ANNA UNIVERSITY, CHENNAI

AFFILIATED INSTITUTIONS

R-2008

B.TECH. INFORMATION TECHNOLOGY

SEMESTER V

Code No.	Course Title	L	Т	Ρ	С		
THEORY							
IT2301	Java Programming	3	0	0	3		
MG2452	Engineering Economics & Financial Accounting	3	0	0	3		
CS2304	System Software	3	1	0	4		
CS2302	Computer Networks	3	0	0	3		
CS2403	Digital Signal Processing	3	0	0	3		
IT2302	Information Theory and Coding	3	0	0	3		
PRACTICAL							
CS2308	System Software Lab	0	0	3	2		
IT2305	Java Programming Lab	0	0	3	2		
GE2321	Communication Skills Lab	0	0	4	2		
	TOTAL	18	1	10	25		

IT2301

JAVA PROGRAMMING

LTPC 3 0 0 3

AIM

To understand the concepts of object-oriented, event driven, and concurrent programming paradigms and develop skills in using these paradigms using Java.

UNIT I

Object oriented programming concepts – objects – classes – methods and messages – abstraction and encapsulation – inheritance – abstract classes – polymorphism.- Objects and classes in Java – defining classes – methods - access specifiers – static members – constructors – finalize method

10

9

Arrays – Strings - Packages – Java-Doc comments – Inheritance – class hierarchy – polymorphism – dynamic binding – final keyword – abstract classes

UNIT III

The Object class – Reflection – interfaces – object cloning – inner classes – proxies - I/O Streams - Graphics programming – Frame – Components – working with 2D shapes.

UNIT IV

Basics of event handling – event handlers – adapter classes – actions – mouse events – AWT event hierarchy – introduction to Swing – Model-View-Controller design pattern – buttons – layout management – Swing Components – exception handling – exception hierarchy – throwing and catching exceptions.

8

UNIT V

Motivation for generic programming – generic classes – generic methods – generic code and virtual machine - inheritance and generics - reflection and generics - Multi-threaded programming - interrupting threads - thread states - thread properties - thread synchronization – Executors – synchronizers.

TOTAL:45 PERIODS

TEXT BOOK

1. Cay S. Horstmann and Gary Cornell, "Core Java: Volume I – Fundamentals", Eighth Edition, Sun Microsystems Press, 2008.

REFERENCES

- 1. K. Arnold and J. Gosling, "The JAVA programming language", Third edition, Pearson Education. 2000.
- 2. Timothy Budd, "Understanding Object-oriented programming with Java", Updated Edition. Pearson Education. 2000.
- 3. C. Thomas Wu, "An introduction to Object-oriented programming with Java", Fourth Edition, Tata McGraw-Hill Publishing company Ltd., 2006.

CS2304

SYSTEM SOFTWAR

LTPC 3 1 0 4

AIM

To have an understanding of foundations of design of assemblers, loaders, linkers, and macro processors.

OBJECTIVES

- To understand the relationship between system software and machine • architecture.
- To know the design and implementation of assemblers .
- To know the design and implementation of linkers and loaders. •
- To have an understanding of macro processors.
- To have an understanding of system software tools.

INTRODUCTION UNIT I

System software and machine architecture - The Simplified Instructional Computer (SIC) - Machine architecture - Data and instruction formats - addressing modes instruction sets - I/O and programming.

UNIT II ASSEMBLERS

Basic assembler functions - A simple SIC assembler – Assembler algorithm and data structures - Machine dependent assembler features - Instruction formats and addressing modes - Program relocation - Machine independent assembler features - Literals -Symbol-defining statements - Expressions - One pass assemblers and Multi pass assemblers - Implementation example - MASM assembler.

8

8

UNIT III LOADERS AND LINKERS

Basic loader functions - Design of an Absolute Loader - A Simple Bootstrap Loader -Machine dependent loader features - Relocation - Program Linking - Algorithm and Data Structures for Linking Loader - Machine-independent loader features - Automatic Library Search - Loader Options - Loader design options - Linkage Editors - Dynamic Linking – Bootstrap Loaders - Implementation example - MSDOS linker.

UNIT IV MACRO PROCESSORS

Basic macro processor functions - Macro Definition and Expansion – Macro Processor Algorithm and data structures - Machine-independent macro processor features Concatenation of Macro Parameters – Generation of Unique Labels – Conditional Macro Expansion – Keyword Macro Parameters-Macro within Macro-Implementation example MASM Macro Processor – ANSI C Macro language.

UNIT V SYSTEM SOFTWARE TOOLS

Text editors - Overview of the Editing Process - User Interface - Editor Structure. -Interactive debugging systems - Debugging functions and capabilities - Relationship with other parts of the system - User-Interface Criteria.

L: 45, T: 15, TOTAL: 60 PERIODS

TEXT BOOK

1. Leland L. Beck, "System Software - An Introduction to Systems Programming", 3rd Edition, Pearson Education Asia, 2000.

REFERENCES

- 1. D. M. Dhamdhere, "Systems Programming and Operating Systems", Second Revised Edition, Tata McGraw-Hill, 1999.
- 2. John J. Donovan "Systems Programming", Tata McGraw-Hill Edition, 1972.
- 3. John R. Levine, Linkers & Loaders Harcourt India Pvt. Ltd., Morgan Kaufmann Publishers, 2000.

CS2302

COMPUTER NETWORKS

LTPC 3 0 0 3

9

9

UNIT I

Network architecture - layers - Physical links - Channel access on links - Hybrid multiple access techniques - Issues in the data link layer - Framing – Error correction and detection – Link-level Flow Control

UNIT II

Medium access – CSMA – Ethernet – Token ring – FDDI - Wireless LAN – Bridges and Switches

UNIT III

9

Circuit switching vs. packet switching / Packet switched networks - IP - ARP - RARP -DHCP - ICMP - Queueing discipline - Routing algorithms - RIP - OSPF - Subnetting CIDR – Interdomain routing – BGP – Ipv6 – Multicasting – Congestion avoidance in network layer

9

UNIT IV

UDP – TCP – Adaptive Flow Control – Adaptive Retransmission - Congestion control – Congestion avoidance – QoS

UNIT V

Email (SMTP, MIME, IMAP, POP3) – HTTP – DNS- SNMP – Telnet – FTP – Security – PGP - SSH

TOTAL: 45 PERIODS

TEXT BOOKS

1. Larry L. Peterson, Bruce S. Davie, "Computer Networks: A Systems Approach" Third Edition, Morgan Kauffmann Publishers Inc., 200UNIT III

REFERENCES

- 1. James F. Kuross, Keith W. Ross, "Computer Networking, A Top-Down Approach Featuring the Internet", Third Edition, Addison Wesley, 2004.
- Nader F. Mir, "Computer and Communication Networks", Pearson Education, 2007
- 3. Comer, "Computer Networks and Internets with Internet Applications", Fourth Edition, Pearson Education, 200UNIT III
- 4. Andrew S. Tanenbaum, "Computer Networks", Fourth Edition, 200UNIT III
- 5. William Stallings, "Data and Computer Communication", Sixth Edition, Pearson Education, 2000

CS2403

DIGITAL SIGNAL PROCESSING

UNIT I SIGNALS AND SYSTEMS Basic elements of DSP – concepts of frequency in Analog and Digital Signals – sampling

theorem - Discrete - time signals, systems - Analysis of discrete time LTI systems - Z transform – Convolution (linear and circular) – Correlation.

FREQUENCY TRANSFORMATIONS UNIT II

Introduction to DFT - Properties of DFT - Filtering methods based on DFT - FFT Algorithms Decimation - in - time Algorithms, Decimation - in - frequency Algorithms -Use of FFT in Linear Filtering – DCT.

IIR FILTER DESIGN

Structures of IIR – Analog filter design – Discrete time IIR filter from analog filter – IIR filter design by Impulse Invariance, Bilinear transformation, Approximation of derivatives - (HPF, BPF, BRF) filter design using frequency translation

UNIT IV FIR FILTER DESIGN

Structures of FIR – Linear phase FIR filter – Filter design using windowing techniques, Frequency sampling techniques - Finite word length effects in digital Filters

LTPC 3003

9

9

9

9

UNIT V APPLICATIONS

Multirate signal processing – Speech compression – Adaptive filter – Musical sound processing – Image enhancement.

TEXT BOOKS

TOTAL:45 PERIODS

9

- 1. John G. Proakis & Dimitris G.Manolakis, "Digital Signal Processing Principles, Algorithms & Applications", Fourth edition, Pearson education / Prentice Hall, 2007.
- 2. Emmanuel C. Ifeachor, & Barrie.W. Jervis, "Digital Signal Processing", Second edition, Pearson Education / Prentice Hall, 2002.

REFERENCES

- 1. Alan V.Oppenheim, Ronald W. Schafer & Hohn. R.Back, "Discrete Time Signal Processing", Pearson Education.
- 2. Andreas Antoniou, "Digital Signal Processing", Tata McGraw Hill.

IT2302 INFORMATION THEORY AND CODING

UNIT I INFORMATION THEORY

Information – Entropy, Information rate, classification of codes, Kraft McMillan inequality, Source coding theorem, Shannon-Fano coding, Huffman coding, Extended Huffman coding - Joint and conditional entropies, Mutual information - Discrete memoryless channels – BSC, BEC – Channel capacity, Shannon limit.

UNIT II SOURCE CODING: TEXT, AUDIO AND SPEECH

Text: Adaptive Huffman Coding, Arithmetic Coding, LZW algorithm – Audio: Perceptual coding, Masking techniques, Psychoacoustic model, MEG Audio layers I,II,III, Dolby AC3 - Speech: Channel Vocoder, Linear Predictive Coding

UNIT III SOURCE CODING: IMAGE AND VIDEO

Image and Video Formats – GIF, TIFF, SIF, CIF, QCIF – Image compression: READ, JPEG – Video Compression: Principles-I,B,P frames, Motion estimation, Motion compensation, H.261, MPEG standard

UNIT IV ERROR CONTROL CODING: BLOCK CODES

Definitions and Principles: Hamming weight, Hamming distance, Minimum distance decoding - Single parity codes, Hamming codes, Repetition codes - Linear block codes, Cyclic codes - Syndrome calculation, Encoder and decoder - CRC

UNIT V ERROR CONTROL CODING: CONVOLUTIONAL CODES

Convolutional codes – code tree, trellis, state diagram - Encoding – Decoding: Sequential search and Viterbi algorithm – Principle of Turbo coding

TOTAL= 45 PERIODS

TEXT BOOKS

- 1. R Bose, "Information Theory, Coding and Crptography", TMH 2007
- 2. Fred Halsall, "Multidedia Communications: Applications, Networks, Protocols and Standards", Perason Education Asia, 2002

L T P C 3 0 0 3

9

9

9

9

REFERENCES

- 1. K Sayood, "Introduction to Data Compression" 3/e, Elsevier 2006
- 2. S Gravano, "Introduction to Error Control Codes", Oxford University Press 2007
- 3. Amitabha Bhattacharya, "Digital Communication", TMH 2006

CS2308

SYSTEM SOFTWARE LAB

L T P C 0 0 3 2

(Using C)

- 1. Implement a symbol table with functions to create, insert, modify, search, and display.
- 2. Implement pass one of a two pass assembler.
- 3. Implement pass two of a two pass assembler.
- 4. Implement a single pass assembler.
- 5. Implement a two pass macro processor
- 6. Implement a single pass macro processor.
- 7. Implement an absolute loader.
- 8. Implement a relocating loader.
- 9. Implement pass one of a direct-linking loader.
- 10. Implement pass two of a direct-linking loader.
- 11. Implement a simple text editor with features like insertion / deletion of a character, word, and sentence.
- 12. Implement a symbol table with suitable hashing

(For loader exercises, output the snap shot of the main memory as it would be, after the loading has taken place)

TOTAL: 45 PERIODS

S.No.	Description of Equipment	Quantity Required	Quantity available	Deficiency %
1.	Hardware – Pentium PC Desktops	30 Nos.		
2.	Software – Turbo C (Freely download)	30 user License		

Requirement for a batch of 30 students

IT2305

- 1. Develop a Java package with simple Stack and Queue classes. Use JavaDoc comments for documentation.
- 2. Design a class for Complex numbers in Java. In addition to methods for basic operations on complex numbers, provide a method to return the number of active objects created.
- 3. Design a Date class similar to the one provided in the java.util package.
- Develop with suitable hierarchy, classes for Point, Shape, Rectangle, Square, Circle, Ellipse, Triangle, Polygon, etc. Design a simple test application to demonstrate dynamic polymorphism.
- 5. Design a Java interface for ADT Stack. Develop two different classes that implement this interface, one using array and the other using linked-list. Provide necessary exception handling in both the implementations.
- 6. Write a Java program to read a file that contains DNA sequences of arbitrary length one per line (note that each DNA sequence is just a String). Your program should sort the sequences in descending order with respect to the number of 'TATA' subsequences present. Finally write the sequences in sorted order into another file.
- 7. Develop a simple paint-like program that can draw basic graphical primitives in different dimensions and colors. Use appropriate menu and buttons.
- 8. Develop a scientific calculator using even-driven programming paradigm of Java.
- 9. Develop a template for linked-list class along with its methods in Java.
- 10. Design a thread-safe implementation of Queue class. Write a multi-threaded producer-consumer application that uses this Queue class.
- 11. Write a multi-threaded Java program to print all numbers below 100,000 that are both prime and fibonacci number (some examples are 2, 3, 5, 13, etc.). Design a thread that generates prime numbers below 100,000 and writes them into a pipe. Design another thread that generates fibonacci numbers and writes them to another pipe. The main thread should read both the pipes to identify numbers common to both.
- 12. Develop a multi-threaded GUI application of your choice.

TOTAL:45 PERIODS

S. No.	Description of Equipment	Quantity Required	Quantity available	Deficiency %
3.	Hardware: Pentium IV with 2 GB RAM, 160 GB HARD Disk, Monitor 1024 x 768 colour 60 Hz.	30 Nodes		
4.	Software: Windows /Linux operating system JDK 1.6(or above)	30 user license		

Requirement for a batch of 30 students

www.EEEEexclusive.boospot.com