K.L.E. SOCIETY'S

BASAVAPRABHU KORE ARTS, SCIENCE AND COMMERCE COLLEGE, CHIKODI – 591 201.

ACCREDITED at "A" with 3.26 CGPA in 3rd Cycle

Department of Computer Science

Computer Science Course Structure Academic Year 2017-18

Course	Course Type	Course Title	Lectures/ Week Theory/Practical
B.Sc-I Semester	Core Course	Programming with C Programing Lab- C Lab	4/4
B.Sc-II Semester	Core Course	Data Structure using C Programing Lab-Data Structures Using C	4/4
B.Sc-III Semester	Core Course	OOPs Using C++ Programing Lab- OOPs Using C++	4/4
B.Sc-IV Semester	Core Course	Introduction to UNIX Programing Lab: **UNIX Lab**	4/4
B.Sc-V Semester	Core Course	Operating Systems (Paper – I) Programing Lab-Linux DBMS(Paper-II) Programing Lab-SQL and PL/SQL lab	8/8
B.Sc-VI Semester	Core Course	Computer Networks (Paper – I) Programing Lab-Networking lab Core Java (Paper – II) Programing Lab- Java programming	8/8
B.Com-II Semester	Core Course	Computer Applications in Business-I	4/2
B.Com –III Semester	Core Course	Computer Applications in Business-II	4/2
B.Com -IV Semester	Core Course	Computer Applications in Business-III	4/2

B.Com -V Semester	Core Course	Computer Applications in Business-IV	4/2
B.Com -VI Semester	Core Course	Computer Applications in Business-V	4/2

FIRST-TERM

Course:B.Sc-I	**Programming with C** **(Lectures/Week:4)** **Facilitator:Miss D.B Patil**

Objectives:

The objective of this course is to provide a comprehensive study of the C programming language, stressing upon the strengths of C, which provide the students with the means of writing modular,efficient,maintainance and portable code.

Learning Outcomes:

- Students should be able to write, compile and debug programs in C language.
- Students should be able to use different data types in a computer program.
- Students should be able to design programs involving decision structures, loops and functions.
- Students should be able to explain the difference between call by value and call by reference.
- Students should be able to explain the difference types string functions.
- Students should be able to use different data structures.

UNIT-I	**Evolution of information processing**: Concept of data and information, data processing. Hardware –CPU, Storage Devices & Media, VDU, Input – Output devices, Types of Software – System Software, Application Software. Overview of OS. Programming Languages and its Classification, Compiler, Interpreter, Linker, Loader. **Problem Solving**: Problem Identification, Analysis, flowcharts, Decision Tables, Pseudo codes and algorithms, Program Coding, Program Testing and Execution..	10Hrs
UNIT-II	**Overview of C**: Elements of C: C character set, identifiers and keywords, Data types, Constants and Variables, Assignment statement, Symbolic constant, Structure of a C Program, printf(), scanf() Functions, Operators & Expression: Arithmetic, relational, logical, bitwise, unary, assignment, shorthand assignment operators, conditional operators and increment and decrement operators, Arithmetic expressions, evaluation of arithmetic expression, type casting and conversion, operator hierarchy & associativity.	10Hrs

UNIT-III	**Decision making & branching**: Decision making with IF statement, IF-ELSE statement, Nested IF statement, ELSE-IF ladder, switch statement, goto statement. Decision making & looping: For, while, and do-while loop, jumps in loops - break, continue statement, Nested loops. **Functions:** Standard Mathematical functions, Input/output: Unformatted & formatted I/O function in C. User defined functions: definition, prototype, Local and global variables, passing parameters, recursion.	10Hrs
UNIT-IV	Arrays, strings and pointers: Definition, types, initialization, processing an array, passing arrays to functions, Array of Strings. String constant and variables, Declaration and initialization of string, Input/output of string data, Storage classes in C: auto, extern, register and static storage class, their scope, storage, & lifetime. **String Handling:** String Library Functions: strlen, strcat, strcmp, strcpy, strrev.	10Hrs
UNIT-V	**Structure & Union**: Definition of Structure, Declaring Structure, Accessing Structure Elements, Array of Structure, Nesting of Structure. Definition of Union, declaring and using Union. Difference between Structure & Union. **Error Handling during I/O Operations**, Command Line Arguments, Documentation, debugging, C Processors, Macros.	

Learning Materials

Text Books:

Programming in ANSI C(Third Edition):E Balaguruswamy

Yashavant P.Kanetkar. “Let Us C”, BPB Publications.

Soft and Hard copy of Notes, References Websites

Assessment

Assessment is carried out as per the guidelines laid down and mandated by the affiliating University.

100 marks exam (20 IA + 80 Semester End Exam)

1. Two Internal Tests (IA): 20marks

Internal Test 1:	20 marks reduced to	04
Internal Test 2:	80 marks reduced to	10
Attendance:		03
Class seminars, Tutorials, Sports & Cultural Activities, Assignments, NSS/NCC:		03

2. Semester End Examination as per University guidelines: 80 marks

Course:B.Sc-III	**OOPs Using C++ (Lectures/Week:4)** **Facilitators:Miss G.B. Kustigar,Miss D.B PatiL**

Objectives:

1. To learn advanced features of the C++ programming language as a continuation of the previous course.

2. 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.

3. To enhance problem solving and programming skills in C++ with extensive programming projects. 4. To become familiar with the UNIX software development environment.

Learning Outcomes:

- Explain how an existing C++ program works
- Discover errors in a C++ program and describe how to fix them
- Analyze a problem and construct a C++ program that solves it
- Choose and apply the required Linux commands to develop C++ programs in a command-line environment

UNIT-I	**1. Introduction:** Procedural Languages, definition of OOP, Basic concept of OOP, Object, Class, Data Abstraction, Data Encapsulation, Data Hiding member functions, Reusability, Inheritance, Creating new Data Types, Polymorphism, Overloading, Dynamic binding, and Message passing. **C++ Features:** The i/o-stream class, C++ Comments, C++ Keywords, Variable declaration, The Const Qualifier. The Endl, Set W, set precision, Manipulators, The scope resolution operator, the new & delete Operators. **Functions:** Simple Functions: Function declaration, calling the function, function definition; Passing argument to, returning value from function; passing constants, Variables, pass by value, passing structure variables, pass by reference, Default arguments, return statements, return by reference, overloaded functions; Different number of arguments, Different Kinds of arguments, inline function.	12Hrs
UNIT-II	**Objects & Classes:** Classes & Objects, Class Declaration, Class members; Data Constructors, Destructors, Member functions, Class member visibility; private, public, protected. The scope of the class object constructors; Default Constructor, Constructor with argument, constructor with default arguments, Dynamic constructor, copy constructor, Overloaded constructor, Objects as function arguments; member functions defined outside the class, Objects as arguments, returning objects from functions, class conversion, manipulating private Data members, Destructors, classes, objects & memory, array as class member data, Array of objects, string as class member	10Hrs
UNIT-III	**Operator Overloading:** Overloading unary operator: Operator Keyword, Operator Arguments, Operator return value, Nameless temporary objects, limitations of increment operator, overloading binary operator, arithmetic operators, comparison. Operator, arithmetic assignment operator, Data conversion.;, conversion .between Basic	10Hrs

	types, Conversion between objects & Basic types, conversion between objects of different classes. **Inheritance:** Derived Class & Base Class: Specifying the Derived class accessing Base class members, the protected access specifier, derived class constructor.	
UNIT-IV	**Inheritance:** Overriding member functions, public and private inheritance; Access Combinations, Classes & Structures, Access Specifies, Level of inheritance; multilevel inheritance, Hybrid inheritance, Multiple inheritance; member functions in multiple inheritance, constructors in multiple inheritance, Containership; Classes within classes, Inheritance &Program Development. **Virtual Functions:** Normal member function accessed with pointers, Virtual member functions accessed with pointers, Dynamic binding, pure virtual functions. **(10 Hrs)**	10Hrs
UNIT-V	**Virtual Functions**: Friend function; Friends for functional notation, friend classes, this pointer Accessing Member Data with this, using this for returning values. **Templates & Exception Handling:** Introduction, Templates, Class Templates, function templates, Member function templates, Template arguments, Exception Handling. **Streams:** The Stream class Hierarchy, Stream classes -Header file, string I/O: Writing strings, reading strings, character J/O, Detecting End - of - file. Object J/O; writing an object to disk, reading an object from disk, J/O with multiple objects; the f stream class, The open function, File Pointers; Specifying the position, Specifying the offset. The tellg Function, Disk I/O with Memory Functions; Closing Files, Error Handling, Command Line Arguments	10Hrs

Learning Materials

Text Books:

1. E. Balaguruswamy: Object Oriented Programming with C++, Tata McGraw Hill Publications.
2. Lafore Robert: Object Oriented Programming in Turbo C++, Galgotia Publications

Soft and Hard copy of Notes, References Websites

Assessment

Assessment is carried out as per the guidelines laid down and mandated by the affiliating University.

100 marks exam (20 IA + 80 Semester End Exam)

1. Two Internal Tests (IA): 20marks

Internal Test 1: 20 marks reduced to 04

Internal Test 2: 80 marks reduced to 10

Attendance: 03

Class seminars, Tutorials, Sports & Cultural Activities, Assignments, NSS/NCC: 03

2. Semester End Examination as per University guidelines: 80 marks

Course:B.Sc-V	**Operating Systems (Paper – I)** **(Lectures/Week:4)** **Facilitator:Miss S.M Hegale**

Objectives: Students will demonstrate a knowledge of process control, threads, concurrency, memory management scheduling, I/O and files, distributed systems, security, networking. Student teams will implement a significant portion of an operating system.

Learning Outcomes:

1. Appreciate the role of operating system as System software.
2. Compare the various algorithms and comment about performance of various algorithms used for management of memory, CPU scheduling, File handling and I/O operations.
3. Apply various concept related with Deadlock to solve problems related with Resources allocation, after checking system in Safe state or not.
4. To appreciate role of Process synchronization towards increasing throughput of system.
5. Describe the various Data Structures and algorithms used by Different Oss like Windows XP , Linux and Unix pertaining with Process , File , I/O management.
6. To control the behavior of OS by writing Shell scripts.

UNIT-I	**Introduction:** Batch Systems, Concepts of Multiprogramming and Time Sharing,Parallel, Distributed and real time Systems, Operating System Structures, Components and Services, System programs, Virtual machines. **Process Management :** Process concept, Process scheduling, Co-operating process, Threads, Inter process communication, CPU scheduling criteria, Scheduling algorithm.	12Hrs
UNIT-II	**Process synchronization and deadlocks:** The critical section problem, Synchronization hardware, Semaphores, Classical problems of synchronization,Critical regions, monitors, Dead locks –System model , characterization, Dead lock prevention, avoidance and detection, Recovery from dead lock.	10Hrs
UNIT-III	**Memory Management:** Logical and Physical address space, Swapping Contiguous allocation, Paging, Segmentation, Virtual memory – Demand paging and it’s performance, page replacement algorithms, Allocation of frames, thrashing**.**	10Hrs
UNIT-IV	**File management (System, Secondary storage structure):** File concepts, Access methods, Directory structure, Protection and consistency, semantics, File system structure, Allocation methods, Free space management.	10Hrs
UNIT-V	**Disk Management (Structure, Disk Scheduling Methods):** Disk structure and Scheduling methods, Disk management, Swap – Space management. **Protection and Security:** Goals of protection, Domain protection, Access matrix security problem, Authentication, One time password.	10Hrs

Learning Materials

Text books:

1. Abraham siberschatz and peter Bear Galvin, Operating System Concepts, Fifth Edition, Addision – Wesley
2. Nutt: Operating system, 3/e person education 2004.

Soft and Hard copy of Notes, References Websites

Assessment

Assessment is carried out as per the guidelines laid down and mandated by the affiliating University.

100 marks exam (20 IA + 80 Semester End Exam)

1. Two Internal Tests (IA): 20marks

Internal Test 1:	20 marks reduced to	04
Internal Test 2:	80 marks reduced to	10
Attendance:		03
Class seminars, Tutorials, Sports & Cultural Activities, Assignments, NSS/NCC:		03

2. Semester End Examination as per University guidelines: 80 marks

Course:B.Sc-V	**Database Management Systems (Paper – II)** **(Lectures/Week:4)** **Facilitator:Miss G.B. Kustigar**

Objectives:

The objective of this course is to introduce the concept of the DBMS with respect to the relational model, to specify the functional and data requirements for a typical database application and to understand creation, manipulation and querying of data in databases

Learning Outcomes:

1. Students should be able to evaluate business problem and find the requirements of a problem in terms of data.
2. Students should be able to design the databases schema with the use of appropriate data types for storage of data in database.
3. Students should be able to create, manipulate, query and back up the databases.

UNIT-I	**Introduction:** Database and Database Users, Characteristics of the Database Approach, Actors on the scene, Workers behind the Scene, Advantages of using DBMS, Brief History. **Database System Concepts and Architecture:** Data Models,Schemas, and Instances, Three Schema Architecture and Data Independence,Database language and interfaces, the database system Environment, Centralized and Client/Server Architectures for DBMS, Classification of Database Management Systems.	10Hrs

UNIT-II	**Data modeling using the Entity–Relationship (ER) model**: High level conceptual data models for database design with an example, 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,Relationship types of degree higher than two, EER Model.	10Hrs
UNIT-III	**Relational Data Model and Relational Algebra**: Relation Data Model and Relational Database Constraints, Relation Algebra, Relational Database Design by ER and EER to Relational Mapping.	10Hrs
UNIT-IV	**Functional dependencies and Normalization for Relational Databases**: Informal Design Guidelines for Relational Schemas, Functional Dependencies, Normal Forms based on Primary Keys, General Definition of 2NF and 3NF, Boyce-Codd Normal Form(BCNF).	10Hrs
UNIT-V	**Relational Database Language**: Data definition in SQL, Queries in SQL, Insert,Delete and Update Statements in SQL, Views in SQL, Specifying General Constraints as Assertions, Specifying indexes, Embedded SQL. **Transaction Processing Concepts**: Introduction, Transaction and System Concepts, Desirable properties of transaction, Schedules and Recoverability, Serializability of Schedules, Transaction Support in SQL, Locking Techniques for Concurrency Control.	10Hrs

Learning Materials
Text Book:

A. Ramez Elmasri & Shamkant B. Navathe, Fundamentals of Database Systems(Sixth Edition), Pearson Education, 2011).

B. Sundarraman , Oracle 9i programming A Primer, 1/e Pearson Education.Soft and Hard copy of Notes, References Websites

Assessment

Assessment is carried out as per the guidelines laid down and mandated by the affiliating University.

100 marks exam (20 IA + 80 Semester End Exam)

1. Two Internal Tests (IA): 20marks

Internal Test 1:	20 marks reduced to	04
Internal Test 2:	80 marks reduced to	10
Attendance:		03
Class seminars, Tutorials, Sports & Cultural Activities, Assignments, NSS/NCC:		03

2. Semester End Examination as per University guidelines: 80 marks

Practical:

B.Sc-I Semester	**Programming Lab- C programming** **Practical Hours: 4 Hrs/week** **Facilitators:Miss D.B Patil, Miss S. M Hegale**
	1.Write a program to enter length and breadth of a rectangle and find its perimeter and area. 2. Write a program to enter P, T, R and calculate Simple Interest. 3. Write a program to find maximum between three numbers. 4. Write a program to check whether year is leap year or not using

	conditional/ternary operator. 5. Write a program to function as a basic calculator; it should ask the user to input what type of arithmetic operation he would like, and then ask for the numbers on which the operation should be performed. The calculator should then give the output of the operation. 6. Write a program that takes in three arguments, a start temperature (in Celsius), an end temperature (in Celsius) and a step size. Print out a table that goes from the start temperature to the end temperature, in steps of the step size; Celsius to Farenheit. 7. Write a program to sort array elements in ascending order. 8. Write a program to subtract/add/multiply two matrices. 9. Write a program to check whether an alphabet is vowel or consonant using switch case. 10. Write a program to display all possible permutations of a given input string--if the string contains duplicate characters, you may have multiple repeated results. Input should be of the form permute *string* and output should be a word per line. Here is a sample for the input *cat* cat cta act act act atc tac tca 11. Write a function that accepts a number, n, and prints all prime numbers between 1 to n. 12. Write an iterative function calculate factorial of a given integer. 13. Write a program to find HCF (GCD) of two numbers by passing two numbers to function comp GCD(). 14. Write a program to find maximum and minimum element in an array by passing array to function. 15. Write a program to input electricity unit charges and calculate total electricity bill according to the given condition: For first 50 units Rs. 0.50/unit For next 100 units Rs. 0.75/unit For next 100 units Rs. 1.20/unit For unit above 250 Rs. 1.50/unit An additional surcharge of 20% is added to the bill 16. Write a program to input marks of five subjects Physics, Chemistry, Biology, Mathematics and Computer. Calculate percentage and grade according to following. Use structure to create array of students and compute percentage and grade by passing structure to function. Percentage >= 90% : Grade A Percentage >= 80% : Grade B Percentage >= 70% : Grade C Percentage >= 60% : Grade D Percentage >= 40% : Grade E Percentage < 40% : Grade F 17. Write a C program to add two complex numbers by passing structure to a function. Consider the following structure definition for complex number typedefstruct complex { float real; floatimag; } complex; 18. Write a C program to illustrate difference between structure and union by defining emp_Name, slaray, job as members and displaying the size of the defined structure and union. (ie. In terms of memory allocation)

	19. Write a program that accepts a base ten (non-fractional) number at the command line and outputs the binary representation of that number. 20. Write a C program to concatenate two strings without using library function 21. Write a C program to compare two strings without using library function 22. Write a C program to illustrate string library functions (copy, concat, uppercase to lower case and vice-versa, length of string, sort set of strings(use strcmp()).
B.Sc-III Semester	**Programming Lab- OOPS Using C++** **Practical Hours: 4 Hrs/week** **Facilitators:Miss G.B Kustigar Miss S. M Hegale**
	Note : All the programs have to be implemented in LINUX environment only 1. Write a program to prepare a shopping list. 2. Write a program to swap two numbers using friend function. 3. Write a program to find sum of complex number using friend function. 4. Write a program to find maximum of two numbers using friend function. 5. Write a program to calculate area and circumference of circle using inline function. 6. Write a program to add two time variable. 7. Write a program to add two distance variable. 8. Write a program to implement area of geometrical figures. 9. Write a program to find the maximum of two numbers using template. 10. Write a program to sort elements using template. 11. Write a program to perform addition of two matrices using operator overloading. 12. Write a program to perform multiplication of two matrices using operator overloading. 13. Write a program to compare two strings using equal to operator. 14. Write a program to concatenate two strings. 15. Write a program to implement operation on stack. 16. Write a program to implement operation on queue. 17. Write a program to create a student report using inheritance technique. 18. Write a program to implement digital clock. 19. Write a program to create electricity bill. 20. Write a program to perform bank transaction **Practical Examination:** Evaluation criteria for practical examinations shall be as follows: **1. Writing of Programs - 15 Marks** a. One program from the journal list – 08 Marks b. Another program given by examiner based on the concepts studied -07Marks **2. Execution of programs – 15 Marks** a. Journal Program - 08 Marks b. Program of Examiner's Choice -07 Marks **3. Viva-Voce - 05 Marks** **4. Journal / Laboratory Report – 5 Marks** **Total Marks -40 Marks**
B.Sc –V Semester	**Operating Systems Lab** **Practical Hours: 4 Hrs/week** **Facilitators: Miss S. M Hegale, Miss D.B Patil**
	Implement the following on LINUX or other Unix like platform. Use C for

	high level language implementation 1. Write programs using the following system calls of UNIX operating system: fork, exec, getpid, exit, wait, close, stat, opendir, readdir 2. Write programs using the I/O system calls of UNIX operating system (open, read, write, etc) 3. Write C programs to simulate UNIX commands like ls, grep, etc. 4. Given the list of processes, their CPU burst times and arrival times, display/print the Gantt chart for FCFS and SJF. For each of the scheduling policies, compute and print the average waiting time and average turnaround time. (2 sessions) 5. Given the list of processes, their CPU burst times and arrival times, display/print the Gantt chart for Priority and Round robin. For each of the scheduling policies, compute and print the average waiting time and average turnaround time. (2 sessions) 6. Developing Application using Inter Process communication (using shared memory, pipes or message queues) 7. Implement the Producer – Consumer problem using semaphores (using UNIX system calls). 8. Implement some memory management schemes – I 9. Implement some memory management schemes – II 10. Implement any file allocation technique (Linked, Indexed or Contiguous) **Practical Examination (Scheme of Valuation)** Evaluation criteria for practical examinations shall be as follows: **1. Writing of Programs -15 Marks** a. One program from the journal list – 08 Marks b. Another program given by examiner based on the concepts studied -07Marks **2. Execution of programs – 15 Marks** a. Journal Program - 08 Marks b. Program of Examiner's Choice -07 Marks **3. Viva-Voce -05 Marks** **4. Journal / Laboratory Report – 5 Marks** **Total Marks -40 Marks**
B.Sc –V Semester	**Database Management Systems Lab** **Practical Hours: 4 Hrs/week** **Facilitators: Miss S. M Hegale, Miss G.B Kustigar** **I. Consider the Insurance database given below. The primary keys are underlined and the data types are specified.** PERSON (driver – id #: String, name: string, address: strong) CAR (Regno : string, model: string, year: int) ACCIDENT (report-number: int, accd-date: date, location: string) OWNS (driver-id #:string, Regno:string) PARTICIPATED (driver-id: string, Regno:string, report-number:int, damageamount:int) a) Create the above tables by properly specifying the primary keys and the foreign keys. b) Enter at least five tuples for each relation. c) Demonstrate how you a. Update the damage amount for the car with a specific Regno in the accident with report number 12 to 25000. b. Add a new accident to the database. d) Find the total number of people who owned cars that were involved in accidents

in 2008.

e) Find the number of accidents in which cars belonging to a specific model were involved.

f) Generate suitable reports.

II. Consider the following relations for an order processing database application in a company.

CUSTOMER (cust #: int , cname: string, city: string)
ORDER (order #: int, odate: date, cust #: int, ord-Amt: int)
ORDER – ITEM (order #: int, item #: int, qty: int)
ITEM (item # : int, unit price: int)
SHIPMENT (order #: int, warehouse#: int, ship-date: date)
WAREHOUSE (warehouse #: int, city: string)

a) Create the above tables by properly specifying the primary keys and the foreign keys.

b) Enter at least five tuples for each relation.

c) Produce a listing: CUSTNAME, of orders, AVG_ORDER_AMT, where the middle column is the total numbers of orders by the customer and the last column is the average order amount for that customer.

d) List the order# for orders that were shipped from *all* the warehouses that the company has in a specific city.

e) Demonstrate the deletion of an item from the ITEM table and demonstrate a method of handling the rows in the ORDER_ITEM table that contain this particular item.

f) Generate suitable reports.

III. Consider the following database of student enrolment in courses & books adopted for each course.

STUDENT (regno: string, name: string, major: string, bdate:date)
COURSE (course #:int, cname:string, dept:string)
ENROLL (regno:string, course#:int, sem:int, marks:int)
BOOK _ ADOPTION (course# :int, sem:int, book-ISBN:int)
TEXT (book-ISBN:int, book-title:string, publisher:string, author:string)

a) Create the above tables by properly specifying the primary keys and the foreign keys.

b) Enter at least five tuples for each relation.

c) Demonstrate how you add a new text book to the database and make this book be adopted by some department.

d) Produce a list of text books (include Course #, Book-ISBN, Book-title) in the alphabetical order for courses offered by the 'CS' department that use more than two books.

e) List any department that has *all* its adopted books published by a specific publisher.

f) Generate suitable reports.

IV. The following tables are maintained by a book dealer.

AUTHOR (author-id: int, name: string, city: string, country: string)
PUBLISHER (publisher-id: int, name: string, city: string, country: string)
CATALOG (book-id: int, title: string, author-id: int, publisher-id: int, category-id: int, year: int, price: int)
CATEGORY (category-id: int, description: string)
ORDER-DETAILS (order-no: int, book-id: int, quantity: int)

a) Create the above tables by properly specifying the primary keys and the foreign keys.

	b) Enter at least five tuples for each relation. c) Give the details of the authors who have 2 or more books in the catalog and the price of the books is greater than the average price of the books in the catalog and the year of publication is after 2000. d) Find the author of the book which has maximum sales. e) Demonstrate how you increase the price of books published by a specific publisher by 10%. f) Generate suitable reports. **V. Consider the following database for a banking enterprise** BRANCH (branch-name: string, branch-city: string, assets: real) ACCOUNT (accno: int, branch-name: string, balance: real) DEPOSITOR (customer-name: string, accno: int) CUSTOMER (customer-name: string, customer-street: string, customer-city: string) LOAN (loan-number: int, branch-name: string, amount: real) BORROWER (customer-name: string, loan-number: int) a) Create the above tables by properly specifying the primary keys and the foreign keys b) Enter at least five tuples for each relation c) Find all the customers who have at least two accounts at the *Main* branch. d) Find all the customers who have an account at *all* the branches located in a specific city.

SECOND-TERM

Course:B.Sc-II	**Data Structure using C** **(Lectures/Week:4)** **Facilitator:D. B Patil**
Objectives: To understand the concepts of Data Structures and its significance in solving problems using programming concepts. Provide holistic approach to design, use and implement abstract data types. Understand the commonly used data structures and various forms of its implementation for different applications using C **Learning Outcomes:** • Design and implement commonly used Data structures. • Design Abstract Data types and its implementation. • Ability to program various applications using appropriate data structures.	

UNIT-I	**Advanced C**: Dynamic memory allocation and pointers in C- Declaring and initializing pointers, Pointer & Functions, Pointer & Arrays, Pointer & Strings, Pointer& Structure, Pointer to Pointer. Static and dynamic memory allocation. Memory allocation functions :malloc, calloc, free and realloc. File Management in C: Defining and Opening & Closing File, Input & Output Operations on Files, Random Access to Files.	10Hrs
UNIT-II	**Introduction to Data structures**: Definition, Classification of data structures: primitive and nonprimitive. Operations on data structures **Search**: Basic Search Techniques- sequential search, Binary search- Iterative and Recursive methods. **Sort-** General Background: Definition, different types: Bubble sort, Selection sort, Merge sort, Insertion sort, Quick sort	10Hrs
UNIT-III	**Recursion:** Definition, Recursion in C, Writing Recursive programs – Binomial coefficient, Fibonacci, GCD, towers of Hanoi. **Stack** – Definition, Array representation of stack, Operations on stack- push and pop, Infix, prefix and postfix notations, Conversion of an arithmetic expression from Infix to postfix, applications of stacks.	10Hrs
UNIT-IV	Queue - Definition, Array representation of queue, Types of queue: Simple queue, circular queue, double ended queue (deque) priority queue, operations on all types of Queues.	10Hrs
UNIT-V	**Linked list** – Definition, components of linked list, representation of linked list, advantages and disadvantages of linked list, Arrays versus linked list, Types of linked list: Singly linked list, doubly linked list, Circular linked list and circular doubly linked list. Operations on singly linked list: creation, insertion, deletion, search and display. Implementation of stack and queues using linked list.	

Learning Materials

Text Books:

1. A. K. Sharma, Data Structures Using C, 2nd edition, Pearson Education.
2. Achuthsankar S. Nair, T. Makhalekshmi, Data Structures in C, PHI.
3. Mark Allen Weiss, Data Structures and Algorithm Analysis in C, Pearson Education.

Soft and Hard copy of Notes, References Websites

Assessment

Assessment is carried out as per the guidelines laid down and mandated by the affiliating University.

100 marks exam (20 IA + 80 Semester End Exam)

1. Two Internal Tests (IA): 20marks

Internal Test 1:	20 marks reduced to	04
Internal Test 2:	80 marks reduced to	10
Attendance:		03
Class seminars, Tutorials, Sports & Cultural Activities, Assignments, NSS/NCC:		03

2. Semester End Examination as per University guidelines: 80 marks

Course:B.Sc-IV	**UNIX Shell programming** **(Lectures/Week:4)** **Facilitator:Miss G.B Kustigar**

Objectives: This course introduces basic understanding of UNIX OS, UNIX commands and File system and to familiarize students with the Linux environment. To make student learn fundamentals of shell scripting and shell programming. Emphases are on making student familiar with UNIX environment and issues related to it.

Learning Outcomes:

1Students will be able to run various UNIX commands on a standard UNIX/LINUX Operating system.

2.Students will be able to run C / C++ programs on UNIX.

3. Students will be able to do shell programming on UNIX OS.

4. Students will be able to understand and handle UNIX system calls.

UNIT-I	**UNIX Architecture:** History of UNIX, UNIX Architecture, Features of UNIX,Internal and External Commands. **General Purpose Utilities** – man, cal, date, echo, printf, bc, script, mailx, passwd,uname, tty,sty. **File system** - The file, The parent-child relationship, the home directory, Directory Commands—pwd, cd, mkdir, rmdir, Absolute pathnames, Relative Pathnames, ls.	10Hrs
UNIT-II	**File Handling Commands** File handling Commands—cat, cp, rm, mv, file, wc, od, comm ,diff, os2unix, unix2dos, gzip, gunzip, tar, zip , unzip **Basic file attributes** ls -l , file ownership ,file permissions - chmod changing file permissions, Directory permissions, changing file ownership **vi Editor** -- Different modes of vi editor: input mode, Ex mode and command mode, search and replace	10Hrs
UNIT-III	**The Shell**--The shell interpretive cycle, pattern matching-- the wild card, Escaping,the backlash (/),quoting, redirection, /dev/null, /dev/tty: Two special files, Pipes, tee,command substitution, shell variables. **The process** Process basics, ps , Mechanism of process creation, running jobs in background, nice,Killing processes, Job Control, at and batch, cron, time commands **Customizing the Environment** customizing the environment using variables, common environment variables, command history.	10Hrs
UNIT-IV	**More File attributes** File systems and Inodes, Hard links, symbolic link and ln,The directory, umask, Modification & access times, find: Locating files **Simple Filters** - pr, head, tail, cut, paste, sort, uniq, tr, **Filters Using Regular Expression -** grep, basic regular expressions,	10Hrs

	extended regular expression, egrep, fgrep **Communication in unix** –finger, mesg , write , talk , wall , news ,mail.	
UNIT-V	**Essential Shell Programming** Shell scripts, **read, exit** ,Using Command Line Arguments, Logical Operators && and ||, **if** conditional, using **test** and [ ] to evaluate the expression, the **case** conditional ,**expr**, while: looping, for: looping with a list,arrays: single dimensional , the here document (<<), set and shift, trap, debugging the shell scripts with set –X.	10Hrs

Learning Materials

Text Books:

1. “Unix Concepts and Applications” by Sumithabha Das, 4nd Edition, Tata McGraw Hill 2006.
2. UNIX and Shell Programming, Behrouz A Forouzan and Richard F Gilberg, 1st edition, Thomson course Tecnology, 2005.

Soft and Hard copy of Notes, References Websites

Assessment

Assessment is carried out as per the guidelines laid down and mandated by the affiliating University.

100 marks exam (20 IA + 80 Semester End Exam)

1. Two Internal Tests (IA): 20marks

Internal Test 1:	20 marks reduced to	04
Internal Test 2:	80 marks reduced to	10
Attendance:		03
Class seminars, Tutorials, Sports & Cultural Activities, Assignments, NSS/NCC:		03

2. Semester End Examination as per University guidelines: 80 marks

Course:BA/B.Sc-IV	**Computer Applications (Compulsory)** **Teaching hour per week: 04** **Facilitators:Miss S.M Hegale,Shri V.M Bagi,Miss D,B Patil**

Objectives:

The course is designed to aim at imparting a basic level appreciation programme for the common man. After completing the course the incumbent is able to the use the computer for basic purposes of preparing his personnel/business letters, viewing information on Internet (the web), sending mails, using internet banking services etc.

UNIT-I	**1. Introduction to Computers:** Introduction, types of computer, components of computer, CPU, motherboard, primary storage devices: ROM, RAM secondary storage: floppy, hard disk and their types; CDROM, pen drive, Input &. output devices: keyboard, mouse, scanner, display units, printers (dot matrix, Inkjet & laser), multimedia components, liquid crystal display (LCD) projector, modems and network interfacing card.	8hrs
UNIT-II	**Windows Operating system:** Introduction, loading and starting windows, concept of plug and play, active desktop environment, control panel, adding new programs and hardware, menus, folders, shortcuts, display properties, system tools, multimedia programs, editing pictures using paint**.**	8hrs
UNIT-III	**MS-Word:** Introduction to' MS-office, installing and removing word, running programs and-managing files, opening, creating and saving documents, templates, navigating and selecting, .editing and sorting, ‘checking spelling and grammar, formatting, importing graphics and pictures, tables, long documents, sharing, data with other users, security, creating and working with web pages, mail merge, editing equations, printing.	8hrs
UNIT-IV	**MS EXCEL:** Introduction, creating , opening and saving files, working with workbooks end worksheets, spreadsheets, entering *and* selecting data, editing and formatting worksheets, mathematical functions, statistical functions, trigonometric functions, date and time functions, text functions, financial functions, lookup End reference functions, creation of charts and graphs, automated tasks, macros, switching from other applications, printing.	8hrs
UNIT-V	**MS-PowerPoint:** Introduction, auto-content wizard, design templates, adding and formatting text, making notes and handouts, adding clip arts, drawings and other objects, equations, tables and charts, controlling the slide show, animations, printing presentations and slides.	8hrs
	MS-Access: Introduction, databases, data structures, creating tables, importing and linking tables, working with data, working with queries, formatting forms and reports, writing expressions, working with macros, modules and events, replication, data access objects, data access methods and-properties.	6hrs
	Internet: Introduction, LAN and WAN, dial-up and broadband networking, internet protocols, TCP/IP protocol, Microsoft internet explorer, Netscape navigator, properties and customization, world wide web, HTML, creation of web page using templates, search engines, chatting, e-mail.	6hrs

Learning Materials:
Text Books: Sagman, MS Office 2000 for windows, Pearson Education, Microsoft-MS-Office 2007 step by step

Soft and Hard copy of Notes, References Websites

Assessment

Assessment is carried out as per the guidelines laid down and mandated by the affiliating University.

100 marks exam (20 IA + 80 Semester End Exam)

1. Two Internal Tests (IA): 20marks

Internal Test 1:	20 marks reduced to	04
Internal Test 2:	80 marks reduced to	10
Attendance:		03
Class seminars, Tutorials, Sports & Cultural Activities, Assignments, NSS/NCC:		03

2. Semester End Examination as per University guidelines: 80 marks

Course:B.Sc-VI	**Computer Networks (Paper – I)** **(Lectures/Week:4)** **Facilitator:Miss S.M Hegale**

Objectives:

- To educate concepts, vocabulary and techniques currently used in the area of computer networks.
- To study protocols, network standards, the OSI model, IP addressing, cabling, networking components, and basic LAN design.
- To accumulate existing state-of-the-art in network protocols, architectures, and applications.
- To be familiar with contemporary issues in networking technologies

Learning Outcomes:

- To understand the organization of computer networks, factors influencing computer network development and the reasons for having variety of different types of networks.
- To design a network routing for IP networks.
- To identify main internal PC components and connections.
- To explain how a collision occurs and how to solve it.
- To demonstrate proper placement of different layers of ISO model and illuminate its function.
- To learn Internet structure and can see how standard problems are solved in that context.
- To determine proper usage of the IP address, subnet mask and default gateway in a routed network.
- To understand internals of main protocols such as HTTP, FTP, SMTP, TCP, UDP, IP
- To analyze simple protocols and can independently study literature concerning computer networks

UNIT-I	**Introduction:** Computer Networks and its applications, Network structure, network architecture, Topologies, LAN, WAN, MAN, The OSI reference model, The TCP/IP reference model, services - SMDS, Frame relay, network standards, example networks, **The Physical Layer:** Transmission Media – Twisted pair, coaxial cable, optical fiber, radio transmission, microwaves and infrared transmission, Switching –message switching Circuit switching, packet switching	10Hrs
UNIT-II	**The Data Link Layer:** Data Link Layer design issues, Error detection – Single parity checking, polynomial codes – CRC, Error correction-Hamming code, Elementary data link protocols, sliding window protocols, Example data link protocols.	10Hrs
UNIT-III	**The Medium Access Control:** The channel allocation problem, multiple access protocols – ALOHA, Slotted ALOHA, CSMA protocols, Collision free protocols, Ethernet, Wireless LAN, Bluetooth.	10Hrs
UNIT-IV	**The network Layer:** Network layer design issues, Routing algorithms – Flooding, Distance vector routing , Hierarchical routing, Link state routing, Congestion control algorithms – Leaky bucket, token bucket algorithm, admission control, hop by hop choke packets , Quality of Service.	10Hrs
UNIT-V	**The Transport Layer and Application Layer:** Transport service, Elements of Transport protocols, Internet transport protocols (TCP & UDP), DNS, Electronic Mailing, and World Wide Web.	

Learning Materials: Text Books:

1. Andrew S. Tanenbaum, David J. Wetherall, Computer Networks, Fifth Edition,

Pearson Pub. 2012.
Soft and Hard copy of Notes, References Websites

Assessment
Assessment is carried out as per the guidelines laid down and mandated by the affiliating University.
100 marks exam (20 IA + 80 Semester End Exam)
1. Two Internal Tests (IA): 20marks

Internal Test 1:	20 marks reduced to 04
Internal Test 2:	80 marks reduced to 10
Attendance:	03
Class seminars, Tutorials, Sports & Cultural Activities, Assignments, NSS/NCC:	03

2. Semester End Examination as per University guidelines: 80 marks

Course:B.Sc-VI	**Core Java (Paper – II)** **(Lectures/Week:4)** **Facilitator:Miss G.B Kustigar**

Objectives:

Covers design, implementation and testing software using Java. Introduces how to write Java programs that solve practical, real world, business-oriented problems using object-oriented design techniques.

Learning Outcomes:

- The **model** of object oriented programming: **abstract data types, encapsulation, inheritance** and **polymorphism**
- **Fundamental features** of an object oriented language like Java: **object classes** and **interfaces, exceptions** and libraries of **object collections**
- How to take the statement of a business problem and from this determine **suitable logic for solving the problem;** then be able to proceed to code that logic as a program written in Java.
- Develop software in the Java programming language, (application)
- Evaluate user requirements for software functionality required to decide whether the Java programming language can meet user requirements (analysis)

UNIT-I	**Introduction:** Internet origin and development – internet architecture frame work world wide web. **Introduction to JAVA:** JAVA Evolution, Java History, Java features, How java differs from C and C++, Java and Internet, Java and World Wide Web. Web Browsers, Hardware and Software requirements, Java support system, Java Environment. Overview of JAVA Language: Introduction, Simple Java Program,More of Java, An Application with Two Classes Java Program structure, Java Tokens,Java Statements, Implementing a Java Program, Java Virtual Machine, Command Line Arguments, Programming Style. Constants, Variables and Data Types: Introduction, Constants, Variables, Data Types, Declaration of Variables, Giving Values to Variables, Scope of variables, Symbolic Constants, Type Casting, Getting Values of Variables, Standard Default Values, Operators and Expressions; Introduction, Arithmetic Operators, Relational Operators, Logical Operators, Bitwise Operators, Special Operators, Arithmetic Expressions, Evaluation of Expressions, Precedence of Arithmetic Operators, Type conversion and Associatively,Mathematical Functions. Decision Making and Branching: Introduction, Decision making with if Statement, Simple if Statement, The if…… else Statement, Nesting of if.. else Statement, The else if Ladder, The Switch Statement, The ?: Operator. Decision Making and Looping: Introduction. The while Statement, The do Statement, The for Statement, Jumps in Loops Labeled Loops.	12Hrs
UNIT-II	**Classes, Arrays, Strings and Vectors:** Classes, Objects and Methods: Introduction, Defining a Class, Adding Variables, Adding methods, Creating Objects,Accessing Class members, Constructors, Methods Overloading, Static members,nesting of Methods, Inheritance: Extending a Class Overriding Methods, Final Variables and methods, Finalizer methods, Abstract methods and Classes, Visibility Control. Arrays, Strings and Vectors: Arrays, One – Dimensional Arrays, Creating an Array, Two – dimensional Arrays, Strings, Vectors, Wrapper Classes.	10Hrs

UNIT-III	**Interfaces, Packages and Multithreaded Programming: Interfaces:** Multiple Inheritance: Introduction, Defining Interfaces, Extending Interfaces, Implementing Interfaces, Accessing Interface Variable. **Packages:** Putting Classes together: Introduction, Java API Package, Using System Packages, Naming Conventions,Creating Packages, Accessing a Packages, Using a Packages, Adding a Class to a Package, Hiding Classes. **Multithreaded Programming:** Introduction, Creating Threads, Extending the ThreadMethods, Thread Exceptions, Thread Priority, Synchronization, Implementing the 'Runnable' Interface.	10Hrs
UNIT-IV	**Managing Exceptions, Applet Programming: Managing Errors and Exception:** Introduction, Types of Errors, Exceptions, Syntax of Exception handling Code,Multiple Catch Statements, Using Finally Statement, Throwing Our Own Exceptions,Using Exceptions for Debugging. Applet Programming: Introduction, how Applets Differ from Applications, Preparing to Write Applets, Building Applet Code, Applet Life Cycle, Creating an Executable applet, Designing a Web Page, Applet Tag,Adding Applet to HTML File, running the Applet, More about Applet Tag, Passing Parameters to Applets, Aligning the Display, More About HTML Tags, Displaying Numerical Values, Getting Input from the User.	12Hrs
UNIT-V	**Graphics Programming, Input / Output: Graphics Programming:** Introduction, The Graphics Class, Lines and rectangles, circles and Ellipses, Drawing Arcs, Drawing Polygons, Line Graphs, Using Control Loops in Applets, Drawing Bar Charts. **Managing Input / Output in JAVA:** Introduction, Concept of Streams,Stream Classes, Byte Stream Classes, Character Stream Classes, Using Streams. Other Useful I/O Classes, Using the File Class, Input / Output Exceptions, Creation of Files,Reading/Writing Characters, Reading/Writing Bytes, handling Primitive Data Types, Concatenating and Buffering Files, Interactive Input and Output, Other Stream Classes.	8Hrs

Learning Materials: Text Books:

1. E. Balaguruswamy, Programming with JAVA,A Primer, 4th Edition., TMH (1999), (Chapter 2 – 16)
2. Shishir Gundavaram, CGI Programming on the "World Wide Web, O'Reilly and Associates, (1996). (Chapter 1)

Soft and Hard copy of Notes, References Websites

Assessment

Assessment is carried out as per the guidelines laid down and mandated by the affiliating University.

100 marks exam (20 IA + 80 Semester End Exam)

1. Two Internal Tests (IA): 20marks

Internal Test 1: 20 marks reduced to 04

Internal Test 2: 80 marks reduced to 10

Attendance: 03

Class seminars, Tutorials, Sports & Cultural Activities, Assignments, NSS/NCC: 03

2. Semester End Examination as per University guidelines: 80 marks

Practical

B.Sc-II Semester	**Programming Lab- Data Structures using C** **Practical Hours: 4 Hrs/week** **Facilitators: Miss S.M Hegale,Miss D.B. Patil, Miss G.B Kustigar**
	1.Write a C program to demonstrate the Dynamic Memory Allocation for Structure by reading and printing n student details. 2. Write a C program to read a one dimensional array, print sum of all elements along with inputted array elements using Dynamic Memory Allocation. 3. Write a C program to add two matrices using pointer to an array concept. 4. Write a program to sort array of integers using array of pointers concept. 5. Write a program that takes a file as an argument and counts the total number of lines. Lines are defined as ending with a newline character. Program usage should be count filename.txt and the output should be the line count. 6. Write a C program to read a text file and convert the file contents in capital (upper-case) and write the contents in an output file. 7. Write a C program to find n Fibonacci numbers using recursion. 8. Write a C program to find factorial of any number using recursion. 9. Write a C program to search for an element in an array using Sequential search 10. Write a C program to search for an element in an array using Binary search 11. Write a C program to sort a list of N elements using Bubble sort Technique 12. Write a C program to sort a list of N elements using Merge sort Technique 13. Write a C program to sort a list of N elements using Quick sort Technique 14. Write a C program to sort a list of N elements using Insertion sort Technique 15. Write a C program to demonstrate the working of stack of size N using an array. The elements of the stack may assume to be of type integer or real, the operations to be supported are 1. PUSH 2. POP 3. DISPLAY. The program should print appropriate messages for STACK overflow, Under flow and empty, use separate functions to detect these cases 16. Write a C program to simulate the working of an ordinary Queue using an array. 17.Using dynamic variables and pointers Write a C program to construct a singly linked list. 18. Write a C program to implement stack operations using linked list. 19. Write a C program to evaluate postfix expression using stack. 20. Write a C program to convert infix expression to postfix expression using stack
	Assessment **Practical Examination- 40 Marks Duration - 3 Hours**. Certified Journal is compulsory for appearing Practical Examination Students shall be given two programming assignments taking into consideration of duration of the time allotted to students for writing, typing and executing the programs. Algorithm/program design : 15 Execution : 15 (includes program code correctness and correct execution results) Journal : 05 Viva-Voce : 05
B.Sc-IV Semester	**UNIX Lab** **Practical Hours: 4 Hrs/week** **Facilitators: Miss S.M Hegale,Miss G.B. Kustigar**

B.Sc-VI Semester	1. Write a shell script that accepts any number of arguments and displays them in reverse order 2. Write a shell script to display the calendar for current month with date replace * or ** depending on whether the date is one digit or two digit. 3. Write a shell script to find smallest of three numbers that are read from keyboard. 4. Write a shell script that accepts file name as argument and display its creation time if file exists and if it does not exist then send output error message. 5. Write shell script using expr command to read in a string and display a suitable message if it does not have at least 10 characters. 6. Write a shell script to compute the sum of number passed to it as argument on the command line and display the result. 7. Write a shell script to convert decimal number to binary & hexadecimal. 8. Write a shell script that computes the gross salary of an employee according to rule given below. If basic salary is < 15000 then HRA = 10% of basic and DA = 85% of basic , otherwise HRA = 15% of basic and DA = 90% of basic 9. Write a shell script to illustrate the use of environment variables using case construct. 10. Write a shell script that gets executed and displays the message "good morning", or " good afternoon" or "good Evening" depending upon the user logs in time. 11. Write a shell script that deletes all the lines containing a specific word in one or more file supplied as arguments to it. 12. Write a shell script that accepts two integers and computes the value of first number raised to the power of second number. 13. Write a shell script that accepts that accepts the filename, starting and ending line number as arguments and displays all the lines between them. 14. Write a shell script to display the following patterns: 1 1 1 2 2 2 1 2 3 3 3 3 1 2 3 4 4 4 4 4 15. Write a shell script that accept one or more filenames as arguments and converts them into uppercase. 16. Write a shell script to sort an array in ascending order. **Computer Networks Lab** 1) Programs using TCP Sockets (like date and time server & client, echo server & Client, etc...) 2) Programs using UDP Sockets (like simple DNS) 3) Programs using raw sockets (like packet capturing and filtering) 4) Programs using RPC 5) Simulation of sliding window protocols 6) Experiments using simulators (like OPNET) 7) Performance comparison of MAC protocols 8) Performance comparison of Routing protocols 9) Study of TCP/UDP performance
B.Sc-VI Semester	**Core Java Lab** **Journal programs** 1. Program to demonstrate typecasting and type promotions in java. 2. Program to implement all bitwise operations by reading the input by user and display input and output errors.

	3. Program to demonstrate method overloading. 4. Program to implement at least 10 string operations on Strings. 5. Program to demonstrate multilevel inheritance. Show the usage of super (). 6. Program to demonstrate method overriding and dynamic method dispatch. 7. Program to demonstrate constructor overloading by passing different number of parameters of different types. 8. Program to demonstrate a) Packages b) Interfaces. 9. Program to illustrate the usage of try, catch, throws and finally to show exception handling in java. 10. Program to show thread synchronization by creating threads using runnable interface. 11. Program to demonstrate thread priorities. Create the thread by extending thread class. 12. Program to create student report using applet, read the input using text boxes and generate the grades. 13. Program to demonstrate a) Abstract class b) Inner class 14. Program to demonstrate drawing bar chart in applets using graphics programming. 15. Program to copy bytes from one file to another. 16. Program to implement mouse events.
	Assessment: Evaluation criteria for practical examinations shall be as follows: **1. Writing of Programs -15 Marks** a. One program from the journal list – 08 Marks b. Another program given by examiner based on the concepts studied -07Marks **2. Execution of programs – 15 Marks** a. Journal Program - 08 Marks b. Program of Examiner’s Choice -07 Marks **2. Viva-Voce -05 Marks** **4. Journal / Laboratory Report – 5 Marks** **Total Marks -40 Marks**