

ANNA UNIVERSITY, CHENNAI

AFFILIATED INSTITUTIONS

R-2008

www.AnnaUnivEdu.Org

<http://facebook.com/annaunivedu.info>

<http://twitter.com/annauniversity4>

Syllabus Semester wise separated & shared by : www.annaunivedu.org

**B.E. COMPUTER SCIENCE AND ENGINEERING
VI SEMESTERS CURRICULA AND SYLLABI**

SEMESTER VI

(Applicable to the students admitted from the Academic year 2008–2009 onwards)

CODE NO.	COURSE TITLE	L	T	P	C
THEORY					
CS2351	<u>Artificial Intelligence</u>	3	0	0	3
CS2352	<u>Principles of Compiler Design</u>	3	0	2	4
CS2353	<u>Object Oriented Analysis and Design</u>	3	0	0	3
CS2354	<u>Advanced Computer Architecture</u>	3	0	0	3
	Elective – I	3	0	0	3
	Elective – II	3	0	0	3
PRACTICAL					
CS2357	<u>Object Oriented Analysis and Design Lab</u>	0	0	3	2
GE2321	<u>Communication Skills Lab</u>	0	0	4	2
CS2358	<u>Internet Programming Lab</u>	1	0	3	2
TOTAL		19	0	12	25

CS2351

ARTIFICIAL INTELLIGENCE

L T P C
3 0 0 3

AIM:

To learn the basics of designing intelligent agents that can solve general purpose problems, represent and process knowledge, plan and act, reason under uncertainty and can learn from experiences

UNIT I PROBLEM SOLVING 9

Introduction – Agents – Problem formulation – uninformed search strategies – heuristics – informed search strategies – constraint satisfaction

UNIT II LOGICAL REASONING 9

Logical agents – propositional logic – inferences – first-order logic – inferences in first-order logic – forward chaining – backward chaining – unification – resolution

UNIT III PLANNING 9

Planning with state-space search – partial-order planning – planning graphs – planning and acting in the real world

UNIT IV UNCERTAIN KNOWLEDGE AND REASONING 9

Uncertainty – review of probability - probabilistic Reasoning – Bayesian networks – inferences in Bayesian networks – Temporal models – Hidden Markov models

UNIT V LEARNING**9**

Learning from observation - Inductive learning – Decision trees – Explanation based learning – Statistical Learning methods - Reinforcement Learning

TOTAL: 45 PERIODS**TEXT BOOK:**

1. S. Russel and P. Norvig, "Artificial Intelligence – A Modern Approach", Second Edition, Pearson Education, 2003.

REFERENCES:

1. David Poole, Alan Mackworth, Randy Goebel, "Computational Intelligence : a logical approach", Oxford University Press, 2004.
2. G. Luger, "Artificial Intelligence: Structures and Strategies for complex problem solving", Fourth Edition, Pearson Education, 2002.
3. J. Nilsson, "Artificial Intelligence: A new Synthesis", Elsevier Publishers, 1998.

CS2352**PRINCIPLES OF COMPILER DESIGN****L T P C****3 0 2 4****UNIT I LEXICAL ANALYSIS****9**

Introduction to Compiling- Compilers-Analysis of the source program-The phases-Cousins-The grouping of phases-Compiler construction tools. The role of the lexical analyzer- Input buffering-Specification of tokens-Recognition of tokens-A language for specifying lexical analyzer.

UNIT II SYNTAX ANALYSIS and RUN-TIME ENVIRONMENTS**9**

Syntax Analysis- The role of the parser-Context-free grammars-Writing a grammar-Top-down parsing-Bottom-up Parsing-LR parsers-Constructing an SLR(1) parsing table. Type Checking- Type Systems-Specification of a simple type checker. Run-Time Environments-Source language issues-Storage organization-Storage-allocation strategies.

UNIT III INTERMEDIATE CODE GENERATION**9**

Intermediate languages-Declarations-Assignment statements - Boolean expressions-Case statements- Backpatching-Procedure calls

UNIT IV CODE GENERATION**9**

Issues in the design of a code generator- The target machine-Run-time storage management-Basic blocks and flow graphs- Next-use information-A simple code generator-Register allocation and assignment-The dag representation of basic blocks - Generating code from dags.

UNIT V CODE OPTIMIZATION**9**

Introduction-The principle sources of optimization-Peepphole optimization- Optimization of basic blocks-Loops in flow graphs- Introduction to global data-flow analysis-Code improving transformations.

TOTAL:45 PERIODS**TEXT BOOK:**

1. Alfred V. Aho, Ravi Sethi Jeffrey D. Ullman, "Compilers- Principles, Techniques, and Tools", Pearson Education Asia, 2007.

REFERENCES:

1. David Galles, "Modern Compiler Design", Pearson Education Asia, 2007
2. Steven S. Muchnick, "Advanced Compiler Design & Implementation", Morgan Kaufmann Pulishers, 2000.
3. C. N. Fisher and R. J. LeBlanc "Crafting a Compiler with C", Pearson Education, 2000.

CS2353**OBJECT ORIENTED ANALYSIS AND DESIGN****L T P C****3 0 0 3****OBJECTIVES:**

- To learn basic OO analysis and design skills through an elaborate case study
- To use the UML design diagrams
- To apply the appropriate design patterns

UNIT I**9**

Introduction to OOAD – What is OOAD? – What is UML? What are the United process(UP) phases - Case study – the NextGen POS system, Inception -Use case Modeling - Relating Use cases – include, extend and generalization.

UNIT II**9**

Elaboration - Domain Models - Finding conceptual classes and description classes – Associations – Attributes – Domain model refinement – Finding conceptual class hierarchies- Aggregation and Composition- UML activity diagrams and modeling

UNIT III**9**

System sequence diagrams - Relationship between sequence diagrams and use cases Logical architecture and UML package diagram – Logical architecture refinement - UML class diagrams - UML interaction diagrams

UNIT IV**9**

GRASP: Designing objects with responsibilities – Creator – Information expert – Low Coupling –Controller – High Cohesion – Designing for visibility - Applying GoF design patterns – adapter, singleton, factory and observer patterns.

UNIT V **9**
UML state diagrams and modeling - Operation contracts- Mapping design to code -UML deployment and component diagrams

TOTAL : 45 PERIODS

TEXT BOOK :

1. Craig Larman, "Applying UML and Patterns: An Introduction to object-oriented Analysis and Design and iterative development", Third Edition, Pearson Education, 2005

REFERENCES:

1. Mike O'Docherty, "Object-Oriented Analysis & Design: Understanding System Development with UML 2.0", John Wiley & Sons, 2005.
2. James W- Cooper, Addison-Wesley, "Java Design Patterns – A Tutorial", 2000.
3. Micheal Blaha, James Rambaugh, "Object-Oriented Modeling and Design with UML", Second Edition, Prentice Hall of India Private Limited, 2007
4. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, "Design patterns: Elements of Reusable object-oriented software", Addison-Wesley, 1995.

CS2354 **ADVANCED COMPUTER ARCHITECTURE** **L T P C**
3 0 0 3

UNIT I **INSTRUCTION LEVEL PARALLELISM** **9**
ILP – Concepts and challenges – Hardware and software approaches – Dynamic scheduling – Speculation - Compiler techniques for exposing ILP – Branch prediction.

UNIT II **MULTIPLE ISSUE PROCESSORS** **9**
VLIW & EPIC – Advanced compiler support – Hardware support for exposing parallelism – Hardware versus software speculation mechanisms – IA 64 and Itanium processors – Limits on ILP.

UNIT III **MULTIPROCESSORS AND THREAD LEVEL PARALLELISM** **9**
Symmetric and distributed shared memory architectures – Performance issues – Synchronization – Models of memory consistency – Introduction to Multithreading.

UNIT IV **MEMORY AND I/O** **9**
Cache performance – Reducing cache miss penalty and miss rate – Reducing hit time – Main memory and performance – Memory technology. Types of storage devices – Buses – RAID – Reliability, availability and dependability – I/O performance measures – Designing an I/O system.

UNIT V **MULTI-CORE ARCHITECTURES** **9**
Software and hardware multithreading – SMT and CMP architectures – Design issues – Case studies – Intel Multi-core architecture – SUN CMP architecture - heterogenous multi-core processors – case study: IBM Cell Processor.

TOTAL : 45 PERIODS

TEXT BOOK:

1. John L. Hennessey and David A. Patterson, "Computer architecture – A quantitative approach", Morgan Kaufmann / Elsevier Publishers, 4th. edition, 2007.

REFERENCES:

1. David E. Culler, Jaswinder Pal Singh, "Parallel computing architecture : A hardware/software approach" , Morgan Kaufmann /Elsevier Publishers, 1999.
2. Kai Hwang and Zhi.Wei Xu, "Scalable Parallel Computing", Tata McGraw Hill, New Delhi, 2003.

CS2357**OBJECT ORIENTED ANALYSIS AND DESIGN LAB****L T P C
0 0 3 2****OBJECTIVES:****To develop a mini-project following the 12 exercises listed below.**

1. To develop a problem statement.
2. Develop an IEEE standard SRS document. Also develop risk management and project plan (Gantt chart).
3. Identify Use Cases and develop the Use Case model.
4. Identify the business activities and develop an UML Activity diagram.
5. Identify the conceptual classes and develop a domain model with UML Class diagram.
6. Using the identified scenarios find the interaction between objects and represent them using UML Interaction diagrams.
7. Draw the State Chart diagram.
8. Identify the User Interface, Domain objects, and Technical services. Draw the partial layered, logical architecture diagram with UML package diagram notation.
9. Implement the Technical services layer.
10. Implement the Domain objects layer.
11. Implement the User Interface layer.
12. Draw Component and Deployment diagrams.

Suggested domains for Mini-project.

1. Passport automation system.
2. Book bank
3. Exam Registration
4. Stock maintenance system.
5. Online course reservation system
6. E-ticketing
7. Software personnel management system
8. Credit card processing
9. e-book management system
10. Recruitment system
11. Foreign trading system
12. Conference Management System
13. BPO Management System

Suggested SoftwareTools

1. ArgoUML, Eclipse IDE, Visual Paradigm, Visual case, and Rational Suite

Globalisation has brought in numerous opportunities for the teeming millions, with more focus on the students' overall capability apart from academic competence. Many students, particularly those from non-English medium schools, find that they are not preferred due to their inadequacy of communication skills and soft skills, despite possessing sound knowledge in their subject area along with technical capability. Keeping in view their pre-employment needs and career requirements, this course on Communication Skills Laboratory will prepare students to adapt themselves with ease to the industry environment, thus rendering them as prospective assets to industries. The course will equip the students with the necessary communication skills that would go a long way in helping them in their profession.

OBJECTIVES:

- To equip students of engineering and technology with effective speaking and listening skills in English.
- To help them develop their soft skills and interpersonal skills, which will make the transition from college to workplace smoother and help them excel in their job.
- To enhance the performance of students at Placement Interviews, Group Discussions and other recruitment exercises.

I. PC based session	(Weightage 40%)	24 periods
----------------------------	------------------------	-------------------

A. English Language Lab (18 Periods)

1. Listening Comprehension: (6)

Listening and typing – Listening and sequencing of sentences – Filling in the blanks - Listening and answering questions.

2. Reading Comprehension: (6)

Filling in the blanks - Close exercises – Vocabulary building - Reading and answering questions.

3. Speaking: (6)

Phonetics: Intonation – Ear training - Correct Pronunciation – Sound recognition exercises – Common Errors in English.

Conversations: Face to Face Conversation – Telephone conversation – Role play activities (Students take on roles and engage in conversation)

B. Discussion of audio-visual materials (6 periods)
(Samples are available to learn and practice)

1. Resume / Report Preparation / Letter Writing (1)

Structuring the resume / report - Letter writing / Email Communication - Samples.

- 2. Presentation skills:** (1)
Elements of effective presentation – Structure of presentation - Presentation tools – Voice Modulation – Audience analysis - Body language – Video samples
- 3. Soft Skills:** (2)
Time management – Articulateness – Assertiveness – Psychometrics – Innovation and Creativity - Stress Management & Poise - Video Samples
- 4. Group Discussion:** (1)
Why is GD part of selection process ? - Structure of GD – Moderator – led and other GDs - Strategies in GD – Team work - Body Language - Mock GD -Video samples
- 5. Interview Skills:** (1)
Kinds of interviews – Required Key Skills – Corporate culture – Mock interviews- Video samples.

II. Practice Session	(Weightage – 60%)	24 periods
-----------------------------	--------------------------	-------------------

- 1. Resume / Report Preparation / Letter writing:** Students prepare their own resume and report. (2)
- 2. Presentation Skills:** Students make presentations on given topics. (8)
- 3. Group Discussion:** Students participate in group discussions. (6)
- 4. Interview Skills:** Students participate in Mock Interviews (8)

REFERENCES:

1. Anderson, P.V, **Technical Communication**, Thomson Wadsworth , Sixth Edition, New Delhi, 2007.
2. Prakash, P, **Verbal and Non-Verbal Reasoning**, Macmillan India Ltd., Second Edition, New Delhi, 2004.
3. John Seely, **The Oxford Guide to Writing and Speaking**, Oxford University Press, New Delhi, 2004.
4. Evans, D, **Decisionmaker**, Cambridge University Press, 1997.
5. Thorpe, E, and Thorpe, S, **Objective English**, Pearson Education, Second Edition, New Delhi, 2007.
6. Turton, N.D and Heaton, J.B, **Dictionary of Common Errors**, Addison Wesley Longman Ltd., Indian reprint 1998.

LAB REQUIREMENTS:

1. Teacher console and systems for students.
2. English Language Lab Software
3. Career Lab Software

GE2321

COMMUNICATION SKILLS LABORATORY

Guidelines for the course

1. A batch of 60 / 120 students is divided into two groups – one group for the PC- based session and the other group for the Class room session.
2. The English Lab (2 Periods) will be handled by a faculty member of the **English Department**. The Career Lab (2 Periods) may be handled by any competent teacher, **not necessarily from English Department**
3. **Record Notebook:** At the end of each session of English Lab, review exercises are given for the students to answer and the computer evaluated sheets are to be compiled as record notebook. Similar exercises for the career lab are to be compiled in the record notebook.
4. **Internal Assessment:** The 15 marks (the other 5 marks for attendance) allotted for the internal assessment will be based on the record notebook compiled by the candidate. 10 marks may be allotted for English Lab component and 5 marks for the Career Lab component.
5. **End semester Examination:** The end-semester examination carries 40% weightage for English Lab and 60% weightage for Career Lab.

Each candidate will have separate sets of questions assigned by the teacher using the teacher-console enabling PC-based evaluation for the 40% of marks allotted.

The Career Lab component will be evaluated for a maximum of 60% by a local examiner & an external examiner drafted from other Institutions, similar to any other lab examination conducted by Anna University.

CS2358

INTERNET PROGRAMMING LAB

L T P C

1 0 3 2

LIST OF EXPERIMENTS

1. Create a web page with the following using HTML
 - i) To embed an image map in a web page
 - ii) To fix the hot spots
 - iii) Show all the related information when the hot spots are clicked.
2. Create a web page with all types of Cascading style sheets.
3. Client Side Scripts for Validating Web Form Controls using DHTML
4. Write programs in Java to create applets incorporating the following features:
5. Create a color palette with matrix of buttons
Set background and foreground of the control text area by selecting a color from color palette.
In order to select Foreground or background use check box control as radio buttons
To set background images

6. Write programs in Java using Servlets:
To invoke servlets from HTML forms
To invoke servlets from Applets
7. Write programs in Java to create three-tier applications using JSP and Databases
 - for conducting on-line examination.
 - for displaying student mark list. Assume that student information is available in a database which has been stored in a database server.
8. Programs using XML – Schema – XSLT/XSL
9. Programs using AJAX
10. Consider a case where we have two web Services- an airline service and a travel agent and the travel agent is searching for an airline. Implement this scenario using Web Services and Data base.

LIST OF ELECTIVES
SEMESTER VI – Elective I

Code No.	Course Title	L	T	P	C
CS2021	<u>Multicore Programming</u>	3	0	0	3
CS2022	<u>Visual Programming</u>	3	0	0	3
CS2023	<u>Advanced JAVA Programming</u>	3	0	0	3
CS2024	<u>Parallel Programming</u>	3	0	0	3
IT2353	<u>Web Technology</u>	3	0	0	3

**SEMESTER VI – Elective
II**

Code No.	Course Title	L	T	P	C
CS2028	<u>UNIX Internals</u>	3	0	0	3
MA2264	<u>Numerical Methods</u>	3	1	0	4
IT2354	<u>Embedded Systems</u>	3	0	0	3
CS2029	<u>Advanced Database Technology</u>	3	0	0	3
IT2043	<u>Knowledge Management</u>	3	0	0	3
CS2030	<u>High Performance Microprocessors</u>	3	0	0	3

CS2406

OPEN SOURCE LAB

**L T P C
0 0 3 2**

OBJECTIVE:

To expose students to FOSS environment and introduce them to use open source packages

1. **Kernel configuration, compilation and installation** : Download / access the latest kernel source code from kernel.org, compile the kernel and install it in the local system. Try to view the source code of the kernel
2. **Virtualisation environment** (e.g., xen, qemu or lguest) to test an applications, new kernels and isolate applications. It could also be used to expose students to other alternate OSs like *BSD
3. **Compiling from source** : learn about the various build systems used like the auto* family, cmake, ant etc. instead of just running the commands. This could involve the full process like fetching from a cvs and also include autoconf, automake etc.,
4. **Introduction to packet management system** : Given a set of RPM or DEB, how to build and maintain, serve packages over http or ftp. and also how do you configure client systems to access the package repository.
5. **Installing various software packages**
 - Either the package is yet to be installed or an older version is existing. The student can practice installing the latest version. Of course, this might need internet access.
 - Install samba and share files to windows
 - Install Common Unix Printing System(CUPS)
6. **Write userspace drivers using fuse** -- easier to debug and less dangerous to the system (Writing full-fledged drivers is difficult at student level)
7. **GUI programming : a sample programme** – using Gambas since the students have VB knowledge. However, one should try using GTK or QT

8. **Version Control System setup and usage** using RCS, CVS, SVN
9. **Text processing with Perl:** simple programs, connecting with database e.g., MYSQL
10. **Running PHP :** simple applications like login forms after setting up a LAMP stack
11. **Running Python :** some simple exercise – e.g. Connecting with MySql database
12. **Set up the complete network interface** using ifconfig command like setting gateway, DNS, IP tables, etc.,

RESOURCES :

An environment like **FOSS Lab Server** (developed by NRCFOSS containing the various packages)

OR

Equivalent system with Linux distro supplemented with relevant packages

Note:

Once the list of experiments are finalised, NRCFOSS can generate full lab manuals complete with exercises, necessary downloads, etc. These could be made available on NRCFOSS web portal.

CS2028

UNIX INTERNALS

L T P C

3 0 0 3

UNIT I

9

General Review of the System-History-System structure-User Perspective-Operating System Services- Assumptions About Hardware. Introduction to the Kernel-Architecture System Concepts-Data Structures- System Administration.

UNIT II

9

The Buffer Cache-Headers-Buffer Pool-Buffer Retrieval-Reading and Writing Disk Blocks - Advantages and Disadvantages. Internal Representation of Files-Inodes-Structure-Directories-Path Name to Inode- Super Block-Inode Assignment-Allocation of Disk Blocks -Other File Types.

UNIT III

9

System Calls for the File System-Open-Read-Write-Lseek-Close-Create-Special files Creation -Change Directory and Change Root-Change Owner and Change Mode-Stat-Fstat-Pipes-Dup-Mount-Unmount-Link-Unlink-File System Abstraction-Maintenance.

UNIT IV

9

The System Representation of Processes-States-Transitions-System Memory-Context of a Process-Saving the Context-Manipulation of a Process Address Space-Sleep Process Control-signals-Process Termination-Awaiting-Invoking other Programs-The Shell-System Boot and the INIT Process.

UNIT V

9

Memory Management Policies-Swapping-Demand Paging-a Hybrid System-I/O Subsystem-Driver Interfaces-Disk Drivers-Terminal Drivers.

TEXT BOOK:

1. Maurice J. Bach, "The Design of the Unix Operating System", Pearson Education, 2002.

REFERENCES:

1. Uresh Vahalia, "UNIX Internals: The New Frontiers", Prentice Hall, 2000.
2. John Lion, "Lion's Commentary on UNIX", 6th edition, Peer-to-Peer Communications, 2004.
3. Daniel P. Bovet & Marco Cesati, "Understanding the Linux Kernel", O'REILLY, Shroff Publishers & Distributors Pvt. Ltd, 2000.
4. M. Beck et al, "Linux Kernel Programming", Pearson Education Asia, 2002

MA2264**NUMERICAL METHODS****L T P C
3 1 0 4****AIM:**

With the present development of the computer technology, it is necessary to develop efficient algorithms for solving problems in science, engineering and technology. This course gives a complete procedure for solving different kinds of problems occur in engineering numerically.

OBJECTIVES:

At the end of the course, the students would be acquainted with the basic concepts in numerical methods and their uses are summarized as follows:

- i. The roots of nonlinear (algebraic or transcendental) equations, solutions of large system of linear equations and eigen value problem of a matrix can be obtained numerically where analytical methods fail to give solution
- ii. When huge amounts of experimental data are involved, the methods discussed on interpolation will be useful in constructing approximate polynomial to represent the data and to find the intermediate values.
- iii. The numerical differentiation and integration find application when the function in the analytical form is too complicated or the huge amounts of data are given such as series of measurements, observations or some other empirical information.
- iv. Since many physical laws are couched in terms of rate of change of one/two or more independent variables, most of the engineering problems are characterized in the form of either nonlinear ordinary differential equations or partial differential equations. The methods introduced in the solution of ordinary differential equations and partial differential equations will be useful in attempting any engineering problem.

UNIT I SOLUTION OF EQUATIONS AND EIGENVALUE PROBLEMS 9

Solution of equation –Fixed point iteration: $x=g(x)$ method - Newton's method – Solution of linear system by Gaussian elimination and Gauss-Jordon method– Iterative method - Gauss-Seidel method - Inverse of a matrix by Gauss Jordan method – Eigen value of a matrix by power method and by Jacobi method for symmetric matrix.

UNIT II INTERPOLATION AND APPROXIMATION 9

Lagrangian Polynomials – Divided differences – Interpolating with a cubic spline – Newton’s forward and backward difference formulas.

UNIT III NUMERICAL DIFFERENTIATION AND INTEGRATION 9

Differentiation using interpolation formulae – Numerical integration by trapezoidal and Simpson’s 1/3 and 3/8 rules – Romberg’s method – Two and Three point Gaussian quadrature formulae – Double integrals using trapezoidal and Simpson’s rules.

UNIT IV INITIAL VALUE PROBLEMS FOR ORDINARY DIFFERENTIAL EQUATIONS 9

Single step methods: Taylor series method – Euler method for first order equation – Fourth order Runge – Kutta method for solving first and second order equations – Multistep methods: Milne’s and Adam’s predictor and corrector methods.

UNIT V BOUNDARY VALUE PROBLEMS IN ORDINARY AND PARTIAL DIFFERENTIAL EQUATIONS 9

Finite difference solution of second order ordinary differential equation – Finite difference solution of one dimensional heat equation by explicit and implicit methods – One dimensional wave equation and two dimensional Laplace and Poisson equations.

L = 45 , TOTAL: 45 PERIODS

TEXT BOOKS:

1. Veerarjan, T and Ramachandran, T. ‘Numerical methods with programming in ‘C’ Second Edition, Tata McGraw-Hill Publishing.Co.Ltd. (2007).
2. Sankara Rao K, ‘Numerical Methods for Scientists and Engineers’ – 3rd edition Printice Hall of India Private Ltd, New Delhi, (2007).

REFERENCES:

1. Chapra, S. C and Canale, R. P. “Numerical Methods for Engineers”, 5th Edition, Tata McGraw-Hill, New Delhi, 2007.
2. Gerald, C. F. and Wheatley, P.O., “Applied Numerical Analysis”, 6th Edition, Pearson Education Asia, New Delhi, 2006.
3. Grewal, B.S. and Grewal, J.S., “ Numerical methods in Engineering and Science”, 6th Edition, Khanna Publishers, New Delhi, 2004

CS2021

MULTICORE PROGRAMMING

**L T P C
3 0 0 3**

UNIT I INTRODUCTION TO MULTIPROCESSORS AND SCALABILITY ISSUES 9

Scalable design principles – Principles of processor design – Instruction Level Parallelism, Thread level parallelism. Parallel computer models -- Symmetric and distributed shared memory architectures – Performance Issues – Multi-core Architectures - Software and hardware multithreading – SMT and CMP architectures – Design issues – Case studies – Intel Multi-core architecture – SUN CMP architecture.

UNIT II PARALLEL PROGRAMMING 9
Fundamental concepts – Designing for threads – scheduling - Threading and parallel programming constructs – Synchronization – Critical sections – Deadlock. Threading APIs.

UNIT III OPENMP PROGRAMMING 9
OpenMP – Threading a loop – Thread overheads – Performance issues – Library functions. Solutions to parallel programming problems – Data races, deadlocks and livelocks – Non-blocking algorithms – Memory and cache related issues.

UNIT IV MPI PROGRAMMING 9
MPI Model – collective communication – data decomposition – communicators and topologies – point-to-point communication – MPI Library.

UNIT V MULTITHREADED APPLICATION DEVELOPMENT 9
Algorithms, program development and performance tuning.

TOTAL: 45 PERIODS

TEXT BOOKS:

1. Shameem Akhter and Jason Roberts, “Multi-core Programming”, Intel Press, 2006.
2. Michael J Quinn, Parallel programming in C with MPI and OpenMP, Tata Macgraw Hill, 2003.

REFERENCES:

1. John L. Hennessey and David A. Patterson, “ Computer architecture – A quantitative approach”, Morgan Kaufmann/Elsevier Publishers, 4th. edition, 2007.
2. David E. Culler, Jaswinder Pal Singh, “Parallel computing architecture : A hardware/ software approach” , Morgan Kaufmann/Elsevier Publishers, 1999.

CS2022 VISUAL PROGRAMMING L T P C
3 0 0 3

UNIT I 9
Windows Programming Fundamentals – MFC – Windows – Graphics – Menus – Mouse and keyboard – Bitmaps – Palettes – Device-Independent Bitmaps

UNIT II 9
Controls – Modal and Modeless Dialog – Property – Data I/O – Sound – Timer

UNIT III 9
Memory management – SDI – MDI – MFC for Advanced windows user Interface – status bar and Toolbars – Tree view – List view – Threads

UNIT IV 9
ODBC – MFC Database classes – DAO - DLLs – Working with Images

UNIT V **9**
COM Fundamentals – ActiveX control – ATL – Internet Programming

TOTAL: 45 PERIODS

TEXT BOOK:

1. Richard C. Leinecker and Tom Archer, "Visual C++ 6 Programming Bible", Wiley DreamTech Press, 2006.

REFERENCES:

1. Lars Klander, "Core Visual C++ 6", Pearson Education, 2000
2. Deital, DEital, Liperi and Yaeger "Visual V++ .NET How to Program" , Pearson Education, 2004.

IT2354 **EMBEDDED SYSTEMS** **L T P C**
3 0 0 3

UNIT I **EMBEDDED COMPUTING** **9**

Challenges of Embedded Systems – Embedded system design process. Embedded processors – 8051 Microcontroller, ARM processor – Architecture, Instruction sets and programming.

UNIT II **MEMORY AND INPUT / OUTPUT MANAGEMENT** **9**

Programming Input and Output – Memory system mechanisms – Memory and I/O devices and interfacing – Interrupts handling.

UNIT III **PROCESSES AND OPERATING SYSTEMS** **9**

Multiple tasks and processes – Context switching – Scheduling policies – Interprocess communication mechanisms – Performance issues.

UNIT IV **EMBEDDED SOFTWARE** **9**

Programming embedded systems in assembly and C – Meeting real time constraints – Multi-state systems and function sequences. Embedded software development tools – Emulators and debuggers.

UNIT V **EMBEDDED SYSTEM DEVELOPMENT** **9**

Design issues and techniques – Case studies – Complete design of example embedded systems.

TOTAL: 45 PERIODS

TEXT BOOKS:

1. Wayne Wolf, "Computers as Components: Principles of Embedded Computer System Design", Elsevier, 2006.
2. Michael J. Pont, "Embedded C", Pearson Education , 2007.

REFERENCES:

1. Steve Heath, "Embedded System Design", Elsevier, 2005.
2. Muhammed Ali Mazidi, Janice Gillispie Mazidi and Rolin D. McKinlay, "The 8051 Microcontroller and Embedded Systems", Pearson Education, Second edition, 2007.

CS2029

ADVANCED DATABASE TECHNOLOGY

L T P C
3 0 0 3

UNIT I RELATIONAL MODEL ISSUES 9

ER Model - Normalization – Query Processing – Query Optimization - Transaction Processing - Concurrency Control – Recovery - Database Tuning.

UNIT II DISTRIBUTED DATABASES 9

Parallel Databases – Inter and Intra Query Parallelism – Distributed Database Features – Distributed Database Architecture – Fragmentation – Distributed Query Processing – Distributed Transactions Processing – Concurrency Control – Recovery – Commit Protocols.

UNIT III OBJECT ORIENTED DATABASES 9

Introduction to Object Oriented Data Bases - Approaches - Modeling and Design - Persistence – Query Languages - Transaction - Concurrency – Multi Version Locks – Recovery – POSTGRES – JASMINE –GEMSTONE - ODMG Model.

UNIT IV EMERGING SYSTEMS 9

Enhanced Data Models - Client/Server Model - Data Warehousing and Data Mining - Web Databases – Mobile Databases- XML and Web Databases.

UNIT V CURRENT ISSUES 9

Rules - Knowledge Bases - Active and Deductive Databases - Multimedia Databases– Multimedia Data Structures – Multimedia Query languages - Spatial Databases.

TOTAL: 45 PERIODS

TEXT BOOKS:

1. Thomas Connolly and Carlolyn Begg, “Database Systems, A Practical Approach to Design, Implementation and Management”, Third Edition, Pearson Education 2003.

REFERENCES:

1. R. Elmasri, S.B. Navathe, “Fundamentals of Database Systems”, Fifth Edition, Pearson Education, 2006.
2. Abraham Silberschatz, Henry F. Korth, S. Sudharshan, “Database System Concepts”, Fifth Edition, Tata McGraw Hill, 2006.
3. C.J.Date, A.Kannan, S.Swamynathan, “An Introduction to Database Systems”, Eighth Edition, Pearson Education, 2006.

IT2043

KNOWLEDGE MANAGEMENT

L T P C
3 0 0 3

UNIT I KNOWLEDGE MANAGEMENT 9

KM Myths – KM Life Cycle – Understanding Knowledge – Knowledge, intelligence – Experience – Common Sense – Cognition and KM – Types of Knowledge – Expert Knowledge – Human Thinking and Learning.

UNIT II KNOWLEDGE MANAGEMENT SYSTEM LIFE CYCLE 9
Challenges in Building KM Systems – Conventional Vrs KM System Life Cycle (KMSLS)
– Knowledge Creation and Knowledge Architecture – Nonaka’s Model of Knowledge
Creation and Transformation. Knowledge Architecture.

UNIT III CAPTURING KNOWLEDGE 9
Evaluating the Expert – Developing a Relationship with Experts – Fuzzy Reasoning and
the Quality of Knowledge – Knowledge Capturing Techniques, Brain Storming – Protocol
Analysis – Consensus Decision Making – Repertory Grid- Concept Mapping –
Blackboarding.

UNIT IV KNOWLEDGE CODIFICATION 9
Modes of Knowledge Conversion – Codification Tools and Procedures – Knowledge
Developer’s Skill Sets – System Testing and Deployment – Knowledge Testing –
Approaches to Logical Testing, User Acceptance Testing – KM System Deployment
Issues – User Training – Post implementation.

UNIT V KNOWLEDGE TRANSFER AND SHARING 9
Transfer Methods – Role of the Internet – Knowledge Transfer in e-world – KM System
Tools – Neural Network – Association Rules – Classification Trees – Data Mining and
Business Intelligence – Decision Making Architecture – Data Management – Knowledge
Management Protocols – Managing Knowledge Workers.

TOTAL: 45 PERIODS

TEXT BOOK:

1. Elias. M. Award & Hassan M. Ghaziri – “Knowledge Management” Pearson
Education 2003.

REFERENCES:

1. Guus Schreiber, Hans Akkermans, Anjo Anjewierden, Robert de Hoog, Nigel
Shadbolt, Walter Van de Velde and Bob Wielinga, “Knowledge Engineering and
Management”, Universities Press, 2001.
2. C.W. Holsapple, “Handbooks on Knowledge Management”, International Handbooks
on Information Systems, Vol 1 and 2, 2003

CS2030 HIGH PERFORMANCE MICROPROCESSORS L T P C
3 0 0 3

AIM

To do a detailed study of CISC and RISC principles, study the architecture & special
features of the Pentium processors and typical RISC processors and to study the
architecture of special purpose processors.

OBJECTIVES

- To study the principles of CISC
- To study the Pentium processor family
- To study the principles of RISC
- To study the architecture & special features of typical RISC processors.
- To study the architecture & function of special purpose processors.

UNIT I CISC PRINCIPLES 9

Classic CISC microprocessors, Intel x86 Family: Architecture - register set - Data formats - Addressing modes - Instruction set - Assembler directives – Interrupts - Segmentation, Paging, Real and Virtual mode execution – Protection mechanism, Task management 80186, 286, 386 and 486 architectures.

UNIT II PENTIUM PROCESSORS 10

Introduction to Pentium microprocessor – Special Pentium Registers – Pentium Memory Management – New Pentium instructions – Introduction to Pentium Pro and its special features – Architecture of Pentium-II, Pentium-III and Pentium4 microprocessors.

UNIT III RISC PRINCIPLES 10

RISC Vs CISC – RISC properties and evaluation – On chip register File Vs Cache evaluation – Study of a typical RISC processor – The PowerPC – Architecture & special features – Power PC 601 – IBM RS/6000, Sun SPARC Family – Architecture – Super SPARC.

UNIT IV RISC PROCESSOR 8

MIPS Rx000 family – Architecture – Special features – MIPS R4000 and R4400 – Motorola 88000 Family – Architecture – MC 88110 – MC 88100 and MC 88200.

UNIT V SPECIAL PURPOSE PROCESSORS 8

EPIC Architecture – ASIPs – Network Processors – DSPs – Graphics / Image Processors.

TOTAL : 45 PERIODS

TEXT BOOK:

1. Daniel Tabak, “Advanced Microprocessors”, Tata McGraw-Hill, 1995, 2nd Edition.

REFERENCES

1. www.intel.com/products/server/processors/server/itanium2 (Unit V:EPIC)
2. www.hpl.hp.com/techreports/1999/HPL-1999-111.html (UnitV: Network Processor)
3. www.intel.com/design/network/products/npfamily (Unit V: Network Processor)
4. www.national.com/appinfo/imaging/processors.html (Unit V: Image Processor)
5. Barry B.Brey, “The Intel Microprocessors, 8086/8088, 80186/80188, 80286, 80386, 80486, Pentium, PentiumPro Processor, PentiumII, PentiumIII, PentiumIV, Architecture, Programming & Interfacing”, 6th Edition, Pearson Education/PHI, 2002.

