Write a menu-driven program, using user-defined functions to find the area of rectangle, square, circle and triangle by accepting suitable input parameters from user.
4. Write a program in python to display the first n terms of Fibonacci series.
5. Write a program in python to find factorial of the given number.
6. Write a program in python to find sum of the following series for n terms: 1 – 2/2! + 3/3! - - - - - - n/n!
7. Write a program in python to calculate the sum and product of two compatible matrices.

Suggested Readings

1. Allen Downey, Jeffrey Elkner, Chris Meyers, (2012). How to think like a computer scientist : learning with Python , Freely available online.
2. Budd,T.,(2011). *Exploring Python*, (1st ed.) TMH

Websites

1. http://docs.python.org/3/tutorial/index.html.
2. http://interactivepython.org/courselib/static/pythonds.
3. http://www.ibiblio.org/g2swap/byteofpython/read/.

Semester – V

19CAU514A DIGITAL IMAGE PROCESSING - PRACTICAL 3H – 1C

Instruction Hours / week: L: 0 T: 0 P: 3 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

- To study the image fundamentals and mathematical transforms necessary for image processing.
- To study the image enhancement techniques
- To study image restoration procedures.
- To study the image compression procedures.
- To apply knowledge of mathematics, science, and engineering
- To design and conduct experiments, as well as to analyze and interpret data

Course Outcomes (COs)

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

1. Review the fundamental concepts of a digital image processing system.
2. Analyze images in the frequency domain using various transforms.
3. Evaluate the techniques for image enhancement and image restoration.
4. Categorize various compression techniques.
5. Interpret Image compression standards.
6. Interpret image segmentation and representation techniques

List of Programs:

1. Write program to read and display digital image using MATLAB or SCILAB
   a. Become familiar with SCILAB/MATLAB Basic commands
   b. Read and display image in SCILAB/MATLAB
   c. Resize given image
   d. Convert given color image into gray-scale image
   e. Convert given color/gray-scale image into black & white image
   f. Draw image profile
   g. Separate color image in three R G & B planes
   h. Create color image using R, G and B three separate planes
   i. Flow control and LOOP in SCILAB
   j. Write given 2-D data in image file
2. To write and execute image processing programs using point processing method
   a. Obtain Negative image b. Obtain Flip image
   b. Thresholding d. Contrast stretching
3. To write and execute programs for image arithmetic operations
   a. Addition of two images
   b. Subtract one image from other image
   c. Calculate mean value of image
   d. Different Brightness by changing mean value
4. To write and execute programs for image logical operations

a. AND operation between two images
b. OR operation between two images
c. Calculate intersection of two images
d. Water Marking using EX-OR operation
e. NOT operation (Negative image)

5. To write a program for histogram calculation and equalization using
   a. Standard MATLAB function
   b. Program without using standard MATLAB functions
   c. C Program
6. To write and execute program for geometric transformation of image
   a. Translation
   b. Scaling
   c. Rotation
   d. Shrinking
   e. Zooming
7. To understand various image noise models and to write programs for
   a. image restoration
   b. Remove Salt and Pepper Noise
   c. Minimize Gaussian noise
   d. Median filter and Weiner filter
8. Write and execute programs to remove noise using spatial filters
   a. Understand 1-D and 2-D convolution process
   b. Use 3x3 Mask for low pass filter and high pass filter
   c. To write and execute program for wavelet transform on given image and perform inverse wavelet transform to reconstruct image.
9. Write and execute programs for image frequency domain filtering
   a. Apply FFT on given image
   b. Perform low pass and high pass filtering in frequency domain
   c. Apply IFFT to reconstruct image
10. Write a program in C and MATLAB/SCILAB for edge detection using different edge detection mask
11. Write and execute program for image morphological operations erosion and dilation.

Suggested Readings

1. Castleman, K R., (1996). *Digital Image Processing*, Pearson Education.
2. Gonzalez, R C., Woods, R E.,(2008). *Digital Image Processing*, (3rd ed.)Pearson Education.
3. Jain,A.K.,(1989). *Fundamentals of Digital image Processing*, Prentice Hall of India.
4. Rafael C. Gonzalez, Richard E. Woods, Steven Eddins, (2004). *Digital Image Processing using* MATLAB', Pearson Education, Inc..
5. Schalkoff, (1989). *Digital Image Processing and Computer Vision*, John Wiley and Sons.

Semester – V

19CAU514B MONGO DB - PRACTICAL 3H – 1C

Instruction Hours / week: L: 0 T: 0 P: 4 **Marks:** Internal:**40** External:**60** Total: **100**

End Semester Exam : 3 Hours

Course Objectives

Enable the student

- To get knowledge and skills to master the NoSQL database mongoDB.
- To write programs using MongoDB
- To design an e-commerce data model
- To gain the knowledge on replication
- To know about Query language
- To know about query optimization

Course Outcomes(COs)

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

1. Gain the right skills and knowledge needed to develop Applications on mongoDB
2. Run Applications on MongoDB
3. Do the query operation in MongoDB
4. Manipulate aggregate function
5. Design an e-commerce data model
6. Gain the knowledge on replication

List of Experiments:

Structure of 'restaurants' collection :

{ "address": { "building": "1007", "coord": [-73.856077, 40.848447], "street": "Morris Park Ave", "zipcode": "10462" }, "borough": "Bronx", "cuisine": "Bakery", "grades": [{ "date": { "$date": 1393804800000 }, "grade": "A", "score": 2 }, { "date": { "$date": 1378857600000 }, "grade": "A", "score": 6 }, { "date": { "$date": 1358985600000 }, "grade": "A", "score": 10 }, { "date": { "$date": 1322006400000 }, "grade": "A", "score": 9 }, { "date": { "$date": 1299715200000 }, "grade": "B", "score": 14 }, "name": "Morris Park Bake Shop", "restaurant_id": "30075445" }

1. Write a MongoDB query
    a. To display all the documents in the collection restaurants.
    b. To display the fields restaurant_id, name, borough and cuisine for all the documents in the collection restaurant.
    c. To display the fields restaurant_id, name, borough and cuisine, but exclude the field _id for all the documents in the collection restaurant
    d. To display the fields restaurant_id, name, borough and zip code, but exclude the field _id for all the documents in the collection restaurant.
    e. To display all the restaurant which is in the borough bronx
    f. To display the first 5 restaurant which is in the borough bronx.

g. To display the next 5 restaurants after skipping first 5 which are in the borough bronx.
h. To find the restaurants who achieved a score more than 90.
i. To Find The Restaurants That Achieved A Score, More Than 80 But Less Than 100.

2. Write a MongoDB query
   a. To find the restaurants which locate in latitude value less than -95.754168.
   b. To Find the restaurants that do not prepare any cuisine of 'american' and their grade score more than 70 and latitude less than -65.754168.
   c. To Find the restaurants which do not prepare any cuisine of 'american' and achieved a score more than 70 and not located in the longitude less than - 65.754168. Note : do this query without using $and operator. Go to the editor
   d. To find the restaurants which do not prepare any cuisine of 'American ' and achieved a grade point 'A' not belongs to the borough Brooklyn. The document must be displayed according to the cuisine in descending order.
3. Write a MongoDB query
   a. To find the restaurant Id, name, borough and cuisine for those restaurants which contain 'Wil' as first three letters for its name. Go to the editor
   b. To find the restaurant Id, name, borough and cuisine for those restaurants which contain 'ces' as last three letters for its name.
   c. To find the restaurant Id, name, borough and cuisine for those restaurants which contain 'Reg' as three letters somewhere in its name.
4. Write a mongodb query
   a. To find the restaurants which belong to the borough Bronx and prepared either American or Chinese dish.
   b. To find the restaurant Id, name, borough and cuisine for those restaurants which belong to the borough Staten Island or Queens or Bronxor Brooklyn.
   c. To find the restaurant Id, name, borough and cuisine for those restaurants which are not belonging to the borough Staten Island or Queens or Bronxor Brooklyn.
   d. To find the restaurant Id, name, borough and cuisine for those restaurants which achieved a score which is not more than 10.
   e. To find the restaurant Id, name, borough and cuisine for those restaurants which prepared dish except 'American' and 'Chinees' or restaurant's name begins with letter 'Wil'.
   f. To find the restaurant Id, name, and grades for those restaurants which achieved a grade of "A" and scored 11 on an isodate "2014-08- 11T00:00:00Z" among many of survey dates
   g. To find the restaurant Id, name and grades for those restaurants where the 2nd element of grades array contains a grade of "A" and score 9 on an isodate "2014-08-11T00:00:00Z".
5. Write a MongoDB query to find the restaurant Id, name, address and geographical location for those restaurants where 2nd element of coord array contains a value which is more than 42 and upto 52
6. Write a MongoDB query
   a. To arrange the name of the restaurants in descending along with all the columns.
   b. To arranged the name of the cuisine in ascending order and for that same cuisine borough should be in descending order.
7. Write a MongoDB query to know whether all the addresses contains the street or not.

8. Write a MongoDB query which will select all documents in the restaurants collection where the coord field value is Double.
9. Write a MongoDB query which will select the restaurant Id, name and grades for those restaurants which returns 0 as a remainder after dividing the score by 7.
10. Write a MongoDB query to find the restaurant name, borough, longitude and attitude and cuisine for those restaurants which contains 'mon' as three letters somewhere in its name.

SUGGESTED READINGS

1. Kyle Banker. (2012). MongoDB in Action. Manning Publications Co.

2. Rick Copeland. (2013). MongoDB Applied Design Patterns, 1st Edition, O"Reilly Media Inc.
3. Gautam Rege, (2012). Ruby and MongoDB Web Development Beginner's Guide. Packt Publishing Ltd
4. Mike Wilson.. (2013). Building Node Applications with MongoDB and Backbone, O"Reilly Media Inc.
5. David Hows. (2009). The definitive guide to MongoDB, 2nd edition, Apress Publication, 8132230485
6. Shakuntala Gupta Edward. 2016. Practical Mongo DB , 2nd edition, Apress Publications, 2016, ISBN 1484206487

WEBSITES

1. http://www.mongodb.org/about/production-deployments/
2. http://docs.mongodb.org/ecosystem/drivers/
3. http://www.mongodb.org/about/applications/
4. http://www.mongodb.org/

Semester – VI

19CAU601A PHP PROGRAMMING 4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

Enable the student

- To write basic PHP syntax using various operators.
- To write PHP scripts to handle HTML forms.
- To analyze different tasks using PHP functions.
- To understand the regular expressions in PHP.
- To learn array data structure using PHP scripts.
- To Create conditional structures

Course Outcomes (COs)

Upon completion of this course, students will be able to:

1. Write PHP scripts using operators to perform various functions
2. Design PHP scripts to handle HTML forms.
3. Implement different types of PHP functions.
4. Write regular expressions including modifiers, operators, and metacharacters.
5. Create PHP scripts using array.
6. Analyze and solve various database tasks using the PHP language.

Unit-I

Introduction to PHP: PHP introduction, inventions and versions, important tools and software requirements (like Web Server, Database, Editors etc.) -PHP with other technologies, scope of PHP -Basic Syntax, PHP variables and constants -Types of data in PHP , Expressions, scopes of a variable (local, global)-PHP Operators: Arithmetic, Assignment, Relational, Logical operators, Bitwise , ternary and MOD operator-PHP operator Precedence and associativity

Unit-II

Handling HTML form with PHP: Capturing Form Data-GET and POST form methods-Dealing with multi value fields Redirecting a form after submission -**PHP conditional events and Loops:** PHP IF Else conditional statements (Nested IF and Else) -Switch case, while ,For and Do While Loop -Goto , Break ,Continue and exit

Unit-III

PHP Functions: Function, Need of Function , declaration and calling of a function -PHP Function with arguments, Default Arguments in Function -Function argument with call by value, call by reference -Scope of Function Global and Local

Unit-IV

String Manipulation and Regular Expression: Creating and accessing String, Searching & Replacing String -Formatting, joining and splitting String , String Related Library functions-Use and advantage of regular expression over inbuilt function -Use of preg_match(), preg_replace(), preg_split()-functions in regular expression

Unit-V

Array: Anatomy of an Array ,Creating index based and Associative array ,Accessing array-Looping with Index based array, with associative array using each() and foreach()-Some useful Library function

Suggested Readings

1. David Sklar, Adam Trachtenberg, (2014). *PHP Cookbook: Solutions & Examples for PHP*.
2. Luke Welling, Laura Thompson,(2008). *PHP and MySQL Web Development*, (4th ed.), Addition Paperback, Addison-Wesley Professsional.
3. Robin Nixon,(2014). *Learning PHP, MySQL, JavaScript, CSS & HTML5, (3rd ed.) Paperback*, O'reilly.
4. Steven Holzner, (2007). *PHP: The Complete Reference Paperback*, McGraw Hill Education (India), 2007.
5. Timothy Boronczyk, Martin E. Psinas, (2008). *PHP and MYSQL (Create-Modify-Reuse)*, Wiley India Private Limited.

Semester – VI

19CAU601B UNIX / LINUX PROGRAMMING 4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

Enable the student to

- Understand fundamental operating system abstractions such as processes, threads, files, semaphores, IPC abstractions, shared memory regions, etc.
- Understand how the operating system abstractions can be used in the development of application programs, or to build higher level abstractions
- Understand how the operating system abstractions can be implemented
- Understand the principles of concurrency and synchronization, and apply them to write correct concurrent programs/software
- Understand basic resource management techniques (scheduling or time management, space management) and principles and how they can be implemented. These also include issues of performance and fairness objectives, avoiding deadlocks, as well as security and protection.
- provide introduction to UNIX Operating System and its File System
- gain an understanding of important aspects related to the SHELL and the process
- develop the ability to formulate regular expressions and use them for pattern matching.

Course Outcome

Upon completion of this course, students will be able to:

1. Describe the architecture and features of UNIX Operating System and distinguish it from other Operating System
2. Demonstrate UNIX commands for file handling and process control
3. Write Regular expressions for pattern matching and apply them to various filters for a specific task
4. Analyze a given problem and apply requisite facets of SHELL programming in order to devise a SHELL script to solve the problem
5. Implement various file processing commands used in UNIX.
6. Construct various shell scripts for simple applications.

Unit-I

Introduction What is Linux/Unix Operating systems, Difference between linux/unix and other operating systems , Features and Architecture, Various Distributions available in the market, Installation, Booting and shutdown process

Unit-II

System processes (an overview), External and internal commands, Creation of partitions in OS, Processes and its creation phases – Fork, Exec, wait

Unit-III

User Management and the File System Types of Users, Creating users, Granting rights
User management commands, File quota and various file systems available, File System Management and Layout, File permissions, Login process, Managing Disk Quotas, Links (hard links, symbolic links)

Unit-IV

Shell introduction and Shell Scripting What is shell and various type of shell, Various editors present in Linux Different modes of operation in vi editor, What is shell script, Writing and executing the shell script , Shell variable (user defined and system variables)

Unit-V

System calls, Using system calls Pipes and Filters, Decision making in Shell Scripts (If else, switch), Loops in shell, Functions, Utility programs (cut, paste, join, tr, uniq utilities), Pattern matching utility (grep)

Suggested Readings

1. Michael Jang, (2011). *RHCSA/ RHCE Red Hat Linux Certification: Exams (Ex200 & Ex300)* ,Certification Press.
2. Nemeth Synder & Hein,(2010). *Linux Administration Handbook*, (2nd ed.) Pearson Education
3. Richard Stevens,W., Bill Fenner, Andrew M. Rudoff, (2014). *Unix Network Programming, The sockets Networking*, Vol. 1, 3rd ed.) API.
4. Sumitabha, Das, (2006). *Unix Concepts And Applications*, Tata McGraw-Hill Education.

SEMESTER-VI

19CAU602A DATABASE ADMINISTRATION 4H - 4C

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

Enable the student to

- Installing Oracle Software
- Creating an Oracle Database Using DBCA
- Managing Database instances and ASM instances
- Managing and controlling database network environment
- Define and devise transaction management, concurrency control, crash recovery components
- Managing storage structures
- Controlling user security
- Designing Database backup and recovery procedures
- Take Decisions related with Database Maintenance

Course Outcomes (COs)

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

1. Design, model and install any database management systems by using Oracle database as sample.
2. Plan, design, construct, control and manage database instances, database network environment, storage structures, user security, database backup and recovery, database maintenance
3. Define and devise transaction management, concurrency control, crash recovery components
4. Examine and perform data base administration roles and operations by using Oracle database system as a sample
5. Compare and contrast by examining the database systems and new trends in data storage, data retrieval and maintenance techniques.
6. Configure, manage and maintain database, to audit and improve database performances and to use tools for database administration.

Unit I

Oracle DBA's: The Oracle DBA's Role- Oracle Database 11g Architecture: Oracle Databases and instances- Oracle Logical Storage structures – Oracle Logical Database structures – Oracle Physical Storage structures- Multiplexing Database Files - Oracle Memory Structures-Oracle Backup and Recovery – Security Capabilities – Tablespace Architecture – Oracle Tablespace installation – Traditional Disk Space Storage – Automatic Storage Management

Unit II

Common Space Management Problems – Oracle Segments, Extents and Blocks – Space Management Methodologies – SYSAUX monitoring and usage – Archived Redo Log File Management – Built in Space Management Tools: Segment Advisor – Undo Advisor and the Automatic Workload Repository – Index usage – Space Usage Warning Levels – Reusable space allocation – Managing alert and Trace Files with ADR – Transaction Basics – Undo Basics – Managing Undo Tablespaces – Flashback features

Unit III

Tuning Application Design – Tuning SQL – Tuning Memory Usage – Tuning Data Access – Tuning Data Manipulation – Tuning Physical Storage – Reducing Network Security – Database Authentication Methods

Unit IV

Database Authorization Methods – Auditing: Auditing Locations – Statement Auditing – Privilege Auditing Schema Object Auditing – Auditing Related Data Dictionary Views – Logical Backups – Physical Backups – Using Data Pump Export and Import – Data Pump Import Options – Integration of Backup Procedures
Overview of Oracle Net – Using the Oracle Net Configuration Assistant – Using the Oracle Net Manager – Starting the Listener Server Process – Controlling the Listener Server Process Using Data links

Unit-V

Creating Tablespaces in a VLDB Environment: Bigfile Tablespace Basics – Creating and Modifying Bigfile Tablespace – Bigfile Tablespace ROWID format – DBMS_ROWID and Bigfile Tablespaces.- Advanced Oracle Table Types – Using Bitmap Indexes – Oracle Data Pump
Remote queries – Remote Data Manipulation: Two Phase Commit – Managing Distributed Data – Managing Distributed Transactions – Monitoring and Tuning Distributed Database

Suggested Readings

1. Bob Bryla, Kevin Loney 2008 Oracle Database 11g DBA Handbook McGraw-Hill Osborne
2. Saikat Basak. 2010. Oracle DBA Concise Handbook ,Ensel Software

Websites :

1. www.oracle.com/technology/software/products/database/oracle10g/index.html
2. www.oracle-base.com/articles/10g/
3. www.adp-gmbh.ch/ora/misc/10g.html

Semester – VI

19CAU602B CLOUD COMPUTING 4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0 **Marks:** Int: **40** Ext: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

Enable the student to

- Basics of cloud computing.
- Key concepts of virtualization.
- Different Cloud Computing services
- Cloud Implementation, Programming and Mobile cloud computing
- Key components of Amazon Web Services
- Cloud Backup and solution

Course Outcomes (COs)

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

1. Define Cloud Computing and memorize the different Cloud service and deployment models
2. Describe importance of virtualization along with their technologies.
3. Use and Examine different cloud computing services
4. Analyze the components of open stack & Google Cloud platform and understand Mobile Cloud Computing
5. Describe the key components of Amazon web Service
6. Design & develop backup strategies for cloud data based on features

Unit-I

Overview of Computing Paradigm: Recent trends in Computing: Grid Computing, Cluster Computing, Distributed Computing, Utility Computing, Cloud Computing. **Introduction to Cloud Computing:** Introduction to Cloud Computing, History of Cloud Computing, Cloud service providers, Benefits and limitations of Cloud Computing.

Unit-II

Cloud Computing Architecture: Comparison with traditional computing architecture (client/server), Services provided at various levels, Service Models- Infrastructure as a Service(IaaS), Platform as a Service(PaaS), Software as a Service(SaaS), How Cloud Computing Works, Deployment Models- Public cloud, Private cloud, Hybrid cloud, Community cloud, Case study of NIST architecture.

Unit-III

Case Studies: Case study of Service model using Google App Engine, Microsoft Azure, Amazon EC2 , Eucalyptus.

Unit-IV

Service Management in Cloud Computing: Service Level Agreements (SLAs), Billing & Accounting, Comparing Scaling Hardware: Traditional vs. Cloud, Economics of scaling.

Unit-V

Cloud Security: Infrastructure Security- Network level security, Host level security, Application level security, Data security and Storage- Data privacy and security Issues, Jurisdictional issues raised by Data location, Authentication in cloud computing.

Suggested Readings

1. Barrie Sosinsky,(2010). *Cloud Computing Bible*,Wiley-India.
2. Dimitris N. Chorafas,(2010). Cloud Computing Strategies ,CRC Press.
3. Gautam Shroff, (2010). *Enterprise Cloud Computing Technology Architecture Application*s, Cambridge University Press.
4. Gautam Shroff, (2010). Enterprise Cloud Computing Technology Architecture Applications, Cambridge University Press.
5. Nikos Antonopoulos, Lee Gillam,(2012). *Cloud Computing: Principles, Systems and Applications*, Springer Publications.
6. Ronald L. Krutz, Russell Dean Vines,(2010). Cloud Security: A Comprehensive Guide to Secure Cloud Computing, , Wiley-India.
7. Toby Velte, Anthony Velte, Robert Elsenpeter,(2010). Cloud Computing, A Practical Approach ,McGraw Hills.

Websites

1. en.wikipedia.org/wiki/Cloud_computing
2. www.ibm.com/cloud-computing/in/en/
3. www.oracle.com/CloudComputing
4. www.microsoft.com/en-us/cloud/default.aspx

SEMESTER-VI

19CAU603A BIG DATA ANALYTICS 3H- 3C

Instruction Hours / week: L: 3 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3Hours

Course Objectives

Enable the student

- To know the fundamental concepts of big data and analytics.
- To explore tools and practices for working with big data
- To learn about stream computing.
- To know about the research that requires the integration of large amounts of data.
- To introduce the tools required to manage and analyze big data like Hadoop, NoSql MapReduce.
- To teach the fundamental techniques and principles in achieving big data analytics with scalability and streaming capability

Course Outcomes (COs)

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

1. Work with big data tools and its analysis techniques
2. Analyze data by utilizing clustering and classification algorithms
3. Learn and apply different mining algorithms and recommendation systems for large volumes of data
4. Perform analytics on data streams
5. Learn NoSQL databases and management.
6. Understand the key issues in big data management and its associated applications in intelligent business and scientific computing

Unit-I

Fundamentals of Big Data - The Evolution of Data Management Understanding the Waves of Managing Data- Defining Big Data - Big Data Management Architecture- The Big Data Journey -Big Data Types-Defining Structured Data-Defining Unstructured Data-Putting Big Data Together.

Unit-II

Big Data Stack- Basics of Virtualization - The importance of virtualization to big data -Server virtualization - Application virtualization - Network virtualization -Processor and memory virtualization - Data and storage virtualization-Abstraction and Virtualization-Implementing Virtualization to Work with Big Data.

Unit-III

Hadoop - Hadoop Distributed File System - Hadoop MapReduce- The Hadoop foundation and Ecosystem.

Unit-IV

Big Data Analytics-Text Analytics and Big Data-Customized Approaches for Analysis of Big Data

Unit-V

Integrating Data Sources-Real-Time Data Streams and Complex Event Processing-Operationalizing Big Data.

Suggested readings

1. Judith Hurwitz, Alan Nugent, Fern Halper, Marcia Kaufman, (2013). Big Data For Dummies, Wiley India,New Delhi.

2. Paul Zikopoulos, Dirk deRoos, Krishnan Parasuraman, Thomas Deutsch, James Giles, David Corrigan,(2012). Harness the Power of Big Data The IBM Big Data Platform, Tata McGraw Hill Publications, New Delhi.

3. Michael Minelli, Michele Chambers, Ambiga Dhiraj,(2013). Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses, Wiley Publications, New Delhi.

4. Zikopoulos, Paul, Chris Eaton, (2011). Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data, Tata McGraw Hill Publications, New Delhi.

Websites

1. www.oracle.com/BigData
2. www.planet-data.eu/sites/default/files/Big_Data_Tutorial_part4.pdf
3. www.ibm.com/developerworks/data
4. www.solacesystems.com
5. en.wikipedia.org/wiki/Big_data
6. www.sap.com/solution/big-data.html

Semester – VI

19CAU603B SYSTEM PROGRAMMING 3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

Enable the student

- To introduce students the concepts and principles of system programming
- To provide students the knowledge about both theoretical and practical aspects of system programming, teaching them the methods and techniques for designing and implementing system-level programs.
- Describe the utility of different system programs & system tools.
- Familiarize with the tradeoffs between run-time and compile-time processing (Linking & Loading techniques).
- Explore the use of compiler with its phases.
- Use of Syntax directed scheme for intermediate code generation.
- Construct & use of different compiler tools as LeX, Yacc for code generation & optimization.

Course Outcome

Upon completion of this course, students will be able to:

- Organize the functionalities & components of system software & tools into different layers for efficient code generation.
- Apply the knowledge & technique to develop solutions to real world problems by compiling application programs.
- ability to identify, formulate, and solve computer engineering problems with proper systematic & semantic approach.
- Develop possible program constructs for further code generation with Type checking & memory management strategy
- Design a simple compiler with tools & different with optimized techniques
- Design and implement system utility programs.

Unit I - ASSEMBLERS & LOADERS, LINKERS

One pass and two pass assembler design of an assembler, Absolute loader, relocation and linking concepts, relocating loader and Dynamic Linking., overview of compilation, Phases of a compiler.

Unit II - LEXICAL ANALYSIS

Role of a Lexical analyzer, Specification and recognition of tokens, Symbol table, lexical

Unit III – PARSING

Bottom up parsing- LR parser, **Intermediate representations:** Three address code generation, syntax directed translation, translation of types, control Statements.

Unit IV- STORAGE ORGANIZATION

Activation records stack allocation.

Unit V- CODE GENERATION

Object code generation

SUGGESTED READINGS

1. Santanu Chattopadhyaya. (2011). Systems Programming. New Delhi: PHI.
2. Alfred, V. Aho., Monica, S. Lam., Ravi Sethi., & Jeffrey, D. Ullman. (2006). Compilers: Principles, Techniques, and Tools (2nd ed.). New Delhi: Prentice Hall.
3. Dhamdhere, D. M. (2011). Systems Programming. New Delhi: Tata McGraw Hill.
4. Leland Beck., & Manjula, D. (2008). System Software: An Introduction to System Programming (3rd ed.). New Delhi: Pearson Education.
5. Grune, D., Van Reeuwijk, K., Bal, H. E., Jacobs, C. J. H., & Langendoen, K.(2012). Modern Compiler Design (2nd ed.). Springer.

WEBSITES

1. https://cs.gmu.edu/~setia/cs365-S02/assembler.pdf
2. https://www.geeksforgeeks.org/compiler-lexical-analysis/
3. https://www.javatpoint.com/parser

Semester – VI

19CAU611A PHP PROGRAMMING - PRACTICAL 4H – 2C

Instruction Hours / week: L: 0 T: 0 P: 4 Marks: Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

Enable the student

- To write basic PHP syntax using various operators.
- To write PHP scripts to handle HTML forms.
- To analyze different tasks using PHP functions.
- To understand the regular expressions in PHP.
- To learn array data structure using PHP scripts.
- To Create conditional structures

Course Outcomes (COs)

Upon completion of this course, students will be able to:

1. Write PHP scripts using operators to perform various functions
2. Design PHP scripts to handle HTML forms.
3. Implement different types of PHP functions.
4. Write regular expressions including modifiers, operators, and metacharacters.
5. Create PHP scripts using array.
6. Analyze and solve various database tasks using the PHP language.

List of Programs:

1. Create a PHP page using functions for comparing three integers and print the largest number.
2. Write a function to calculate the factorial of a number (non-negative integer). The function accept the number as an argument.
3. Write a simple PHP program to check whether the given number is prime or not.
4. Create a PHP page which accepts string from user. After submission that page displays the reverse of provided string.
5. Write a PHP function that checks if a string is all lower case.
6. Write a PHP script that checks whether a passed string is palindrome or not? (A palindrome is word, phrase, or sequence that reads the same backward as forward, e.g., madam or nurses run)
7. WAP to sort an array.
8. Write a PHP script that removes the whitespaces from a string.

Sample string : 'The quick " " brown fox' Expected Output : Thequick""brownfox

9. Write a PHP script that finds out the sum of first n odd numbers.

10. Create a login page having user name and password. On clicking submit, a welcome message should be displayed if the user is already registered (i.e.name is present in the database) otherwise error message should be displayed.
11. Write a PHP script that checks if a string contains another string.
12. Create a simple 'birthday countdown' script, the script will count the number of days between current day and birth day.

Suggested Readings

1. David Sklar, Adam Trachtenberg, (2014). *PHP Cookbook: Solutions & Examples for PHP*.
2. Luke Welling, Laura Thompson,(2008). *PHP and MySQL Web Development*, (4th ed.), Addition Paperback, Addison-Wesley Professsional.
3. Robin Nixon,(2014). *Learning PHP, MySQL, JavaScript, CSS & HTML5, (3rd ed.) Paperback*, O'reilly.
4. Steven Holzner, (2007). *PHP: The Complete Reference Paperback*, McGraw Hill Education (India), 2007.
5. Timothy Boronczyk, Martin E. Psinas, (2008). *PHP and MYSQL (Create-Modify-Reuse)*, Wiley India Private Limited.

Semester – VI

19CAU611B UNIX / LINUX PROGRAMMING - PRACTICAL 4H – 2C

Instruction Hours / week: L: 0 T: 0 P: 4 Marks: Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

Enable the student to

- Understand fundamental operating system abstractions such as processes, threads, files, semaphores, IPC abstractions, shared memory regions, etc.
- Understand how the operating system abstractions can be used in the development of application programs, or to build higher level abstractions
- Understand how the operating system abstractions can be implemented
- Understand the principles of concurrency and synchronization, and apply them to write correct concurrent programs/software
- Understand basic resource management techniques (scheduling or time management, space management) and principles and how they can be implemented. These also include issues of performance and fairness objectives, avoiding deadlocks, as well as security and protection.
- provide introduction to UNIX Operating System and its File System
- gain an understanding of important aspects related to the SHELL and the process
- develop the ability to formulate regular expressions and use them for pattern matching.

Course Outcome

Upon completion of this course, students will be able to:

1. Describe the architecture and features of UNIX Operating System and distinguish it from other Operating System
2. Demonstrate UNIX commands for file handling and process control
3. Write Regular expressions for pattern matching and apply them to various filters for a specific task
4. Analyze a given problem and apply requisite facets of SHELL programming in order to devise a SHELL script to solve the problem
5. Implement various file processing commands used in UNIX.
6. Construct various shell scripts for simple applications.

List of Programs:

1. Write a shell script to check if the number entered at the command line is prime or not.
2. Write a shell script to modify ―call command to display calendars of the specified months.
3. Write a shell script to modify ―call command to display calendars of the specified range of months.
4. Write a shell script to aCSUept a login name. If not a valid login name display message – ―Entered login name is invalid.

5. Write a shell script to display date in the mm/dd/yy format.
6. Write a shell script to display on the screen sorted output of ―who command along with the total number of users .
7. Write a shell script to display the multiplication table any number,
8. Write a shell script to compare two files and if found equal asks the user to delete the duplicate file.
9. Write a shell script to find the sum of digits of a given number.
10. Write a shell script to merge the contents of three files, sort the contents and then display them page by page.
11. Write a shell script to find the LCD (least common divisor) of two numbers.
12. Write a shell script to perform the tasks of basic calculator.
13. Write a shell script to find the power of a given number.
14. Write a shell script to find the binomial coefficient C(n , x).
15. Write a shell script to find the permutation P(n,x).
16. Write a shell script to find the greatest number among the three numbers.
17. Write a shell script to find the factorial of a given number.
18. Write a shell script to check whether the number is Armstrong or not.
19. Write a shell script to check whether the file have all the permissions or not

Suggested Readings

1. Michael Jang, (2011). *RHCSA/ RHCE Red Hat Linux Certification: Exams (Ex200 & Ex300)* ,Certification Press.
2. Nemeth Synder & Hein,(2010). *Linux Administration Handbook,* (2nd ed.) Pearson Education
3. Richard Stevens,W., Bill Fenner, Andrew M. Rudoff, (2014). *Unix Network Programming, The sockets Networking,* Vol. 1, 3rd ed.) API.
4. Sumitabha, Das, (2006). *Unix Concepts And Applications*, Tata McGraw-Hill Education.

Semester – VI

19CAU612A DATABASE ADMINISTRATION - PRACTICAL 4H - 2C

Instruction Hours / week: L: 0 T: 0 P: 4 Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3 Hours

Course Objectives

Enable the student to

- Installing Oracle Software
- Creating an Oracle Database Using DBCA
- Managing Database instances and ASM instances
- Managing and controlling database network environment
- Define and devise transaction management, concurrency control, crash recovery components
- Managing storage structures
- Controlling user security
- Designing Database backup and recovery procedures
- Take Decisions related with Database Maintenance

Course Outcomes (COs)

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

1. Design, model and install any database management systems by using Oracle database as sample.
2. Plan, design, construct, control and manage database instances, database network environment, storage structures, user security, database backup and recovery, database maintenance
3. Define and devise transaction management, concurrency control, crash recovery components
4. Examine and perform data base administration roles and operations by using Oracle database system as a sample
5. Compare and contrast by examining the database systems and new trends in data storage, data retrieval and maintenance techniques.
6. Configure, manage and maintain database, to audit and improve database performances and to use tools for database administration.

List of Programs:

1. Demo for Globalization Support
2. Setup Listener Security
3. Configuring Recovery Manager
4. Write a program Using Recovery Manager
5. Write a program for Managing Diagnostic Sources

6. Implement Database Recovery
7. Demo for Flashback Database
8. Implement User Error Recovery
9. Write a program for Dealing with Corruption
10. Show the demo for Automated Management
11. Creating a database and do the manipulation.
12. Managing index tables

Suggested Readings

3. Bob Bryla, Kevin Loney 2008 Oracle Database 11g DBA Handbook McGraw-Hill Osborne
4. Saikat Basak. 2010. Oracle DBA Concise Handbook ,Ensel Software

Websites :

4. www.oracle.com/technology/software/products/database/oracle10g/index.html
5. www.oracle-base.com/articles/10g/
6. www.adp-gmbh.ch/ora/misc/10g.html

Semester – VI

19CAU612B CLOUD COMPUTING - PRACTICAL 4H – 2C

Instruction Hours / week: L: 0 T: 0 P: 4 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

Enable the student to

- Basics of cloud computing.
- Key concepts of virtualization.
- Different Cloud Computing services
- Cloud Implementation, Programming and Mobile cloud computing
- Key components of Amazon Web Services
- Cloud Backup and solution

Course Outcomes (COs)

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

1. Define Cloud Computing and memorize the different Cloud service and deployment models
2. Describe importance of virtualization along with their technologies.
3. Use and Examine different cloud computing services
4. Analyze the components of open stack & Google Cloud platform and understand Mobile Cloud Computing
5. Describe the key components of Amazon web Service
6. Design & develop backup strategies for cloud data based on features

List of Programs:

1. Create virtual machines that aCSUess different programs on same platform.

2. Create virtual machines that aCSUess different programs on different platforms.

3. Working on tools used in cloud computing online-

   a) Storage

   b) Sharing of data

   c) Manage your calendar, to-do lists,

   d) A document editing tool

4. Exploring Google cloud

5. Exploring Microsoft cloud

6. Exploring Amazon cloud

Suggested Readings

1. Barrie Sosinsky,(2010). *Cloud Computing Bible*,Wiley-India.
2. Dimitris N. Chorafas,(2010). Cloud Computing Strategies ,CRC Press.
3. Gautam Shroff, (2010). *Enterprise Cloud Computing Technology Architecture Application*s, Cambridge University Press.
4. Gautam Shroff, (2010). Enterprise Cloud Computing Technology Architecture Applications, Cambridge University Press.
5. Nikos Antonopoulos, Lee Gillam,(2012). *Cloud Computing: Principles, Systems and Applications*, Springer Publications.
6. Ronald L. Krutz, Russell Dean Vines,(2010). Cloud Security: A Comprehensive Guide to Secure Cloud Computing, , Wiley-India.
7. Toby Velte, Anthony Velte, Robert Elsenpeter,(2010). Cloud Computing, A Practical Approach ,McGraw Hills.

Websites

1. en.wikipedia.org/wiki/Cloud_computing
2. www.ibm.com/cloud-computing/in/en/
3. www.oracle.com/CloudComputing
4. www.microsoft.com/en-us/cloud/default.aspx

Semester – VI

19CAU613A BIG DATA ANALYTIC - PRACTICAL 3H – 1C

Instruction Hours / week: L: 0 T: 0 P: 3 **Marks:** Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

Enable the student

- To know the fundamental concepts of big data and analytics.
- To explore tools and practices for working with big data
- To learn about stream computing.
- To know about the research that requires the integration of large amounts of data.
- To introduce the tools required to manage and analyze big data like Hadoop, NoSql MapReduce.
- To teach the fundamental techniques and principles in achieving big data analytics with scalability and streaming capability

Course Outcomes (COs)

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

1. Work with big data tools and its analysis techniques
2. Analyze data by utilizing clustering and classification algorithms
3. Learn and apply different mining algorithms and recommendation systems for large volumes of data
4. Perform analytics on data streams
5. Learn NoSQL databases and management.
6. Understand the key issues in big data management and its associated applications in intelligent business and scientific computing

List of Programs:

1. Implement a quicksort using scala.
2. Implement a auction service using scala.
3. Write a scala function to perform any 10 arithmetic operations.
4. Write a program to find the factorial of a given number using recursion.
5. Write a program for string manipulations.
6. Write a program for alphabetic order arrangement of a set of names.
7. Write a program for student records using scala list.
8. Implement any 5 map methods for maintaining customer details.
9. Implement employee records using Files
10. Write a program to copy the files using command line arguments.

Suggested readings

1. Judith Hurwitz, Alan Nugent, Fern Halper, Marcia Kaufman, (2013). Big Data For Dummies, Wiley India,New Delhi.

2. Paul Zikopoulos, Dirk deRoos, Krishnan Parasuraman, Thomas Deutsch, James Giles, David Corrigan,(2012). Harness the Power of Big Data The IBM Big Data Platform, Tata McGraw Hill Publications, New Delhi.

3. Michael Minelli, Michele Chambers, Ambiga Dhiraj,(2013). Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses, Wiley Publications, New Delhi.

4. Zikopoulos, Paul, Chris Eaton, (2011). Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data, Tata McGraw Hill Publications, New Delhi.

Websites

1. www.planet-data.eu/sites/default/files/Big_Data_Tutorial_part4.pdf
2. www.ibm.com/developerworks/data
3. www.solacesystems.com
4. en.wikipedia.org/wiki/Big_data
5. www.sap.com/solution/big-data.html

Semester – VI

19CAU613B SYSTEM PROGRAMMING - PRACTICAL 3H – 1C

Instruction Hours / week: L: 0 T: 0 P: 3 Marks: Internal: **40** External: **60** Total: **100**

End Semester Exam: 3 Hours

Course Objectives

- To introduce students the concepts and principles of system programming
- To provide students the knowledge about both theoretical and practical aspects of system programming, teaching them the methods and techniques for designing and implementing system-level programs.
- Describe the utility of different system programs & system tools.
- Familiarize with the tradeoffs between run-time and compile-time processing (Linking & Loading techniques).
- Explore the use of compiler with its phases.
- Use of Syntax directed scheme for intermediate code generation.
- Construct & use of different compiler tools as LeX, Yacc for code generation & optimization.

Course Outcome

Upon completion of this course, students will be able to:

1. Organize the functionalities & components of system software & tools into different layers for efficient code generation.
2. Apply the knowledge & technique to develop solutions to real world problems by compiling application programs.
3. ability to identify, formulate, and solve computer engineering problems with proper systematic & semantic approach.
4. Develop possible program constructs for further code generation with Type checking & memory management strategy
5. Design a simple compiler with tools & different with optimized techniques
6. Design and implement system utility programs.

List of Programs:

1. To implement an assembler for a hypothetical language.

2. Write a program to recognize numbers, identifiers.

3. Write a program to develop desk calculator.

SUGGESTED READINGS

1. Santanu Chattopadhyaya. (2011). Systems Programming. New Delhi: PHI.
2. Alfred, V. Aho., Monica, S. Lam., Ravi Sethi., & Jeffrey, D. Ullman. (2006). Compilers: Principles, Techniques, and Tools (2nd ed.). New Delhi: Prentice Hall.

3. Dhamdhere, D. M. (2011). Systems Programming. New Delhi: Tata McGraw Hill.
4. Leland Beck., & Manjula, D. (2008). System Software: An Introduction to System Programming (3rd ed.). New Delhi: Pearson Education.
5. Grune, D., Van Reeuwijk, K., Bal, H. E., Jacobs, C. J. H., & Langendoen, K.(2012). Modern Compiler Design (2nd ed.). Springer.

WEBSITES

1. https://cs.gmu.edu/~setia/cs365-S02/assembler.pdf