UNIVERSITY OF KERALA

B. TECH. DEGREE COURSE

(2013 SCHEME)

SYLLABUS FOR

V SEMESTER

COMPUTER SCIENCE & ENGINEERING

SCHEME -2013

V SEMESTER

COMPUTER SCIENCE & ENGINEERING (R)

Course No	Name of subject	Credits	Weekly load, hours			C A Marks	Exam Duration Hrs	U E Max Marks	Total Marks
			L	T	D/ P				
13.501	Engineering Mathematics IV (ABFHRT) (Complex Analysis and Linear Algebra)	4	3	1	-	50	3	100	150
13.502	Engineering Mathematics- V (FR) (Advanced Mathematics and Queueing Models)	4	3	1	-	50	3	100	150
13.503	Operating Systems (FR)	3	2	1	-	50	3	100	150
13.504	Systems Programming (FR)	3	2	1	-	50	3	100	150
13.505	Microprocessors and Interfacing(R)	4	2	2	-	50	3	100	150
13.506	Object Oriented Design and JAVA Programming(R)	3	2	1	-	50	3	100	150
13.507	Object Oriented Programming Lab (R)	4	-	-	4	50	3	100	150
13.508	Application Software Development Lab(R)	4	-	-	4	50	3	100	150
	Total	**29**	**14**	**7**	**8**	**400**		**800**	**1200**

13.501 ENGINEERING MATHEMATICS – IV (ABFHRT)

(COMPLEX ANALYSIS AND LINEAR ALGEBRA)

Teaching Scheme: 3(L) - 1(T) - 0(P) **Credits**: 4

Course Objective:

- *To introduce the basic notion in complex analysis such as Analytic Functions, Harmonic functions and their applications in fluid mechanics and differentiations and integration of complex functions ,transformations and their applications in engineering fields*.
- *Many fundamental ideas of Linear Algebra are introduced as a part of this course. Linear transformations provide a dynamic and graphical view of matrix-vector multiplication. Orthogonality plays an important role in computer calculations.*

Module – I

Complex Differentiation: Limits, continuity and differentiation of complex functions. Analytic functions – Cauchy Riemann equations in Cartesian form (proof of necessary part only).Properties of analytic functions – harmonic functions. Milne Thomson method.

Conformal mapping: Conformality and properties of the transformations w $= \frac{1}{z}$, w = z^2, w $= z + \frac{1}{z}$, w = sin z , w = e^z - Bilinear transformations.

Module – II

Complex Integration: Line integral – Cauchy's integral theorem – Cauchy's integral formula – Taylor's and Laurent's series – zeros and singularities – residues and residue theorem. Evaluation of real definite integrals – $\int_0^{2\pi} f(sinx, cosx)dx$,$\int_{-\infty}^{\infty} f(x)dx$ (with no poles on the real axis). (Proof of theorems not required).

Module – III

Vector spaces and subspaces- Null spaces, Column spaces and linear transformations-Kernal and range of a linear transformation -Linearly independent sets-Bases –Bases for nulA and ColA-Co-ordinate systems -Dimension of vector space -Rank -Change of basis.

Module – IV

Inner product spaces -Length and orthogonality -Orthogonal sets-Orthogonal and orthonormal bases -Orthogonal projection -Gram-Schmidt process -Least square problem - Quadratic forms- Constrained optimization of quadratic forms -Singular value decomposition (proof of the theorem are not included).

References:

1. O’Neil P. V., Advanced Engineering Mathematics, Cengage Learning, 2011.
2. Kreyszig E., *Advanced Engineering Mathematics*, 9/e, Wiley India, 2013.
3. Grewal B. S., *Higher Engineering Mathematics*, 13/e, Khanna Publications, 2012.
4. Lay D. C., Linear Algebra with Applications, 3/e, Pearson Education, 2006.
5. Bronson R. and G. B. Costa, Linear Algebra-an introduction, Elsevier Academic Press, 2007.
6. Williams G., Linear Algebra with Applications, Jones and Bartlett Learning, 2012.
7. Strang G., Linear Algebra with Applications, Thomson Learning, 2006.

Internal Continuous Assessment *(Maximum Marks-50)*

50% - Tests (minimum 2)
30% - Assignments (minimum 2) such as home work, problem solving, quiz, literature survey, seminar, term-project, software exercises, etc.
20% - Regularity in the class

University Examination Pattern:

Examination duration: 3 hours Maximum Total Marks: 100

The question paper shall consist of 2 parts.

Part A (20 marks) - Five Short answer questions of 4 marks each. All questions are compulsory. There should be at least one question from each module and not more than two questions from any module.

Part B (80 Marks) - Candidates have to answer one full question out of the two from each module. Each question carries 20 marks.

Course Outcome:

After successful completion of this course, the students master the basic concepts of complex analysis and linear algebra which they can use later in their career.

13.502 ENGINEERING MATHEMATICS - V (FR)
(ADVANCED MATHEMATICS AND QUEUEING MODELS)

Teaching Scheme: 3(L) - 1(T) - 0(P) **Credits**: 4

Course Objectives:

- *Mathematical programming techniques are introduced as a part of this course. These techniques are concerned with the allotment of available resources so as to minimize cost or maximize profit subject to prescribed restrictions.*
- *The study of queueing models provides us methods to minimize the sum of cost of providing service and cost of obtaining service which are primarily associated with the value of time spent by the customer in a queue.*
- *Network models such as PERT and CPM are introduced as a part of this course which are used for planning, scheduling and controlling complex projects.*

Module – I

Linear programming-Formation of LPP-Graphical solution-General linear programming problem-Slack and surplus variables-Standard forms –solution of LPP-Basic solution –Basic feasible solution- Degenerate and non-degenerate solutions-Optimal solution-solution by simplex method - Artificial variables – Big-M method.

Module – II

Duality in LPP - Properties of primal and dual optimal solutions –solution using Duality-Transportation problem and Assignment problem-Travelling salesman problem.

Module – III

Network Analysis-Project scheduling –construction of project networks-critical path method(CPM)-Identification of critical path using CPM-Estimation of Floats-Total float-Free float-Independent float-Project evaluation and review technique (PERT)-Computation of expected completion times by PERT.

Module – IV

Queueing Theory-Queues –Characteristic of queues-Random arrivals-Arrival and Departure Distributions-Types of queues-Little’s Formulae-Basic queueing models – M/M/I:∞/FIFO,M/M/C:∞/FIFO,M/M/I:K/FIFO-Basic queue characteristics of models.

References

1. KantiSwarup, Manmohan, *Operations Research*, S. Chand and Sons, (6th edition).

2. Hadly G., *Linear Programming*, Addison Wesley.
3. Ravindran, Philips, Solberg, *Operations Research*, Wiley.
4. S. D. Sharma, *Operations Research*, Wiley Publications.

Internal Continuous Assessment *(Maximum Marks-50)*

50% - Tests (minimum 2)

30% - Assignments (minimum 2) such as home work, problem solving, literature survey, seminar, term-project, software exercises, etc.

20% - Regularity in the class

University Examination Pattern:

Examination duration: 3 hours *Maximum Total Marks: 100*

The question paper shall consist of 2 parts.

Part A (20 marks) - Five Short answer questions of 4 marks each. All questions are compulsory. There should be at least one question from each module and not more than two questions from any module.

Part B (80 Marks) - Candidates have to answer one full question (question may contain sub-divisions), out of the two from each module. Each question carries 20 marks.

Course outcome:

At the end of the course, the student will be familiar with large scale applications of operations research techniques computations which require only few minutes on the computer.

13.503 OPERATING SYSTEMS (FR)

Teaching Scheme: 2(L) - 1(T) - 0(P) **Credits**: 3

Course Objectives:

To provide an understanding of concepts those underlie operating systems.

Module – I

Introduction: Basic concepts – terminology. Historical perspective - early system - simple monitor - performance - types of OS - batch processing - multiprogramming - time sharing - real time system - different classes of computers - functions and components of an operating system - OS structure - Multiprocessor system - distributed system. Operating system services. ***Information management:*** File concepts file support - file system - directory structure - gaining access to files - basic file system calls - sharing and security - operation on files - file protection - allocation methods - implementation issues – case study.

Module – II

Processor management : CPU scheduling - Review of Multiprogramming concepts - scheduling concepts - scheduling algorithm - Multiprocessor scheduling , Concurrent process synchronization. Critical section problem - Semaphores - process coordination.

Interprocess communication – pipes, shared files, shared memory, message based IPC.

Module – III

Memory management : Preliminaries - Memory architecture evolution - Resident monitor - Swapping – fixed partitions - variable partitions - paging - segmentation - combined system - virtual memory concepts - overlay - demand paging - page replacement - space allocation policies - segmented paging – dynamic linking ***.Device management :*** Physical characteristics – FCFS, SSF, C–SCAN - selecting a disk scheduling algorithm . I/O scheduling policies - terminal I/O handling - channels and control units -virtual devices.

Module – IV

Dead locks : Dead lock problem - characteristics - prevention - avoidance - detection - Recovery from dead lock - combined approach to dead lock handling. ***Protection :*** Goals of protection sms and policies - domain of protection - access matrix and its implementation. Dynamic protection structures, security.

References:

1. Peterson J. L. and A. Silberschats, *Operating System Concepts*, Addison Wesley.

2. Bhatt P. C. P., *An Introduction to Operating Systems Concepts and Practice* (3rd Edition), PHI.[Chapter -7]
3. Madnick S. and J. J. Donovan, *Operating Systems*, McGraw Hill.
4. Hanson P. B., *Operating System Principle*, Prentice Hall of India.
5. Shaw A. C., *The Logical Design of Operating Systems*, Prentice Hall.
6. Deite H. M., *Operating System Principles*, Addison-Wesley.

Internal Continuous Assessment *(Maximum Marks-50)*

50% - Tests (minimum 2)

30% - Assignments (minimum 2) such as home work, problem solving, quiz, literature survey, seminar, term-project, software exercises, etc.

20% - Regularity in the class

University Examination Pattern:

Examination duration: 3 hours *Maximum Total Marks: 100*

The question paper shall consist of 2 parts.

Part A (20 marks) - Five Short answer questions of 4 marks each. All questions are compulsory. There should be at least one question from each module and not more than two questions from any module.

Part B (80 Marks) - Candidates have to answer one full question (question may contain sub-divisions), out of the two from each module. Each question carries 20 marks.

Note: *The question paper shall contain at least 50% analytical/problem solving questions.*

Course Outcome:

After successful completion of this course, the student will be able to understand how operating system works in the background and makes the user interact with the machine.

13.504 SYSTEM PROGRAMMING (FR)

Teaching Scheme: 2(L) - 1(T) - 0(P) **Credits**: 3

Course Objective:

- *To impart the basic concepts of system software design.*
- *To equip the student with the right kind of tools for computer systems design and development.*

Pre-requisites: 13.402 - Computer Organisation and Design
13.306 - Data Structures and Algorithms.

Module – I

Systems Programming – Background, System software and Machine architecture. SIC & SIC/XE Architecture and Programming. Traditional (CISC) machines – VAX architecture, Pentium Pro architecture, RISC machine – Ultra SPARK, Power PC.

Module – II

Assemblers – Basic assembler functions, machine dependent assembler features, machine independent assembler features, Hand assembly of SIC/XE programming. Assembler design options – one pass assembler, multi pass assembler. Assembler implementation – MASM, SPARC assemblers, Assemblers Vs Compilers and Interpreters.

Module – III

Loaders and Linkers - Basic loader functions, machine dependent loader features, machine independent loader features. Loader design options – linkage editors, dynamic linking, bootstrap loaders. Loader Implementation – DOS linker.

Macro processors – Basic macro processor functions, machine dependent and machine independent macro processor features, Design options. Macro implementation – MASM, ANSI C macro processors.

Module – IV

Text Editors – overview of the editing process, user interface, editor structure. Debuggers – Overview of Debugger features, Breakpoint mechanism, Hardware support for debugging, Context of Debugger Check pointing and Reverse Execution.

General overview of the UNIX operating system - history of UNIX - system structure - user perspective - services - hardware assumptions - unix architecture - system concepts - kernel data structures - system administration process (concepts only).

References:

1. Beck L.L., System Software - An introduction to Systems Programming, 3/e, Pearson Education.
2. Santanu Chattopadhyay, System Software, Prentice Hall of India.
3. Bach M. J., The Design of the Unix Operating System, Prentice Hall India.
4. John J. Donovan, Systems Programming,1/e, Tata McGraw Hill.
5. Damdhare, Operating Systems and Systems Programming, 2/e, Tata McGraw Hill.
6. Godbole S., Operating Systems, 3/e, Tata McGraw Hill.

Internal Continuous Assessment *(Maximum Marks-50)*

50% - Tests (minimum 2)

30% - Assignments (minimum 2) such as home work, problem solving, quiz, literature survey, seminar, term-project etc.

20% - Regularity in the class

University Examination Pattern:

Examination duration: 3 hours *Maximum Total Marks: 100*

The question paper shall consist of 2 parts.

Part A (20 marks) - Five Short answer questions of 4 marks each. All questions are compulsory. There should be at least one question from each module and not more than two questions from any module.

Part B (80 Marks) - Candidates have to answer one full question (question may contain sub-divisions), out of the two from each module. Each question carries 20 marks.

Note: *The question paper shall contain at least 30% analytical/problem solving questions.*

Course Outcome:

After the successful completion of the course students will be able to:

- *Design and develop various system softwares.*
- *Take more advanced software courses.*
- *Self learn advance features in system softwares.*

13.505 MICROPROCESSORS AND INTERFACING (R)

Teaching Scheme: 2(L) - 2(T) - 0(P) **Credits:** 4

Course Objectives:

- *To impart knowledge on the basics of Microprocessor Architecture*
- *To acquire knowledge on the concepts of Peripheral Interfacing*
- *To develop assembly language Programming skills*

Pre-requisites: 13.402 - Computer Organization
13.305 - Digital System Design.

Module – I

Introduction to Microprocessors, Microprocessor operations, Bus structure, Memory – Memory map and address decoding, ROM/RAM/Port decoder

Intel 8085 Microprocessor – Internal Architecture, Signals, Simple Data transfer instructions.

Module – II

8085 Bus cycles and Timing – 8085 bus activities during a read/write machine cycle, Basic I/O interfacing—peripheral I/O and Memory mapped I/O, Addressing modes, Interrupt handling.

Intel 8086 Microprocessor – Internal architecture, Signals and System connections, Memory System.

Module – III

Programming 8086-- Addressing modes, Assembler Directives, Instruction set, Assembly Language Programming. Interrupts and interrupt applications.

Module – IV

Interfacing 8086- 8255 Programmable Peripheral Interface , 8237 DMA controller, , 8279 key board/ display interfacing, 8259A Priority Interrupt controller, 8254 software programmable timer/counter (Programming not included).

References:

1. Ramesh S Gaonkar, *Microprocessor Architecture, Programming and Applications with the 8085*, Penram International Publishing (India) 5th edn (Chapters 3, 5, 6, 7, 8, 4)
2. Douglas V Hall, *Microprocessors & Interfacing - Programming and Hardware*, Tata McGraw-Hill 3rd Edn (Chapters 2, 3, 4, 5, 6, 8, 9)

3. Ray A K & K M Bhurchandi, *Advanced Microprocessors & Peripherals*, 3rd edition
4. Mathivanan N, *Microprocessors, PC Hardware and Interfacing* –PHI Learning Pvt. Ltd.
5. Udayakumar K, B S Umashankar, *The 8085 Microprocessors , Architecture, Programming and Interfacing* –, Pearson
6. Mukopadhyaya A. K., *Microprocessor, Microcomputer and Applications*, Narosa ,3rd Edn.

Internal Continuous Assessment *(Maximum Marks-50)*

50% - Tests (minimum 2)

30% - Assignments (minimum 2) such as class room/home work, problem solving, quiz, literature survey, seminar, term-project, software exercises, etc.

20% - Regularity in the class

University Examination Pattern:

Examination duration: 3 hours *Maximum Total Marks: 100*

The question paper shall consist of 2 parts.

Part A (20 marks) - Five Short answer questions of 4 marks each. All questions are compulsory. There should be at least one question from each module and not more than two questions from any module.

Part B (80 Marks) - Candidates have to answer one full question (question may contain sub-divisions), out of the two from each module. Each question carries 20 marks.

Note: *The question paper shall contain at least 50% analytical/problem solving questions.*

Course Outcome:

After successful completion of this course,

- *Attain a thorough understanding of 8 bit and 16 bit microprocessor architecture.*
- *Attain ability to design interfacing external devices with a microprocessor.*
- *Ability to develop programs in assembly language.*

13.506 OBJECT ORIENTED DESIGN AND JAVA PROGRAMMING (R)

Teaching Scheme: 2(L) - 1(T) - 0(P) **Credits**: 3

Course Objective:

- *To impart the basic concepts of Object Oriented Design Techniques.*
- *To develop a thorough understanding of Java language.*
- *To study the techniques of creating GUI based applications.*

Pre-requisites: **13.403- Object Oriented Techniques**

Module – I

Review of Object Oriented Concepts – Object Oriented Systems Development Life cycle- Object Oriented Methodologies – Rumbaugh methodology – Booch methodology – Jacobson et. al methodology – Patterns – Frameworks – Unified Approach - Unified Modeling Language – Static and Dynamic Models – UML diagrams – UML Class Diagram – Use-Case Diagram.

Module – II

Java Overview – Java Virtual Machine – Introduction to Java Programming. Classes and objects – Constructors – Access Modifiers – Parameter Passing. Inheritance – Abstract classes and Interfaces. Polymorphism – Method overriding and overloading. Packages in Java – defining and importing packages. Wrapper classes. String Handling – String and StringBuffer class. Exception Handling – use of *try, catch, throw, throws* and *finally* – nested try statements – user defined exception.

Module – III

Generics – Generic class – Bounded types – Generic interfaces. Input/Output in Java – Files – Stream classes – Serialization – Reading console input. Collection framework. Threads – Thread class and Runnable interface – Thread synchronization and priorities – Multithreading. Networking basics – communication using Stream sockets and Datagram sockets. Applets – Applet basics – lifecycle - Passing Parameters to Applets.

Module – IV

Event Handling – Delegation Event Model – Event Classes – Sources – Listener Interfaces. Introduction to AWT – Working with Frames, Graphics, Color, Font. AWT Controls – Label, Button, CheckBox, Choice, List, TextField, TextArea – Layout Managers. Swing overview – Creating simple GUI applications using Swing. Java database Connectivity – JDBC overview – Types of Statement – Creating and executing queries – Dynamic queries. Introduction to Java Beans.

References:-

1. Herbert Schildt, Java: The Complete Reference, 8th Edn –TMH.
2. Ali Bahrami, Object Oriented Systems Development using the Unified Modeling Language –McGraw Hill.
3. David Flanagan, Java in a Nutshwell, 5th Edn - O'Reilly.
4. R. Nageswara Rao, Core Java: An Integrated Approach –Dreamtech Press.
5. K. Barclay, J. Savage, Object Oriented Design with UML and Java –Elsevier Publishers.
6. Kathy Sierra, Head First Java, 2nd Edn –O'Reilly.
7. E. Balagurusamy, Programming JAVA a Primer, 4th Edn –TMH.

Internal Continuous Assessment *(Maximum Marks-50)*

50% - Tests (minimum 2)

30% - Assignments (minimum 2) such as class room/home work, problem solving, quiz, literature survey, seminar, term-project, software exercises, etc.

20% - Regularity in the class

University Examination Pattern:

Examination duration: 3 hours *Maximum Total Marks: 100*

The question paper shall consist of 2 parts.

Part A (20 marks) - Five Short answer questions of 4 marks each. All questions are compulsory. There should be at least one question from each module and not more than two questions from any module.

Part B (80 Marks) - Candidates have to answer one full question (question may contain sub-divisions), out of the two from each module. Each question carries 20 marks.

Note: *The question paper shall contain at least 60% analytical/problem solving questions.*

Course Outcome:

After successful completion of this course, students will be able to

- *Implement object oriented principles for reusability.*
- *Assign priorities and resolve run-time errors with Multithreading and Exception Handling techniques.*
- *Interpret Events handling techniques for interaction of the user with GUI.*
- *Analyze JDBC drivers to connect Java applications with relational databases.*
- *Develop client/server applications using socket programming.*

13.507 OBJECT ORIENTED PROGRAMMING LAB (R)

Teaching Scheme: 0(L) - 0(T) - 4(P) **Credits**: 4

Course Objective :

- *To acquaint students with Object Oriented concepts and terminology.*
- *To design and implement object oriented software to solve moderately complex problems.*

List of Exercises:

Programming exercises based on the course Object Oriented Techniques. The exercises may include the following:-

1. Functions
    a. Call by value, Call by reference, Call by name, return by reference
    b. Function overloading
    c. Default arguments

2. Classes and Objects
    a. Classes with primitive data members, arrays and pointers as data members
    b. Classes with static data members and static member functions
    c. Arrays of objects, objects as function arguments, returning objects
    d. Constructors and destructors - Parameterized constructor, copy constructor etc.
    e. Friend functions and classes

3. Operator overloading
    a. Overloading unary and binary operators
    b. Overloading using Friend functions

4. Inheritance
    a. Single, multiple, multilevel and hierarchical inheritance, Constructors in derived classes
    b. Virtual base classes , abstract classes
    c. Virtual functions

5. File handling & Templates
    a. Basic file operations
    b. Function templates and Class Templates

Internal Continuous Assessment *(Maximum Marks-50)*

40% - Test

40% - Class work and Record (Up-to-date lab work, problem solving capability, keeping track of rough record and fair record, term projects, assignment, software/hardware exercises, etc.)

20% - Regularity in the class

University Examination Pattern:

Examination duration: 3 hours *Maximum Total Marks: 100*

Questions based on the list of exercises prescribed.

Marks should be awarded as follows:

20% - Algorithm/Design

30% - Implementing / Conducting the work assigned

25% - Output/Results and inference

25% - Viva voce

Candidate shall submit the certified fair record for endorsement by the external examiner.

Course Outcome:

After successful completion of this course, students will be able to:

- *familiarize classes and attributes in real world applications.*
- *Perform programs using OOP concepts.*
- *Distinguish the types of inheritance in different problems.*
- *Perform applications by overloading operators and functions.*
- *Use virtual functions and ABC for problem solving.*

13.508 APPLICATION SOFTWARE DEVELOPMENT LAB (R)

Teaching Scheme: 0(L) - 0(T) - 4(P) **Credits**: 4

Course Objective :

- *To acquaint students with DDL , DML and DCL statements for database manipulation.*
- *To make them build full fledged database applications using JDBC.*

Programming exercises based on the courses 13.405 Data Base Design and 13.506 Object Oriented Design and JAVA Programming will be covered in this subject.

The exercises may include the following so that the students get trained in

(i) Practicing database commands

(ii) Developing GUI based application using database.

List of Exercises:

1. Familiarization of creation of databases, SQL commands (DDL, DML & DCL) and group functions to access data from the database. Suitable exercises to practice SQL commands in the above category may be given.
2. Creation of views, indexes, sequences.
3. Security management using SQL – granting and revoking privileges.
4. SQL procedures and Functions.
5. SQL cursors, triggers, and packages.
6. Exception handling in SQL.
7. Importing and exporting of databases using SQL.
8. Develop a menu driven, GUI based user friendly database application in any one of the domains such as Banking, Electricity Billing, Library management, Payroll, Insurance, Inventory, Health care etc. integrating all the features specified in the above exercise.

Note: A report containing analysis and design for the above database application should be included in the laboratory record immediately after the write up for the programming exercises.

Internal Continuous Assessment *(Maximum Marks-50)*

40% - Test

40% - Class work and Record (Up-to-date lab work, problem solving capability, keeping track of rough record and fair record, term projects, assignment, software/hardware exercises, etc.)

20% - Regularity in the class

University Examination Pattern:

Examination duration: 3 hours *Maximum Total Marks: 100*

A complete GUI based database application incorporating one/more features listed in the exercises above will be used to test the students' knowledge in the topic. Students have to demonstrate the database application softwares developed by them (the 6th exercise) as part of the viva voce.

Marks should be awarded as follows:

45% - Implementing / Conducting the work assigned

25% - Output/Results and inference

30% - Viva voce (30% weightage should be given to Exercise No. 6)

Candidate shall submit the certified fair record for endorsement by the external examiner.

Course Outcome:

After successful completion of this course, students will be

- *Familiar with SQL queries using oracle database.*
- *able to use PLSQL to handle queries in procedures.*
- *Familiar with java programming language.*
- *able to design and code GUI applications using Netbeans.*