Syllabus

(With effect from 2013-2014 Academic year)

ಪಠ್ಯಕ್ರಮ

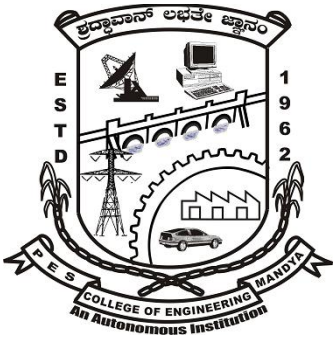
(ಶೈಕ್ಷಣಿಕವರ್ಷ 2013–14)

V & VI Semester

Bachelor Degree
in
Computer Science and Engineering

Out Come Based Education

ಫಲತಾಂಶ ಆಧಾರಿತ ಶಿಕ್ಷಣ



P.E.S. College of Engineering, Mandya - 571 401, Karnataka

(An Autonomous Institution Affiliated to VTU, Belagavi)

Grant -in- Aid Institution

(Government of Karnataka)

Accredited by NBA, New Delhi

Approved by AICTE, New Delhi.

ಪಿ.ಇ.ಎಸ್. ತಾಂತ್ರಿಕ ಮಹಾವಿದ್ಯಾಲಯ

ಮಂಡ್ಯ–571 401, ಕರ್ನಾಟಕ

(ವಿ.ಟಿ.ಯು, ಬೆಳಗಾವಿ ಅಡಿಯಲ್ಲಿನ ಸ್ವಾಯತ್ತ ಸಂಸ್ಥೆ)

Ph : 08232- 220043, Fax : 08232 – 222075,Web : www.pescemandya.org

Preface

PES College of Engineering, Mandya, started in the year 1962, has become autonomous in the academic year 2008-09. Since, then it has been doing the academic and examination activities successfully. The college is running eight undergraduate and eight Postgraduate programs. It consists of six M.Tech programs, which are affiliated to VTU. Other postgraduate programs are MBA and MCA.

India has recently become a Permanent Member by signing the Washington Accord. The accord was signed by the National Board of Accreditation (NBA) on behalf of India on 13th June 2014. It enables not only the mobility of our degree globally but also establishes equivalence to our degrees with that of the member nations such as Taiwan, Hong Kong, Ireland, Korea, Malaysia, New Zealand, Russia, Singapore, South Africa, Turkey, Australia, Canada and Japan. Among other signatories to the international agreement are the US and the UK. Implementation of Outcome Based Education (OBE), has been the core issue for enabling the equivalence and of Indian degrees and their mobility across the countries.

Our Higher Educational Institution has adopted the semester structure with OBE scheme and grading system.

The credit based OBE semester system provides flexibility in designing curriculum and assigning credits based on the course content and hours of teaching.

The OBE, emphasize setting clear standards for observable, measurable outcomes of programs in stages. There lies a shift in thinking, teaching and learning processes moving towards Students Centric from Teacher Centric education. OBE standards focus on mathematics, language, science, attitudes, social skills & moral values.

The key features which may be used to judge, if a system has implemented an outcome based education system is mainly Standard based assessments that determines whether students have achieved the stated standard. Assessments may take any form, so long as the process actually measure whether the student knows the required information or can perform the required task. Outcome based education is a commitment that all students of all groups will ultimately reach the same minimum standards. Outcome Based Education is a method or means which begins with the end in mind and constantly emphasizes continuous improvement.

In order to increase the Industry/Corporate readiness, many Soft Skills and Personality Development modules have been added to the existing curriculum of 2013-14.Industry Interactions have been made compulsory to enhance the field experience. In order to enhance creativity and innovation Mini Project is included in all undergraduate programs.

(Dr.H.V.RAVINDRA)
Dean (Academic)
Professor,
Dept. of Mechanical Engg.

(B.DINESH PRABHU)
Deputy Dean (Academic)
Associate Professor,
Dept. of Automobile Engg

P.E.S.COLLEGE OF ENGINEERING, MANDYA-571401
(KARNATAKA)
(An Autonomous Institution under VTU, Belagavi)

Vision

"An institution of high repute, imparting quality education to develop innovative and Humane engineers"

Mission

"Committed to develop students potential through high quality teaching- learning processes and state of the art infrastructure"

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

About the Department

The Department of Computer Science and Engineering was established in 1983. The department offers B.E. program with an intake of 120 students, M.Tech. in Computer Science and Engineering with 18 students, M.Tech. in Computer Engineering with 24 students and also Ph.D. programme. Currently the strength of teaching faculty is 32 and that of non teaching staff is 14. The teacher - student ratio is 1:16. The department has a research centre under VTU and University of Mysore, with 2 research guides and 8 research students. During the last five years, the department has published 85 technical papers in international/national journals/conferences. So far, the department has organized four international and 8 national conferences. The department is equipped with all the required infrastructure, laboratories, class rooms, departmental library. The departments wish to achieve the mission of developing and nourishing computer science engineers through well-trained, committed and experienced faculty members. Faculty members of the departments are involved in research activities in different fields such as Image Processing, Pattern Recognition, Data Mining, Wireless Networks, Big Data Analytics and Computer Vision.

Vision

"*To develop globally competent computer professionals by exploring latest technologies through continuous learning, research and innovation."*

Mission

"To impart quality technical education in modern practices of Computer Science and Engineering through competent faculties, state of the art teaching–learning infrastructure and methodologies".

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Programme Educational Objectives (PEOS) are

I. *To provide students with a strong basis in the mathematical, scientific and engineering fundamentals to solve computer science engineering problems and to prepare them for employment, higher learning, R&D and consultancy.*
II. *To provide technical knowledge, skills and awareness of current technologies of computer science engineering and to develop an ability to design and provide novel engineering solutions for software/hardware problems through entrepreneurial skills.*
III. *To provide an exposure to emerging cutting edge technologies to work as teams on multidisciplinary projects with effective communication skills and leadership qualities.*
IV. *To provide an ability to function ethically and responsibly in a rapidly changing environment by applying innovative technologies that allows them to become effective professionals in Computer Science to sustain a life-long career in related areas.*

Program outcomes (POs)

A graduate of the Computer Science and Engineering Program will demonstrate

a) An ability to apply knowledge of computing, mathematical foundations, computer science and engineering fundamentals. (Fundamental engineering analysis skills).
b) An ability to analyze a problem, and identify and formulate the computing requirements appropriate to its solution (Information retrieval skills).
c) An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs. (creative skills).
d) An ability to identify, formulate and solve computer science engineering problems for obtaining its solutions using the appropriate computing and engineering requirements (engineering problem solving skills).
e) An ability to use current techniques, skills, and modern engineering tools necessary for computing and engineering practice (practical engineering analysis skills).
f) The broad education necessary to understand the impact of engineering solutions in a global and societal context (engineering impact assessment skills).

g) A knowledge of contemporary issues (social awareness).
h) An understanding of professional, ethical, legal, security and social issues and responsibilities (professional integrity).
i) An ability to function effectively individually and on teams to accomplish a common goal (teamwork).
j) An ability to communicate effectively, both in writing and orally with a range of audiences (speaking / writing skills).
k) Recognition of the need for, and an ability to engage in continuing professional development and life-long learning (continuing education awareness).
l) An ability to acquire concepts to integrate computer science and engineering principles in the construction of software and hardware systems of varying complexity (software hardware interface).
m) An ability to recognize the importance of professional development by pursuing postgraduate studies or face competitive examinations that offer challenging and rewarding careers in computing (successful career and immediate employment).

These programme outcomes (POs) are achieved through an array of courses. To ensure the achievement of POs, the course learning outcomes (CLOs) are so formulated that they address these POs.

P.E.S COLLEGE OF ENGINEERING, MANDYA
(An Autonomous Institution under VTU)

SCHEME OF TEACHING AND EXAMINATION

V Semester

Sl. No.	Course code	Course Title	Teaching Dept.	Hours Pattern	Total Credits	Examination Marks			Exam Duration in hours
				L:T:P::H		CIE	SEE	Total	
1	P13CS51	Operating Systems	CS	3:2:0:5	4	50	50	100	3
2	P13CS52	Data Base Management System	CS	3:0:2:5	4	50	50	100	3
3	P13CS53	Data Communications	CS	4:0:0:4	4	50	50	100	3
4	P13CS54	System Software	CS	4:0:0:4	4	50	50	100	3
5	P13CS55	Computer Graphics and Visualization	CS	4:0:0:4	4	50	50	100	3
6	P13CS56	Management, Entrepreneurship& IPR	CS	2:2:0:4	3	50	50	100	3
7	P13CSL57	Microprocessor Lab	CS	0:1:2:3	1.5	50	50	100	--
8	P13CSL58	System software Lab	CS	0:1:2:3	1.5	50	50	100	--
9	P13HU59	Professional and Efficient Avocation-I(PEA-I)	HS&M	2:0:0:2	--	(50)	---	---	---
10	P13CSL510	Industry visit and interaction-II	CS	0:0:1:1	--	(50)	--	--	--
Total					26	400	400	800	

VI Semester

Sl. No.	Course code	Course Title	Teaching Dept.	Hours Pattern	Total Credits	Examination Marks			Exam Duration in hours
				L:T:P:H		CIE	SEE	Total	
1	P13CS61	Web Technologies	CS	3:0:2:5	4	50	50	100	3
2	P13CS62	Computer Networks	CS	3:2:0:5	4	50	50	100	3
3	P13CS63	Compiler Design	CS	4:0:0:4	4	50	50	100	3
4	P13CS64	Advanced Computer Architecture	CS	4:0:0:4	4	50	50	100	3
5	P13CS65	Software Engineering	CS	2:2:0:4	3	50	50	100	3
6	P13CS66*	Elective – I	CS	4:0:0:4	4	50	50	100	3
7	P13CSL67	Computer Graphics Lab	CS	0:1:2:3	1.5	50	50	100	--
8	P13CSL68	Operating Systems Lab	CS	0:1:2:3	1.5	50	50	100	--
9	P13HU69	Professional and Efficient Avocation-I (PEA-II)	HS&M	2:0:0:2	---	(50)	---	---	---
10	P13CSL610	Mini Project - II	CS	0:0:1:1	--	(50)	--	--	--
Total					26	400	400	800	

* PEA-I, , Industry visit and interaction-II, PEA-II, Mini Project: All students shall have to pass this mandatory learning courses before completion of VIII-Semester

Elective-I

Sl. No.	Course Code	Course Title
1	P13CS661	Data Warehousing and Data Mining
2	P13CS662	Multimedia Computing
3	P13CS663	Java and J2EE
4	P13CS664	Data Compression
5	P13CS665	C# and .Net Programming

EVALUATION SCHEME							
Scheme	**Weightage**	**Marks**	**Event Break Up**				
CIE	50%	50	**Test I**	**Test II**	**Quiz I**	**Quiz II**	**Assignment**
			35	35	5	5	10
SEE	50%	100	**Questions to Set: 10**		**Questions to Answer: 5**		
Scheme of SEE Question Paper (100 Marks)							
Duration: 3Hrs			**Marks: 100**		**Weightage: 50%**		

- Each of the two questions set shall be so comprehensive as to cover the entire contents of the unit.
- There will be direct choice between the two questions within each Unit
- Total questions to be set are 10. All carry equal marks of 20
- The number of subdivisions in each main question shall be limited to three only
- Number of questions to be answered by students is 5

Course Title : OPERATING SYSTEM			
Course Code: P13CS51	**Semester : V**	**L:T:P:H- 3 : 2 : 0 : 5**	**Credits:4**
Contact Period : Lecture :52 Hrs., Exam: 3Hrs.		**Weightage :CIE: 50% SEE:50%**	

Prerequisites:

- Programming language such as C.
- Data Structures and Algorithms

Course Learning Objectives

The students should be able to

1. **Explain** operating system structure, services and **determine** the interfaces between OS and other components of a computer system.
2. **Illustrate** the main principles and techniques used to implement processes and threads as well as the different algorithms for process scheduling.
3. **Analyze** the main problems related to concurrency and the different synchronization mechanisms.
4. **Describe** different approaches of memory management and **Apply** different page replacement algorithms to resolve page faults.
5. **Describe** the structure and organization of file system and **analyze** the data storage in secondary storage and **understand** the protection issues in computer systems.

Course content

Unit – 1

Introduction to operating systems

Overview: Need of operating systems, Computer System organization, Computer System architecture, Operating System structure, Operating System operations, Process management, Memory management, Storage management, Protection and security, Distributed system, computing environments.

System structure: Operating System Services, User- Operating System interface, System calls, Types of system calls, System programs, Operating System design and implementation, Operating System structure, Virtual machines, System boot. 10 Hours

Unit – 2

Process management

Process concepts: Overview, Process scheduling, operations on processes, Inter-process communication.

Multi-Threaded Programming: Overview, Multi-threading models, Thread Libraries, threading issues.

Process Scheduling: Basic concepts, Scheduling criteria, Scheduling algorithms, Multiple-Processor scheduling, thread scheduling 11 Hours

Unit – 3

Process synchronization

Synchronization: Background, The Critical section problem, Peterson's solution, Synchronization hardware, Semaphores, Classical problems of synchronization, Monitors

Deadlocks: System model, Deadlock characterization, Methods for handling deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection and recovery from deadlock 10 Hours

Unit – 4

Memory and i/o management

Memory Management Strategies: Background, Swapping, Contiguous memory allocation, Paging, Structure of page table, Segmentation.

Virtual Memory Management: Background, Demand paging, Copy-on-write, Page replacement, Allocation of frames, Thrashing.

I/O Systems: Overview, I/O hardware, Application I/O interface, STREAMS. 11 Hours

Unit – 5

Storage management and protection

File system: File concept, Access methods, Directory structure, File system mounting, File sharing, Protection.

Implementing File System: File system structure, File system implementation, Directory implementation, Allocation methods, Free space management.

Secondary storage structures: Mass storage structures, Disk structure, Disk attachment, Disk scheduling, Disk management, Swap space management.

Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix, Implementation of access matrix, Access control, Revocation of access rights.

10 Hours

Text Books:

1. **Operating System Principles –** Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, 7th edition, Wiley-India, 2006.

Reference Books:

1. **Operating Systems: A Concept Based Approach** – D.M Dhamdhere, 2nd Edition, Tata McGraw- Hill, 2002.
2. **Operating Systems** – William Stallings, 5th Edition, PHI, 2006.
3. **Operating Systems** – Harvey M Deital, 3rd Edition, AddisonWesley, 1990

Course Outcomes

The students should be able to

1. **Describe** the basic principles of operating systems.
2. **Describe** the concept of process management and **Apply** different CPU scheduling algorithms.
3. **Write** algorithmic solutions to process synchronization problems, and identify the deadlock status of the system.
4. **Apply** different algorithms for memory management and **Describe** device, I/O management functions in operating systems.
5. **Implement** file system and **Show** the data storage using disk scheduling algorithms and **describe** protection issues in computer systems.

Topic Learning Objectives

Unit 1

Topic : Operating System structure, need and operations. Distributed system, computing environments. Operating System Services, User- Operating System interface, System calls, System programs, Operating System design and implementation, Virtual machines.

Learning Objectives:

1. **Explain** the need of an operating system.(L1)
2. **Identify** different components of operating system and Explain how an operating system works.(L2)
3. **Explain** the organization of operating system.(L2)
4. **Explain** the different operations performed by the operating system(L2)
5. **Explain** how the operating system components are used in variety of computing environments.(L2)
6. **List** and **explain** the different services carried out by the operating system.(L1,L2)
7. **Explain** different approaches for users to interface with the computer. (L2)
8. **Define** a system call and List and Explain major categories of system calls. (L1,L2)
9. **List** and **explain** different categories of system programs, Operating system structure. (L1,L2)
10. **Understand** the concept of Virtual machines and system Boot. (L1)

Unit 2

Topic: Operations on processes, Inter-process communication. Multi-Threaded Programming, Process Scheduling algorithms, Multiple-Processor scheduling, thread scheduling.

Learning Objectives:

1. **Define** a process, Discuss different states of a process.(L1)
2. **Explain** the concept of process scheduling.(L2)
3. **List** different operations on processes.(L2)
4. **Differentiate** between different models of Interprocess communication.(L4)
5. **Explain** the multithreaded models.(L2)
6. **Distinguish** between process and a thread.(L4)
7. **Explain** different threading models and the threading issues.(L2)
8. **Explain** the concept of process scheduling.(L2)
9. **Compare** different scheduling algorithms.(L4)
10. **Write** program to create ,execute and terminate the processes.(L6)

Unit 3

Topic The Critical section problem, Peterson's solution, Synchronization hardware, Semaphores, Monitors, Deadlock characterization, Methods for handling deadlocks, Deadlock prevention, and avoidance, Deadlock detection, deadlock recovery methods.

Learning Objectives :

1. **Define** process synchronization.(L1)
2. **Explain** critical section problem.(L2)
3. **Explain** the need of semaphores.(L2)
4. **Discuss** different solutions for critical section problems.(L2)
5. **Develop** monitor solutions for critical section problems.(L6)
6. **Explain** necessary conditions for deadlock.(L2)
7. **List** the methods for handling deadlocks.(L1)
8. **Describe** the methods for recovering from deadlock.(L2)
9. **Write** monitor solutions to the dining philosopher's problem.(L6)
10. **Determine** if the system is in safe state or not(L4).

Unit 4

Topic: Memory Management Strategies Swapping, Paging, Segmentation, Virtual Memory Management, Demand paging, Copy-on-write, Page replacement algorithms, Allocation of frames, I/O Systems,I/O hardware, Application I/O interface, STREAMS.

Learning Objectives:

1. **Demonstrate** the concept of paging.(L3)
2. **Explain** the need of segmentation.(L2)
3. **Compare** various memory management strategies.(L4)
4. **Explain** the basic method of segmentation with respect to memory management.(L2)
5. **Solving** problems related to paging and segmentation.(L4)
6. **Differentiate** between internal and external fragmentation.(L4)
7. **Explain** the need of virtual memory.(L2)
8. **Apply** the concept of demand paging.(L3)
9. **Explain** the different steps in handling page faults.(L2)
10. **Discuss** the problems on page replacement algorithms.(L2)

Unit 5

Topic: File System concept, Directory structure, implementation of File system and Directory, Allocation methods, Free space and Swap space management. Mass storage and Disk structure, Disk attachment, scheduling and management.

Protection: Goals and Principles of protection, Domain of protection, Access matrix, Implementation of access matrix, Access control, Revocation of access rights.

Learning Objectives :

1. **Explain** the structure of file file system. (L2)
2. **Describe** the structure of directory.(L2)
3. **List** the different file types with their functions.(L1)
4. **Distinguish** between different file allocation methods.(L4)
5. **Solve** problems related to file allocation.(L4)
6. **Discuss** the secondary storage structure.(L2)
7. **Explain** different disk scheduling algorithms.(L2)
8. **Explain** the need of swap space.(L2)
9. **Discuss** the goals of protection .(L2)
10. **Illustrate** the access matrix with an example.(L3)

Review Questions

Unit - 1Define operating system Explain 2 different views of OS.

2. **Explain** dual-mode operation of the OS with neat block diagram.
3. Write the differences between peer-to-peer computing and client server computing?
4. With diagram **explain** migration of data from disk to register.
5. **List and explain** the set of services provided by the OS to the users and also the system.
6. With an example **illustrate how system calls are used.**
7. **Describe** layered approach and modular approach for designing the OS.
8. **Defend** modular approach is efficient than layered and microkernel approach?
9. **Describe** VMware and JVM virtual machines.
10. With diagram **illustrate** the difference between VM and non-VM.

Unit - 2Define process? With a block diagram explain the process structure in memory.

2. What is process control block? What are its contents? **Explain.**
3. **Define** process scheduler? **Describe** different types of schedulers used in process scheduling.
4. **Show** the steps that followed to create, execute and terminate a process on Unix system.
5. **List** out the **difference** between shared memory model and message passing model.
6. **Explain** different multithreading models?
7. **Illustrate** the following scheduling algorithm with example
   I. FCFS
   II. SJF
   III. PS
8. Consider the following set of process.

Process	Burst time	Arrival time
P1	10	0
P2	6	1
P3	2	2
P4	4	3
P5	8	4

Draw the gantt chart and **calculate** turnaround time and average waiting time for the following algorithm.

I. FCFS
II. Preemptive SJF

Which is the best scheduling algorithm for the above case?

9. Consider the following set of process.

Priority	Process	Burst time	Arrival time
3	P1	10	0
5	P2	6	1
2	P3	2	2
1	P4	4	3
4	P5	8	4

Draw the gantt chart and **calculate** turnaround time and average waiting time for the following algorithm.

I. PS
II. RR(2 ms)

Which is the best scheduling algorithm for the above case?

10. **Explain** multilevel queue and multilevel feedback queue scheduling algorithm

Unit -3Write the requirements to the critical section problem solution?

2. **Write** Peterson's solution to the critical section problem.
3. **List out** the different types of semaphores? Explain each with examples.
4. **Define** monitor? **Describe** the implementation of monitors using semaphores.
5. **State** Dining philosophers problem? **Write** the solution to this problem using semaphores.
6. **Write and Explain** monitor solution to the Dining Philosopher problem.
7. **List** the four necessary conditions must hold for an occurrence of the deadlock.
8. **List** and explain the Data Structures used to implement Banker's algorithm.
9. **Write** safety algorithm and resource request algorithm.
10. Given a system with a total resources of R1 (9), R2 (9), R3 (10) and R4 (6). Consider the following snapshot of the system

Process	Allocation				Max			
	R1	R2	R3	R4	R1	R2	R3	R4
P1	2	2	3	1	6	7	6	3
P2	2	0	2	2	7	5	7	6
P3	1	2	0	0	2	3	0	0
P4	2	2	2	1	4	7	5	3
P5	1	2	0	0	3	5	3	1

Use banker's algorithm to answer the following questions.

a) is the system is in a safe state?
b) Can the request made by process P4(0,1,2,1) be granted safely?

Unit - 4 On a paging system with a page table containing 512 entries of 16 bits each and a page size of 1024, **Calculate**

i) How many bits in the logical address specifies the page number?
ii) How many bits in the logical address specifies the offset within the page?
How many bits are in the logical address?

2. On a system, **apply** first fit, best fit, worst fit allocation strategies to allocates holes/memory for the additional requisite for 10k, 25k, 120k. Assume memory is allocated as specified in figure.

Used	Hole	U	H	U	H	U	H	U	H	U	H
10k	30k	20	15	10	20	50	50	10	45	20	30

3. Define hit ratio. **Calculate** the effective access time of 80% hit ratio. Assume, searching a page number in TLB takes 20 ns and a mapped memory access takes 100 ns.
4. **Describe** the concept of paging in memory management scheme.

5. With neat diagram **explain** the concept of paging with TLB.
6. **List and explain** the different techniques used for structuring the page table.
7. **Define** segment? **Explain** the hardware support for segmentation.
8. **Define** demand paging? **Describe** the procedure for handling the page fault with neat labeled diagram.
9. **Show** the steps involved in a DMA transfer.
10. Consider the following reference streams 1 2 3 0 1 4 0 0 1 2 3 4 2 0 1 2 3. **Find** How many page fault occurs for the following replacement algorithm, assuming 3 & 5 frames.
    I. FIFO
    II. LQU
    III. Optimal

    And which one is efficient for the given reference string.

Unit – 5

1. **List and explain** file attributes and file operations.
2. **Define** file? **Describe** the different file accessing methods.
3. With neat diagram **explain** virtual file system.
4. **Describe** two different ways used for accessing disk storage.
5. With example **explain** different file allocation methods and bring out the advantages and disadvantages of each.
6. With a diagram **describe** the two-level Directory.
7. **Explain** the different approaches of free space management.
8. With an example **illustrate** the concepts of Access matrix.
9. **Describe** how access matrix can be implemented effectively.
10. **List and explain** the schemes implement the revocation for capabilities.

Lesson Plan

Unit -1

1. Need of operating systems, Computer System organization.
2. Computer System architecture, Operating System structure.
3. Operating System operations, Process management, Memory management, Storage management, Protection and security.
4. Asymptotic notations and basic efficiency classes.
5. Distributed system.
6. Operating System Services, User- Operating System interface, System calls.
7. Types of system calls, System programs.
8. Operating System design and implementation.
9. Operating System structure.
10. Virtual machines, System boot.

Unit – 2

1. Process scheduling.
2. Operations on processes, Inter-process communication..
3. Inter-process communication. .
4. Overview, Multi-threading models.
5. Thread Libraries, threading issues.
6. Basic concepts, Scheduling criteria.
7. Scheduling algorithms explanation.
8. Scheduling algorithms problems.
9. Scheduling algorithms problems.
10. Multiple-Processor scheduling,
11. Thread scheduling.

Unit – 3

1. Synchronization, The Critical section problem.
2. Peterson's solution.
3. Semaphores.
4. Monitors.
5. Deadlock characterization, Methods for handling deadlocks.
6. Deadlock prevention.
7. Deadlock avoidance.
8. Deadlock detection.
9. Deadlock detection.
10. Recovery from deadlock.

Unit – 4

1. Memory Management, Swapping.
2. Contiguous memory allocation, paging.
3. Paging, Structure of page table.
4. Segmentation.
5. Virtual Memory Management, Demand paging, Copy-on-write.
6. Page replacement algorithms.
7. Page replacement algorithms.
8. Allocation of frames.
9. Allocation of frames, Thrashing.
10. I/O Systems, I/O hardware.
11. Application I/O interface, STREAMS.

Unit – 5

1. File concept, Access methods, Directory structure.
2. File system mounting, File sharing, Protection.
3. File system structure, File system implementation, Directory implementation.
4. Allocation methods, Free space management.
5. Secondary storage structures, Mass storage structures, Disk structure.
6. Disk attachment, Disk scheduling.
7. Disk management, Swap space management.
8. Goals of protection, Principles of protection.
9. Domain of protection, Access matrix, Implementation of access matrix.
10. Access control, Revocation of access rights.

A. Course Articulation Matrix (CAM)

Course Outcome (CO)	Program Outcome (ABET/NBA-(3a-m))													
		a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Describe the basic principles of operating systems.	**L1**	-	-	M	H	-	-	-	**-**	**M**	M	M	L	M
Describe the concept of process managent concept and **Apply** different CPU scheduling algorithms.	**L1, L3**	H	H	M	M	H	-	-	-	-	-	H	M	M
Write algorithmic solutions to process synchronization problems, and identify the deadlock status of the system.	**L6**	L	M	-	H	M	-	-	M	-	-	M	M	M
Apply different algorithms for memory management and **Desribe** device , I/O management functions in operating systems.	**L3**	-	**-**	**L**	**L**	L	**-**	**M**	**-**	**-**	-	-	M	M
Implement file system, **Show** the data storage using disk scheduling algorithms and **desribe** protection issues in computer systems.	**L5, L1**	-	M	-	M	-	-	-	-	-	L	L	M	M
L- Low, M- Moderate, H-High														
Illustration of CAM of Operating systems														

B. Course Assessment Matrix (CaM)

Course Outcome (CO)	Program Outcome (ABET/NBA-(3a-m))													
		a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Describe the basic principles of operating systems.	**L1**	-	-	2	3	-	-	-	**-**	2	2	2	1	2
Describe the concept of process management concept and **Apply** different CPU scheduling algorithms.	**L1,L 3**	3	3	2	2	3	-	-	-	-	-	3	2	2
Write algorithmic solutions to process synchronization problems, and identify the deadlock status of the system.	**L6**	1	2	-	3	2	-	-	2	-	-	2	2	2
Apply different algorithms for memory management and **Describe** device, I/O management functions in operating systems.	**L3**	-	**-**	**1**	**1**	1	**-**	**2**	**-**	**-**	-	-	2	2
Implement file system, **Show** the data storage using disk scheduling algorithms and **describe** protection issues in computer systems.	**L5, L1**	-	2	-	2	-	-	-	-	-	1	1	2	2
1 – Low, 2 – Moderate and 3 – High														

Course Title: Data Base Management System			
Course Code: P13CS52	**Semester : V**	**L-T– P-H : 3:0:2:5**	**Credit:4**
Contact period : Lecture: 52 Hours	**Exam:3 hours**	**Weightage: CIE: 50% SEE:50%**	

Prerequisites

- Computer systems
- Relations and Set theory

Course learning objectives

This course aims to

1. State the importance of DBMS and explain how DBMS is better than traditional File Processing Systems.
2. Analyze the basic structure of Database and recognize the different views of the database.
3. Identify attributes, entities and relationship of the given system and draw Entity Relationship Diagrams.
4. Analyze and use Relational Data Model, while comparing with other data models.
5. Formulate data retrieval queries in SQL and the Relational Algebra and Calculus.
6. Apply normalization steps in database design using the design guidelines and functional dependencies
7. Understand and explain the terms like Transaction Processing and Concurrency Control.
8. Understand types of database failure and recovery

Course content

Unit-1

INTRODUCTION

An example: Characteristics of Database approach; Advantages of using DBMS approach; A brief history of database applications; Data models, schemas and instances; Three-schema architecture and data independence; Database languages and interfaces; The database system environment.

ENTITY-RELATIONSHIP MODEL:

Using High-Level Conceptual Data Models for Database Design; An Example Database Application; Entity Types, Entity Sets, Attributes and Keys; Relationship types, Relationship Sets, Roles and Structural Constraints; Weak Entity Types; Refining the ER Design; ER Diagrams, Naming Conventions and Design Issues; Relationship types of degree higher than two. **10 Hours**

Unit-2

RELATIONAL MODEL AND RELATIONAL ALGEBRA:

Relational Model Concepts; Relational Model Constraints and Relational Database Schemas; Update Operations, Transactions and dealing with constraint violations; Unary Relational Operations: SELECT and PROJECT; Relational Algebra Operations from Set Theory; Binary Relational Operations : JOIN and DIVISION; Additional Relational Operations; Examples of Queries in Relational Algebra; Relational Database Design Using ER- to-Relational Mapping. **10 Hours**

Unit-3

STRUCTURED QUERY LANGAUGE

SQL Data Definition and Data Types; Specifying basic constraints in SQL; Basic Retrieval Queries in SQL, INSERT, DELETE, and UPDATE Statements in SQL, More complex SQL Retrieval Queries, Specifying constraints as Assertion and Actions as Trigger; Views (Virtual

Tables) in SQL; Additional features of SQL; Schema Change Statements in SQL; Additional Features of SQL **12 Hours**

Unit-4

DATABASE DESIGN:

Informal Design Guidelines for Relation Schemas; Functional Dependencies; Normal Forms Based on Primary Keys; General Definitions of Second and Third Normal Forms; Boyce-Codd Normal Form; Multi valued Dependencies and Fourth Normal Form; Join Dependencies and Fifth Normal Form. **10 Hours**

Unit – 5

TRANSACTION PROCESSING CONCEPTS:

Introduction to Transaction processing; Transactions and System concepts; Desirable properties of transactions; Characterizing Schedules based on Serializability.

Concurrency control and recovery techniques:

Two-phase locking techniques for concurrency control; concurrency control based on timestamp ordering; recovery concepts; recovery techniques based on deferred update; recovery techniques based on immediate update; shadow paging; **10 Hours**

Text Books:

1. **Fundamentals of Database Systems –** Elmasri and Navathe, 6th Edition, Addison-Wesley, 2011

Reference Books:

1. **Data Base System Concepts** – Silberschatz, Korth and Sudharshan, 5th Edition, Mc-GrawHill, 2006.
2. **An Introduction to Database Systems** – C.J. Date, A. Kannan, S.Swamynatham, 8th Edition, Pearson Education, 2006.
3. **Database Management Systems** – Raghu Ramakrishnan and Johannes Gehrke – 3rd Edition, McGraw-Hill, 2003.

Course outcomes

At the end of the course the student should be able to

1. **Explain** the fundamental concepts, applications and functions of Databases and **Design** an E-R diagram for a given mini world description.
2. **Construct** relational database schema for the given application and **develop** queries in relational algebraic expressions.
3. **Develop** basic and complex queries using SQL to retrieve the required information from database.
4. **Understand** the design guidelines with functional dependencies and **Apply** the suitable normal form to normalize the given data base.
5. **Describe** the roles of transaction processing, concurrency control and recovery techniques in data base design.

Topic learning objectives

After learning all the topics of unit-I, the student is able to

1. Define database and its characteristics (L1)
2. Distinguish between conventional file system and Database systems(L2)
3. List out the advantages of database approach over conventional file system L1)
4. List out the data base applications .(L1)
5. Differentiate between the data models, schemas and instances .(L4)
6. Explain the 3-schema architecture of database.(L2)
7. Describe the different database languages and interfaces.(L1)

8. Describe the different database system components.(L1)
9. Identify Entity types, Entity sets, Attributes and Keys from a given application.(L1)
10. Design the database using the High-Level Conceptual data models.(L5)
11. Differentiate between relationship instance, relationship type and a relationship set.(L4)

After learning all the topics of unit-II, the student is able to

1. Define Domains, Tuples, Relations, super key, candidate key, and primary key.(L1)
2. Explain the relational model concepts and constraints. (L2)
3. List the different Database update anomalies.(L1)
4. Identify the type of constraints violated by each update operation. (L1)
5. Apply the different unary relational algebra operations on Relations.(L3)
6. Describe the different set operations using the relational algebra.(L1)
7. Differentiate between different types of JOIN operations.(L4)
8. Explain the DIVISION operations on Relations.(L2)
9. Write a given query in relational algebra.(L5)
10. Apply the ER-to-Relational Mapping algorithm to create the Relational Database Schema.(L3)

After learning all the topics of unit-III, the student is able to

1. Definition of SQL, data definition and data types allowed in SQL.(L1).
2. Explain the main types of constraints with examples. (L2)
3. Writing the basic and more complex queries in SQL for the given statement.(L5)
4. Use of Insert, Delete and Update commands in SQL Statements. (L3)
5. List the statements for modifying schemas, Tables and constraint.(L1)
6. Explain the Assertion and Trigger; Views (Virtual Tables) in SQL.(L2)

After learning all the topics of unit-IV, the student is able to

1. Explain informal guidelines for the database design.(L2)
2. Illustration with an example for insertion, deletion and modification anomalies.(L4)
3. Explain the concept of functional dependency.(L2)
4. Apply inference rules for functional dependency.(L3)
5. Define 1NF, 2NF, 3NF and BCNF.(L1)
6. Design a Normalized database up to BCNF database.(L5)
7. Apply normal forms to normalize the given relation schemas.(L3)
8. Define Multi valued dependencies 4NF and 5NF.(L1)
9. Explain join dependencies, inclusion dependencies and other dependencies.(L2)
10. Design a Normalized database up to 5NF for sample example.(L5)

After learning all the topics of unit-V, the student is able to

1. Explain the concepts of Transaction processing.(L2)
2. List desirable properties of Transaction.(L1)
3. Describe Schedules and their characteristics.(L1)
4. Describe Transaction support in SQL.(L1)
5. Explain the necessity of concurrency control. (L2)
6. Describe locking and recovery techniques.(L1)
7. Explain recovery techniques based on deferred update and immediate update. (L2)
8. Differentiate between deferred update and immediate update recovery techniques.(L4)
9. Explain the shadow paging(L2)
10. Design schedules based on recoverability and serializability(L5)

Review questions

1. Define database and its characteristics
2. Distinguish between conventional file system and Database systems
3. List out the advantages of database approach over conventional file system
4. List out the data base applications .
5. Differentiate between the data models, schemas and instances .
6. Explain the 3-schema architecture of database.
7. Describe the different database languages and interfaces.
8. Describe the different database system components.
9. Identify Entity types, Entity sets, Attributes and Keys from a given application.
10. Design the database using the High-Level Conceptual data models.
11. Differentiate between relationship instance, relationship type and a relationship set.
12. Identify relationships among different entities.
13. Identify Roles, structural constraints and weak entity types.
14. Design an E-R diagram for the given mini world description.
15. Define Domains, Tuples, Relations, super key, candidate key, and primary key.
16. Explain the relational model concepts and constraints.
17. List the different Database update anomalies.
18. Identify the type of constraints violated by each update operation.
19. Apply the different unary relational algebra operations on Relations.
20. Describe the different set operations using the relational algebra.
21. Differentiate between different types of JOIN operations.
22. Explain the DIVISION operations on Relations.
23. Write a given query in relational algebra.
24. Apply the ER-to-Relational Mapping algorithm to create the Relational Database Schema.
25. Definition of SQL, data definition and data types allowed in SQL.
26. Explain the main types of constraints with examples.
27. Construct tables using different data types using SQL command.
28. Writing the basic and more complex queries in SQL for the given statement.
29. Use of Insert, Delete and Update commands in SQL Statements.
30. List the statements for modifying schemas, Tables and constraint.
31. Explain the Assertion and Trigger; Views (Virtual Tables) in SQL.
32. Explain informal guidelines for the database design.
33. Illustration with an example for insertion, deletion and modification anomalies.
34. Explain the concept of functional dependency.
35. Apply inference rules for functional dependency.
36. Define 1NF, 2NF, 3NF and BCNF.
37. Design a Normalized database up to BCNF database.
38. Apply normal forms to normalize the given relation schemas.
39. Define Multi valued dependencies 4NF and 5NF.
40. Design a Normalized database up to 5NF for sample example.
41. Explain the concepts of Transaction processing.
42. List desirable properties of Transaction.
43. Describe Schedules and their characteristics.
44. Describe Transaction support in SQL.
45. Explain the necessity of concurrency control.
46. Describe locking and recovery techniques.
47. Explain recovery techniques based on deferred update and immediate update.
48. Differentiate between deferred update and immediate update recovery techniques.
49. Explain the shadow paging.
50. Design schedules based on recoverability and serializability.

51. Interpret a set of transactions to create serializability.

Lesson plan

Unit-1

1. Introduction
2. Characteristics of Database approach advantages of using DBMS
3. Data models, schemas and instances.
4. Three-schema architecture and data independence.
5. The database system environment.
6. Entity Types, Entity Sets, Attributes and Keys.
7. Relationship types, Relationship Sets, Roles and Structural Constraints; Weak Entity Types;.
8. ER Diagrams,
9. Naming Conventions and Design Issues.
10. Relationship types of degree higher than two

Unit-2

1. Relational Model Concepts;
2. Relational Model Constraints and Relational Database Schemas.
3. Unary Relational Operations: SELECT and PROJECT
4. Relational Algebra Operations from Set Theory
5. Binary Relational Operations: JOIN
6. Division operator
7. Examples of Queries in Relational Algebra
8. ER- to-Relational Mapping

Unit-3

1. SQL Data Definition and Data Types,
2. INSERT, DELETE, and UPDATE Statements in SQL.
3. Examples: Basic Retrieval Queries in SQL
4. The complex SQL Retrieval Queries
5. nested queries -examples
6. correlated queries-examples
7. Assertion and Trigger
8. Views (Virtual Tables) in SQL
9. Additional features of SQL
10. Schema Change Statements in SQL

Unit4

1. Informal Design Guidelines for Relation Schemas
2. Functional Dependencies , Normalization
3. Normal Forms Based on Primary Keys
4. First normal form, second normal form
5. Third normal form
6. Boyce-Codd Normal Form
7. Multi valued Dependencies
8. Fourth Normal Form
9. Join Dependencies
10. Fifth Normal Form.

Unit-5

1 Introduction to Transaction processing
2. Transactions and System concepts

3. Desirable properties of transactions
4. Characterizing Schedules based on Serializability
5. Two-phase locking techniques for concurrency control
6. Concurrency control based on timestamp ordering;
7. Recovery techniques based on deferred update
8. Recovery techniques based on immediate update
9. Shadow paging.

Course Articulation matrix

Course outcomes	Program Outcome - * (General)												
	a	b	c	d	e	f	g	h	i	j	k	l	m
Explain the fundamental concepts, applications and functions of Databases and **Design** an E-R diagram for a given mini world description.	M	H	L	-	L	-	M	-	L	L	-	-	H
Construct relational database schema for the given application and **develop** queries in relational algebraic expressions.	M	H	M	-	H	-	-	L	M	-	M	M	H
Develop basic and complex queries using SQL to retrieve the required information from database.	H	H	M	-	M	-	-	-	-	-	-	L	M
Understand the design guidelines with functional dependencies and **Apply** the suitable normal form to normalize the given data base.	M	H	H	-	M	-	-	M	-	-	L	L	-
Describe the roles of transaction processing, concurrency control and recovery techniques in data base design.	L	L	M	L	M	-	-	-	-	-	-	-	-

Course Assessment matrix

Course outcomes	Program Outcome - * (General)												
	a	b	c	d	e	f	g	h	i	j	k	l	m
Explain the fundamental concepts, structures, operations, importance, role and applications and functions of different components of Databases. **Develop** the ability to identify required components like entities, attributes and relations for a given mini world description	2	3	1	-	1	-	2	-	1	1	-	-	3
Design an E-R diagram for the real life application and **Construct** relational database schema, given the ER diagram of a database application.	2	3	2	-	3	-	-	1	2	-	2	2	3
Construct relational algebraic expressions and **Develop** SQL to retrieve the database contents.	3	3	2	-	2	-	-	-	-	-	-	1	2
Understand the design process and **Apply** the suitable normal form to normalize the given data base.	2	3	3	-	2	-	-	2	-	-	1	1	-
Describe the roles of transaction processing in DBMS and the role of concurrency control and recovery techniques in data base design.	1	1	2	1	2	-	-	-	-	-	-	-	-

Course Title : DATA COMMUNICATIONS			
Course Code: P13CS53	**Semester : V**	**L-T-P-H: 4 :0: 0:4**	**Credit: 4**
Contact Period : Lecture :52 Hrs., Exam: 3 Hrs.		**Weightage :CIE:50% SEE:50%**	

Prerequisites:

- Modular Arithmetic
- Basic Knowledge of Electronics

Course Learning Objectives

The students should be able to

1. Understand the properties of digital and analog signals, functionality of different layers in OSI and TCP/IP network models and the factors which impact performance of data communication systems
2. Understand the analog and digital transmission, properties of communication medias, and the concept of multiplexing of data on common communication channel
3. Understand different switching circuits, link layer addressing and exemplify the different coding methods and error detection and correction methods for digital data.
4. Understand data link protocols and different media access control
5. Understand the architecture of wired and wireless Local Area Networks (LANs).

Course content

Unit – 1

Introduction : Data Communications, Networks, Network Types, Standards And Administration.

Network Models : Protocol Layering, TCP/IP Protocol Suite, The OSI Model.

Introduction To Physical Layer : Data And Signals, Periodic Analog Signals, Digital Signals, Transmission Impairment, Data Rate Limits, Performance.

11 Hours

Unit – 2

Digital And Analog Transmission: Digital-To-Digital Conversion, Analog-To-Digital Conversion, Transmission Modes, Digital-To-Analog Conversion.

Bandwidth Utilization: Multiplexing, Spread Spectrum.

Transmission Media: Guided Media: Twisted Pair, Optical Cable Unguided Media : Radio Waves, Microwaves, Infrared

11 Hours

Unit – 3

Switching : Introduction, Circuit-Switched Networks, Packet Switching, Structure of A Switch.

Data Link Layer: Introduction, Link Layer Addressing

Error Detection & Correction: Introduction, Block Coding, Cyclic Codes, Checksum, Forward Error Correction

10 Hours

Unit – 4

Data Link Control : DLC Services, Data-Link Layer Protocols, HDLC, Point-To-Point Protocol.

Media Access Control : Random Access, Controlled Access, Channelization.

10 Hours

Unit – 5

Wired LANs: Ethernet: Ethernet Protocol, Standard Ethernet, Fast Ethernet (100 Mbps), Gigabit Ethernet, 10 Gigabit Ethernet.

Wireless LANs: Introduction, Ieee 802.11 Project, Bluetooth- A Wireless Lan Technology

Connecting Devices And Virtual LANs : Connecting Devices, Virtual LANs.

10 Hours

Text Books:

1. Data Communication and Networking, Behrouz A.Forouzan**,** McGraw Hill, 5th Edition, 2012.

Reference Books:

1. Computer Networks, Tanenbaum, 5th Edition, Pearson Education/PHI, 2011.
2. Communication Networks-Fundamental Concepts and key architectures, Alberto Leon-Garcia and Indra Widjaja, Tata Mc-Graw-Hill 2nd Edition, Pearson Education, 2014

Course Outcomes

At the end of the course the student will be able to

- **Describe** different network models and **calculate** the performance of the Network .
- **Select** encoding scheme, multiplexing methods and suitable media for data transmission .
- **Describe** different switching circuits, link addressing and **apply** different error detection and correction methods for digital data.
- **Differentiate** different data link protocols and **select** suitable media access control protocol for data transmission.
- **Explain** the architecture of wired and wireless Local Area Networks (LANs).

Topic Learning Objectives

Unit 1

Topic: Introduction To Data Communications, Networks, Network Types, Standards And Administration. Protocol Layering, TCP/IP Protocol Suite, the OSI Model. Data and Signals, Periodic Analog Signals, Digital Signals, Transmission Impairment, Data Rate Limits, Performance.

Learning Objective:

1. Identify and know the role of different components of data communication.
2. Recognize the different type of connection and topology in a given network.
3. Distinguish the different categories of network based on size.
4. Know about internet hierarchy and protocol standard.
5. Recognize the different network architecture and their functions - TCP/IP Protocol Suite and OSI model.
6. Understand different levels of addressing with respect to network layers - Physical, logical, port and specific addressing.
7. Understand the different characteristics and parameters of Analog and Digital Signals
8. Differentiate between Base band and Broad band transmission for digital data transmission
9. Understand the reasons for transmission impairments and able to find the strength of the signal and noise in the signal
10. Describe the bit rate in the noiseless channel and noisy channel taking into account bandwidth, level of the signal and the quality of the channel
11. Define the performance of the network with respect bandwidth, throughput and latency

Unit 2

Topic : Digital-To-Digital Conversion, Analog-To-Digital Conversion, Transmission Modes, Digital-To-Analog Conversion. Multiplexing, Spread Spectrum. Guided Media: Twisted Pair, Optical Cable Unguided Media : Radio Waves, Microwaves, Infrared

Learning Objective:

1. Understand the characteristics of Digital signal
2. Encode the digital data using –Unipolar, polar, bipolar, multilevel and multitransition methods.

3. Understand different Block coding methods its use in error detection and correction (Adding redundancy bits) (4B/5B, 8B/10B). Scrambling methods for long distance communication (B8ZS, HDB3).
4. Understand different ways of digital data transmission - serial and parallel.
5. Explain different methods of converting digital signal to analog signals with respect to amplitude, frequency and phase of the analog signal.
6. Describe the theory behind how, different stations can share a common communication channel (multiplexing) for digital and analog data (FDM, WDM, TDM) and multiplexing in Wireless communication(FHSS and DSSS)
7. Describe the structure and the properties like data rate, use and performance of different types of guided medias.
8. Understand the frequency bands, propagation methods and some applications for wireless communication.
9. Describe the structure and the properties like data rate, use and performance of different types of guided medias.
10. Understand the frequency bands, propagation methods and some applications for wireless communication.

Unit 3

Topic: Introduction, Circuit-Switched Networks, Packet Switching, Structure Of A Switch. Introduction, Link Layer Addressing Introduction, Block Coding, Cyclic Codes, Checksum, Forward Error Correction.

Learning Objective:

1. Differentiate between different switching networks - Circuit switched networks, Packet switch networks, Message switching circuit networks with structure of switch.
2. Define single bit, burst error, redundancy, forward error correction and retransmission.
3. Apply the principle of VRC and LRC (parity) checksum for error detection.
4. Understand 3 types of Link layer addressing and ARP
5. Define Minimum Hamming distance for a code, which helps in finding the number of errors a code can detect and correct.
6. Compute one bit error for Hamming code.
7. Compute and detect error in a stream of bits - Cyclic code- Cyclic Redundancy Code.
8. Compute internet checksum

Unit 4

Topic: DLC Services, Data-Link Layer Protocols, HDLC, Point-To-Point Protocol. Random Access, Controlled Access, Channelization

Learning Objective:

1. Understand framing, flow and error control and connectionless and connection oriented .
2. Analyze simple, stop and wait protocol and piggybacking
3. Understand services, framing in HDLC
4. Understand services, framing and transition phases of POINT-TO-POINT protocol
5. Understand how the different stations share a common communication media
6. Describe Multiple access Protocols :
   a) Random access Protocols – ALOHA and CSMA protocols
   b) Controlled access Protocols – Reservation, Polling and Token passing
   c) Channelization Protocols – FDMA,TDMA,CDMA
7. Standard Ethernet: 802 .3 MAC- Frame format, Access methods and categories.

Unit 5

Topic: Ethernet Protocol, Standard Ethernet, Fast Ethernet (100 Mbps), Gigabit Ethernet, 10 Gigabit Ethernet. Introduction, IEEE 802.11 Project, Bluetooth- A Wireless LAN Technology Connecting Devices, Virtual LANs.

Learning Objective:

1. Understand characteristics, addressing, access methods of different Ethernets
2. Describe the IEEE 802.11 Architecture, different services provided and access method.
3. Understand Bluetooth Architecture
4. Know the use of connecting devices:
    - Hubs, Repeaters, Bridges, Switches, and Routers.

Review Questions

Unit - 1

1. Identify the five components of a data communications system.
2. What are the three criteria necessary for an effective and efficient network?
3. For n devices in a network, what is the number of cable links required for a mesh, ring, bus, and star topology?
4. When a party makes a local telephone call to another party, is this a point-to-point or multipoint connection? Explain your answer.
5. What is the difference between network layer delivery and transport layer delivery?
6. Match the following to one or more layers of the OSI model:
    a. Route determination
    b. Flow control
    c. Interface to transmission media
    d. Provides access for the end user
7. Explain three types of transmission impairment.
8. A signal has passed through three cascaded amplifiers, each with a 4 dB gain.
9. What is the total gain? How much is the signal amplified?
10. A periodic composite signal with a bandwidth of 2000 Hz is composed of two sine waves. The first one has a frequency of 100 Hz with a maximum amplitude of 20 V; the second one has a maximum amplitude of 5 V. Draw the bandwidth.
11. A non-periodic composite signal contains frequencies from 10 to 30 KHz. The
12. peak amplitude is 10 V for the lowest and the highest signals and is 30 V for the 20-KHz signal. Assuming that the amplitudes change gradually from the minimum to the maximum, draw the frequency spectrum.

Unit - 2

1. Distinguish between a signal element and a data element and data rate and signal rate.
2. Apply Manchester and differential Manchester scheme for the following data streams; 01010101 and 00110011
3. An NRZ-I signal has a data rate of 100 Kbps. Using Figure 4.6, calculate the value of the normalized energy (P) for frequencies at 0 Hz, 50 KHz, and 100 KHz.
4. Which characteristics of an analog signal are changed to represent the digital signal in each of the following digital-to-analog conversion?
    a. ASK
    b. FSK
    c. PSK
    d. QAM
5. Describe the goals of multiplexing and list three main multiplexing techniques
6. We need to transmit 100 digitized voice channels using a pass-band channel of 20 KHz. What should be the ratio of bits/Hz if we use no guard band?
7. We need to use synchronous TDM and combine 20 digital sources, each of 100 Kbps. Each output slot carries 1 bit from each digital source, but one extra bit is added to each frame for synchronization. Answer the following questions:
    a. What is the size of an output frame in bits?
    b. What is the output frame rate?

8. Distinguish between multilevel TDM, multiple slot TDM, and pulse-stuffed TDM.
9. Define FHSS and explain how it achieves bandwidth spreading.
10. How do guided media differ from unguided media? What are the three major classes of guided media?

Unit – 3

1. Differentiate between different switching networks - Circuit switched networks, Packet switch networks
2. Explain 3 types of Link layer addressing and ARP Packet switch networks: Datagram Networks Virtual circuit networks & structure of a switch.
3. Compare space-division and time-division switches.
4. Discuss the concept of redundancy in error detection and correction and
5. distinguish between forward error correction versus error correction by retransmission.
6. Find the minimum Hamming distance for the following cases:
   a. Detection of two en-ors.
   b. Correction of two errors.
   c. Detection of 3 errors or correction of 2 errors.
   d. Detection of 6 errors or correction of 2 errors.
7. Given the dataword 1010011110 and the divisor 10111,
   a. Show the generation of the codeword at the sender site (using binary division).
   b. Show the checking of the codeword at the receiver site (assume no error).
8. A sender needs to send the four data items Ox3456, OxABCC, Ox02BC, and OxEEEE. Answer the following:
   a. Find the checksum at the sender site.
   b. Find the checksum at the receiver site if there is no error.
9. Assuming even parity, find the parity bit for each of the following data units.
   a. 1001011
   b. 0001100
10. A sender has two data items to send: Ox4567 and OxBA98. What is the value of the checksum?

Unit – 4

1. Explain any one protocols for noiseless channels?
2. Define framing and explain different types of framing
3. Compare and contrast HDLC with PPP. Which one is byte-oriented; which one is bit-oriented?
4. We have a pure ALOHA network with 100 stations. If Tfr = 1 }.is, what is the number of frames/s each station can send to achieve the maximum efficiency.Repeat for slotted ALOHA.
5. List three categories of multiple access protocols
6. Compare and contrast a random access protocol with a controlled access protocol.
7. In a CSMA/CD network with a data rate of 10 Mbps, the minimum frame size is found to be 512 bits for the correct operation of the collision detection process. What should be the minimum frame size if we increase the data rate to 100 Mbps? To 1 Gbps? To 10 Gbps?
8. Explain with a flow chart CSMA/CA
9. Explain different Channelization Protocols
10. Explain different Controlled access Protocols

Unit –5

1. What is the purpose of an NIC
2. Differentiate between a unicast, multicast, and broadcast address
3. What is the hexadecimal equivalent of the following Ethernet address? 01011010 00010001 01010101 00011000 10101010 00001111

4. The address 43:7B:6C:DE: 10:00 has been shown as the source address in an Ethernet frame. The receiver has discarded the frame. Why?
5. Differentiate between a unicast, multicast, and broadcast address
6. What are the common Fast Ethernet implementations
7. Discuss the three types of mobility in a wireless LAN.
8. Match the layers in Bluetooth and the Internet model.
9. Create a system of three LANs with four bridges. The bridges (B 1 to B4) connect the LANs as follows:
   a. B1 connects LAN 1 and LAN 2.
   b. B2 connects LAN 1 and LAN 3.
   c. B3 connects LAN 2 and LAN 3.
   d. B4 connects LAN 1, LAN 2, and LAN 3.
   Choose B1 as the root bridge. Show the forwarding and blocking ports, after applying the spanning tree procedure.

Lesson Plan

Unit – 1

1. Different n/w components, data representation, data flow, performance reliability and security of a network and type of connections.
2. LAN,MAN,WAN, circuit switching and packet switching, internet accessing, standards and administration.
3. Principles of protocol standards and TCP/IP Protocol suit
4. TCP/IP Protocol suit contd. OSI Model and its lack of success
5. Analog data, signal, periodic and non periodic , sine wave, phase, wavelength
6. Time and frequency domain, composite signal, band width and Problems on analog signals
7. Digital data, signal, periodic and non periodic, bit rate and length, transmission of digital signal and problems on analog signals
8. Transmission impairments: attenuation, distortion and noise. Problems
9. Noiseless channel: Nyquist bit rate, Noisy channel: Shannon capacity, problems
10. Performance: bandwidth, throughput and latency
11. Problems on performance

Unit – 2

1. Define Line coding and their common characteristics and problems
2. Unipolar: NRZ, Polar: NRZ,RZ, Manchester and differential Manchester
3. Bipolar: AMI Pseudo stationary, multilevel and multitransition methods.
4. Block coding methods its use in error detection and correction - 4B/5B, 8B/10B.
5. Scrambling methods for long distance communication (B8ZS, HDB3). And digital data transmission - serial and parallel.
6. Pulse code and delta Modulation
7. Digital signal to analog signal: amplitude, frequency and phase shift keying.
8. Multiplexing: FDM , WDM and problems
9. TDM and problems and Multiplexing in Wireless communication: FHSS
10. DSSS and Twisted pair and optical cables
11. Radio waves, Microwaves and Infrared

Unit – 3

1. Switches and method of Switches . Three phases of Circuit switched networks,
2. Packet switch networks: Datagram Networks
3. Packet switch networks : Virtual circuit networks & structure of a switch.
4. 3 types of Link layer addressing and ARP

5. Single bit, burst error, redundancy, forward error correction and retransmission, Minimum Hamming distance for error detection
6. Parity check code and - Cyclic code- Cyclic Redundancy Code. Using polynomial form
7. Compute and detect error in a stream of bits - Cyclic code- Cyclic Redundancy Code. Using polynomial form
8. More problems on CRC
9. Compute internet checksum and problems and other approaches
10. Forward error correction using Hamming distance

Unit – 4

1. Framing and different types of framing , flow and error control and connectionless and connection oriented .
2. Simple, stop and wait protocol and piggybacking
3. Configuration, transfer modes and framing in HDLC
4. Services, framing and transition phases of POINT-TO-POINT protocol
5. Random access Protocols : ALOHA and problems
6. CSMA and CSMA/CD
7. CSMA/CA and Controlled access Protocols – Reservation
8. Controlled access Protocols: Polling and Token passing
9. Channelization Protocols –FDMA, TDMA, CDMA
10. Problems on CDMA

Unit – 5

1. Ethernet protocol, and evolution, characteristics, addressing of standard Ethernet
2. Access methods and efficiency and changes in the standards of standard Ethernet
3. Access method Fast Ethernet (100MBPS)
4. Physical layer of Fast Ethernet (100MBPS)
5. MAC sub layer and Physical layer of Gigabit Ethernet
6. Implementation of 10 Gigabit Ethernet and discussion and Architectural comparison, characteristics and access control of wired and wireless LAN's
7. The IEEE 802.11 Architecture, different services provided and access method
8. Addressing Mechanism and physical layer of IEEE 802.11
9. Understand Bluetooth Architecture
10. Hubs, Repeaters, Bridges, Switches, and Routers

Course Articulation Matrix (CAM)														
Course Outcome (CO)	**Program Outcome (ABET/NBA-(3a-m))**													
		a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Describe different network models and **calculate** the performance of the Network .	L1& L3	H	H	H	L	L	-	-	-	-	-	L	-	-
Select encoding scheme, multiplexing methods and suitable media for data transmission .	L4	H	H	H	L	M	-	-	-	-	-	-	-	-
Describe different switching circuits, link addressing and **implement** different error detection and correction methods for digital data.	L2& L6	H	H	H	M	L	-	-	-	-	-	-	-	-
Differentiate different data link protocols and **Select** suitable media access control protocol for data transmission	L2 & L4	L	L	M	M	M	-	-	-	-	-	-	-	-
Explain the architecture of wired and wireless Local Area Networks (LANs).	L2	-	-	M	M	-	-	-	-	-	-	-	-	-
L- Low, M- Moderate, H-High														
Illustration of CAM of Data Communication														

Course Assessment Matrix (CAM)														
Course Outcome (CO)		**Program Outcome (ABET/NBA-(3a-m))**												
		a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Describe different network models and **calculate** the performance of the Network .	L1& L3	3	3	3	1	1	-	-	-	-	-	1	-	-
Select encoding scheme, multiplexing methods and suitable media for data transmission .	L4	3	3	3	1	2	-	-	-	-	-	-	-	-
Describe different switching circuits, link addressing and **implement** different error detection and correction methods for digital data.	L2& L6	3	3	3	2	1	-	-	-	-	-	-	-	-
Differentiate different data link protocols and **Select** suitable media access control protocol for data transmission	L2 & L4	1	1	2	2	2	-	-	-	-	-	-	-	-
Explain the architecture of wired and wireless Local Area Networks (LANs).	L2	-	-	2	2	-	-	-	-	-	-	-	-	-
1- Low, 2- Moderate, 3-High														
Illustration of CAM of Data Communication														

Course Title : SYSTEM SOFTWARE			
Course Code: P13CS54	**Semester : V**	**L-T-P-H: 4- 0 – 0-4**	**Credits:4**
Contact Period : Lecture :52 Hrs., Exam: 3Hrs.		**Weightage :CIE:50% SEE:50%**	

Prerequisites :

- Assembly language programming
- Data Structure and algorithms
- Basics of computer organization

Course Objectives

This course aims to

1. Understand the basic machine architecture.
2. Design of various system software.
3. Know the working principle of assemblers, linkers and loaders.
4. Analyze the working principle of macro processor.
5. Understand the application of lex and yacc tools for developing lexical analyzers and parsers.

Course Content

Unit - 1

MACHINE ARCHITECTURE :

Introduction to System Software and Machine Architecture, Simplified Instructional Computer (SIC) - SIC Machine Architecture, SIC/XE Machine Architecture, SIC Programming Examples. CISC machines, RISC machines, comparison of CISC and RISC

:

9 hours

Unit-2

ASSEMBLERS :

Basic Assembler Function - A Simple SIC Assembler, Assembler Algorithm and Data Structures, Machine Dependent Assembler Features - Instruction Formats & Addressing Modes, Program Relocation.

Machine Independent Assembler Features - Literals, Symbol-Definition Statements, Expression, Program Blocks, Control Sections and Programming Linking, Assembler Design Operations - One- Pass assembler, Multi-Pass Assembler.

12 hours

Unit-3

LOADERS AND LINKERS :

Basic Loader Functions - Design of an Absolute - Relocation, Program Linking, Algorithm and Data Structures for a Linking Loader , Machine-Independent Loader Features - Automatic Library Search, Loader Options, Loader Design Options - Linkage Editor, Dynamic Linking

10 hours

Unit - 4

MACROPROCESSOR-:

Basic Macro Processor Functions - Macro Definitions 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 processor Design Options – Recursive Macro Expansion, General-Purpose Macro Processors, Macro Processing Within Language Translators, Implementation Examples - MASM Macro Processor, ANSI C Macro Processor.

11 hours

Unit 5

LEX :

Recognizing Words With LEX, Symbol Tables, Grammars, Parser-Lexer Communication, The Parts of Speech Lexer, A YACC Parser, The Rules Section, Running LEX and YACC, LEX and Hand- Written Lexers, Examples of Regular Expressions, A Word Counting Program, Parsing a Command Line.

YACC : Using YACC - Grammars, Recursive Rules, Shift/Reduce Parsing, What YACC Cannot Parse; A YACC Parser - The Definition Section, The Rules Section, Symbol Values and Actions, The LEXER, Compiling and Running a Simple Parser, Arithmetic Expressions and Ambiguity, Variables and Typed Token

10 hours

Text Books :

1. System Software - an introduction to system programming Leland. L. Beck,D.Manjula 3rd Edition, reprinted in 2013.
2. John. R. Levine, Mason and Doug Brown:Lex and Yacc, O'Reilly, SPD, Reprint March 2005. chapters 1,2(page 27-42),3(page 51-65)

REFERENCE BOOK:

1. **System Programming and Operating Systems,** D M Dhamdhere, TATA McGraw Hill, 2nd Edition.

Course Outcome

On successful completion of the course the students will be able to

1. Implement the application using SIC or SIC/XE machine.(L1,L6)
2. Generate the machine code for any set of instruction.(L5,L6)
3. Analyze the working principles of linkers and loaders.(L4)
4. Analyze the working principles of Macro processor.(L4)
5. Describe the tools used during lexical and syntax analysis phase of compiler.(L2,L3)

Topic Learning Objectives

After learning all the topics of unit – I, the student is able to

1. Define and explain the components of system software.(L1,L2)
2. Distinguish between system software and application software.(L2)
3. Compare between SIC machine architecture & SIC/XE machine architecture(L4)
4. Illustrate how input and output operations are performed in SIC and SIC/XE(L4)
5. Describe how to perform looping and indexing operations of SIC/XE.(L1)
6. Explain the architectures of various CISC & RISC machines(L2)
7. Differentiate between RISC & CISC machines. (L2)
8. Determine the target address for the instruction given.(L3)
9. Describe how to perform data movement operations of SIC and SIC/XE machine(L1)

Write an SIC or SIC/XE program based on the application given(L5)

After learning all the topics of unit – II, the student is able to

1. List basic functions of assemblers.(L1)
2. Discuss the data structures used by assemblers.(L2)
3. Define machine dependent assembler features.(L1)
4. List the advantages and disadvantages of assembly language.(L1,L2)
5. Explain the structure of various records used in object program.(L2)
6. Apply various assembler directives in programs(L3)
7. Compare two pass assembler with single pass assemblers (L4)
8. Describe the various machine independent assembler features.(L1)
9. Justify when we use one pass assembler and why.(L6)
10. Compare between program blocks and control sections. (L4)

After learning all the topics of unit – III, the student is able to

1. Explain the concept of linkers and loaders.(L2)
2. List and explain the task of a linker.(L1,L2)
3. Explain how relocation is done using bit masking and modification record.(L2)
4. reproduce an algorithm for boot strap loader.(L1)
5. Explain how loading and calling a subroutine can be done using dynamic linking.(L2)
6. Explain the procedure of performing program linking when sub programs use external references. (L2)
7. Justify how ESTAB data structure is used in 2 pass linking loader.(L6)
8. Reproduce an algorithm for absolute loader.(L1)
9. Determining the object code after performing linking operation.(L4)
10. Distinguish between linking loader and linkage editor.(L2)

After learning all the topics of unit – IV, the student is able to

1. Explain the data structures used in macro processor.(L2)
2. Explain machine independent macro processor features.(L2)
3. Discuss the problems associated in implementing macros calls within macros.(L6)
4. Explain the pros and cons of general purpose macro processor.(L2)
5. Analyze various design options of macro processor. (L4)
6. Justify why one pass macro processor preferred instead of two pass macro processor.(L6)
7. Explain the basic functions of macro processor (L2)
8. Decide how macro processor and micro processor varies (L6)
9. Analyze the various procedures used in one pass macro processor (L4)
10. Justify how recursive macro expansion is resolved in one pass macro processor (L6)

After learning all the topics of unit – V, the student is able to

1. Identify fundamentals of using LEX.(L1)
2. Explain the special characters to describe regular expression in LEX.(L2)
3. Describe special purpose routines used by LEX generated state machines.(L1)
4. Create a simple application using LEX such as word count in a file, computing statistics on C programs.(L5)
5. Conclude how LEX is used to create Compilers and Interpreters.(L6)
6. Determine what yacc cannot parse (L4)
7. Explanations about how to build shift reduce parser.(L2)
8. Apply yacc in applications (L3)
9. Create a simple application using YACC to count no-of if statements in nested-if loop (L5)
10. Difference between lex and yacc structure (L2)

Review Questions

Unit 1

1.Differentiate between system software and application software.
2. Define system software. list few examples of system software
3. Explain the architecture of SIC machine
4.Explain the architecture of SIC/XE machine
5.Determine the TA for SIC or SIC/XE machine instruction
6.Write a SIC/XE program to move a string "INFORMATION " from one location to another location
7.Differentiate between RISC or CISC machines.
8.Explain any one architecture of CISC machines
9.Explain any one architecture of RISC machines
10.Define why SIC/XE machine is said to be upward compatible

Unit 2

1. Explain any ten assembler directives with example.
2. Differentiate between literals and constants.
3. What is the information passed from pass1 to pass 2 and why?
4. Write the functions of symbol table and opcode table w.r.t. pass1 and pass2 algorithms
5. Explain the data structures and variables used in 2pass assembler
6. Differentiate between program blocks and control section
7. Explain one pass assembler with examples
8. Write the difference between absolute expression and relative expression
9. Write the format for object program.
10. Explain multi pass assembler

Unit 3

1 Discuss simple bootstrap loader with algorithm.
2. List and explain the different function utilities of loader.
3. Discuss automatic library search with reference to linker and loader
4. Differentiate between linking loader and linkage editor
5. Explain pass1 & pass2 algorithms of linking loader
6. Explain the data structures and variables used in linking loader
7. Explain how the programs are been linked with examples
8. Explain absolute loader.
9. Explain dynamic linking with example
10. List the advantages of dynamic linking

Unit 4

1. Define macro processor
2. Explain the basic functions of macro processor
3.Explain the data structures used in macro processor
4 list and explain the disadvantage of 2 pass macro processor
5. Explain the various procedures used in one pass macro processor.
6. Explain any two machine independent macro processor features.
7. Explain recursive macro expansion.
8. Explain the procedures of DEFINE and EXPAND in one pass macro processor
9. List and explain the various macro processor design options.
10. Explain the advantages and disadvantages of general purpose macro processor

Unit 5

1. Explain the structure of a lex program
2. Define regular expression. Explain all the characters used in regular expression
3. Define shift reduce parsing? Explain the parsing of the input "fred=12+13" represent it using parse tree
4. Write a yacc program to evaluate the expression. Consider all possible cases
5. Write a lex program to count number of vowels and consonants in a given string
6. Explain parser lexer communication
7. Explain the structure of the yacc program
8. Write a LEX specification to count the number of positive, negative integer and positive, negative real number in a given input.
9. Compare and contrast LEX and YACC tools with examples.
10. Define : (i) LEXER (ii) Parser
(iii) Token (iv) yylex()
With reference to LEX and YACC tools

Lesson Plan

Unit –1

1. **MACHINE ARCHITECTURE** Introduction to System Software and Machine Architecture
2. Simplified Instructional Computer (SIC) Machine Architecture,
3. SIC/XE Machine Architecture
4. SIC Programming Examples.
6. 5.SIC/XE programming examples
7. 6.CISC machines architecture
8. Continuation of CISC machine architecture
9. RISC machines architecture
10. 9. Comparison of CISC and RISC and comparison between SIC and SIC/XE machine

Unit – 2

1. ASSEMBLERS: Basic Assembler Function
2. A Simple SIC Assembler, Data Structures
3. Pass1 algorithm
4. Pass2 algorithm
5. Machine Dependent Assembler Features - Instruction Formats & Addressing Modes,
6. Program Relocation.
7. Machine Independent Assembler Features - Literals,Symbol-Definition Statements
8. Various types of expressions
9. Program blocks with examples
10. Control section with examples
11. One pass assembler
12. Multi pass assemblers

Unit – 3

1. 1 . **LOADERS AND LINKERS :** Basic Loader Functions
2. Design of an Absolute
3. Bootstrap algorithm with explanation
4. Program Relocation with example
5. Program Linking
6. Data Structures for a Linking Loader with pass1 algorithm
7. pass2 algorithm
8. Machine-Independent Loader Features - Automatic Library Search, Loader Options
9. Loader Design Options – linking loader and Linkage Editor
10. Dynamic Linking with example

Unit – 4

1. MACRO PROCESSOR-:Basic Macro Processor Functions - Macro Definitions and Expansion
2. Data structures and variables used in one pass algorithm
3. Macro Processor Algorithm
4. Machine- Independent Macro Processor Features - Concatenation of Macro Parameters, Generation of Unique Labels
5. Conditional Macro Expansion
6. Keyword Macro Parameters
7. Macro processor Design Options – Recursive Macro Expansion
8. General-Purpose Macro Processors with its advantages and disadvantages
9. Macro Processing Within Language Translators
10. Implementation Examples - MASM Macro Processor
11. Implementation Examples continuation - ANSI C Macro Processor.

Unit – 5

1. **LEX :**Recognizing Words With LEX
2. Symbol Tables, Grammars, Parser-Lexer Communication
3. The Parts of Speech Lexer, A YACC Parser, The Rules Section
4. Running LEX and YACC, LEX and Hand- Written Lexers
5. Examples of Regular Expressions, A Word Counting Program, Parsing a Command Line.
6. **YACC :** Using YACC - Grammars, Recursive Rules
7. Shift/Reduce Parsing, What YACC Cannot Parse
   A YACC Parser - The Definition Section, The Rules Section, Symbol Values and Actions
8. The LEXER, Compiling and Running a Simple Parser
9. Arithmetic Expressions and Ambiguity, Variables and Typed Token

A. Course Articulation Matrix (CAM)														
Course Outcome (CO)		**Program Outcome (ABET/NBA-(3a-k))**												
		a	**b**	**C**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Implement the application using SIC or SIC/XE machine	**L1 L6**	L	H	H		M				H				*M*
Generate the machine code for any set of instruction	**L5 L6**	L	H	H	H	H				H		M	M	H
Analyze the working principles of linkers and loaders	**L4**	L	M	L						M			*M*	*L*
Analyze the working principles of Macro processor	**L4**	L	M	L						M			*M*	*L*
Describe the tools used during lexical and syntax analysis phase of compiler	**L2 L3**		H	H		M				M		H		
L- Low, M- Moderate, H-High														

Course Outcome (CO)		**Program Outcome (ABET/NBA-(3a-k))**												
		a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**K**	**l**	**m**
Implement the application using SIC or SIC/XE machine	**L1 L6**	**1**	**3**	**3**		**2**				**3**				***2***
Generate the machine code for any set of instruction	**L5 L6**	**1**	**3**	**3**	**3**	**3**				**3**		**2**	**2**	**3**
Analyze the working principles of linkers and loaders	**L4**	**1**	**2**	**1**						**2**			***2***	***1***
Analyze the working principles of Macro processor	**L4**	**1**	**2**	**1**						**2**			***2***	***1***
Describe the tools used during lexical and syntax analysis phase of compiler	**L2 L3**		**3**	**3**		**2**				**2**		**3**		
1- Low, 2- Moderate, 3-High														

Course Title: COMPUTER GRAPHICS AND VISUALIZATION			
Course Code: P13CS55	**Sem : V**	**L:T:P:H- 4:0:0:4**	**Credits:4**
Contact Period : Lecture :52 Hrs., Exam: 3Hrs.		**Weightage :CIE:50% SEE:50%**	

Prerequisites

- Basic familiarity with fundamental algorithms and data structures.
- Good programming skills.
- Basics of linear algebra and geometry.

Course learning objectives:

Students will be able to,

- Learn the basics of Application Programming Interface (API) implementation based on graphics pipeline approach.
- Apply mathematical transformations and vector techniques in the production of computer graphics.
- Gain familiarity of line drawing, clipping algorithms and rasterization techniques and interaction with input devices.
- Understand viewing, lighting and shading techniques.
- Design and create graphics applications using OpenGL.

Course Content

Unit – 1

INTRODUCTION: Applications of computer graphics, A graphics system, Images: Physical and synthetic, Imaging Systems, The synthetic camera model, The programmer's interface, Graphics architectures, Programmable Pipelines, Performance Characteristics, Graphics Programming: The OpenGL: The OpenGL API, Primitives and attributes, Color, Viewing, Control functions, The Gasket program, Polygons and recursion, The three-dimensional gasket. 10 Hours

Unit –2

GEOMETRIC TRANSFORMATIONS: Basic Two-Dimensional Geometric Transformations, Matrix Representations and Homogeneous Coordinates, Inverse Transformations, Two-Dimensional Composite Transformations, Other Two-dimensional Transformations, Raster Methods for Geometric Transformations, OpenGL Raster Transformations, Transformations Between Two Dimensional Coordinate Systems, Geometric transformations in Three Dimensional Space, Three Dimensional Translation, Three Dimensional Rotation, Three Dimensional Scaling, Composite Three Dimensional Transformations, Other Three Dimensional Transformations, Transformations Between Three Dimensional Coordinate Systems, Affine Transformations, OpenGL Geometric Transformation Functions.

(Self-Study Component : Scalars, vectors and points and Frames in OpenGL. refer Book 1 Chapter 4: 4.1, 4.4) 11 Hours

Unit – 3

IMPLEMENTATION: Coordinate Reference Frames, Line Drawing Algorithms, Circle Generating Algorithms, Fill-Area Primitives, Polygon Fill Areas, OpenGL Polygon Fill Area Functions, Clipping Algorithms, Two-Dimensional Point Clipping, Two Dimensional Line Clipping: Cohen Sutherland Line Clipping, Liang Barsky Line Clipping, Polygon Fill Area Clipping: Sutherland –Hodgeman Polygon Clipping.

INPUT AND INTERACTION: Interaction, Input devices, Clients and Servers, Display Lists, Display Lists and Modeling, Programming Event Driven Input, Menus, Picking, ,

Building Interactive Models, Animating Interactive Programs, Design of Interactive Programs, Logic Operations. 11 Hours

Unit – 4

VIEWING : Classical and computer viewing, Viewing with a Computer, Positioning of the camera, Simple projections, Projections in OpenGL, Hidden-surface removal, Interactive Mesh Displays, Parallel-projection matrices, Perspective-projection matrices, Projections and Shadows. 10 Hours

Unit – 5

LIGHTING AND SHADING: Light and matter, Light sources, The Phong lighting model, Computation of vectors, Polygonal shading, Approximation of a sphere by recursive subdivisions, Light sources in OpenGL, Specification of materials in OpenGL. Shading of the sphere model, Global illumination.

CURVES AND SURFACES: Representation of Curves and surfaces, Hermite Curves and Surfaces, Bezier curves and Surfaces, Cubic B-Splines, The Utah Teapot. 10 Hours

TEXT BOOK:

1. Interactive Computer Graphics A Top-Down Approach with OpenGL, Edward Angel 6th Edition, Addison Wesley.
2. Computer Graphics, OpenGL Version – Donald Hearn and Pauline Baker, 4th Edition, Pearson publications

REFERENCE BOOK:

1. F.S. Hill,Jr, and M. Kelley, Jr. "Computer Graphics Using OpenGL", Pearson/PHI, 3rd Edition, 2009.
2. James D Foley, Andries Van Dam, Steven K Feiner, John F Hughes, "Computer Graphics", Addison-Wesley.

Course outcomes

Upon completion of this course students should be able to,

1. Explain the basics of Application Programming Interface (API) implementation based on graphics pipeline approach.
2. Develop mathematical transformations and vector techniques in the production of computer graphics.
3. Analyze line drawing, clipping algorithms and rasterization techniques and interaction with input devices.
4. Identify and apply different types of viewing, lighting, shading and projections to create 2D or 3D images in OpenGL.
5. Design and create graphics applications in 2D or 3D using OpenGL.

Topic Learning Objectives

Unit 1

Topic: Introduction

Learning Objectives:

1. Discuss the applications of computer graphics(L2).
2. Describe how image is formed using imaging techniques (L2).
3. Explain the synthetic camera model(L3).
4. Describe the three different architectures of graphics system(L3).
5. Construct a simple Sierpinski gasket (L6).
6. Create 2d applications in OpenGL (L6).
7. Explain and apply different primitives and attributes to application programs (L3).
8. Distinguish between RGB color model and Indexed color model (L2).
9. Recognize the viewing and control functions in OpenGL programs (L1).
10. Develop an 3d gasket (L6).

Unit 2

Topic: Geometric transformations

Learning Objectives:

1. Define scalars, points and vectors and basic operations on them (L1).
2. Explain rotation, translation and scaling (L3).
3. Apply the transformation to an homogeneous coordinate system (L3).
4. Describe the different methods of transformation (L2).
5. Write a program using transformation matrices (L6).
6. Compare reflection and shear transformation.(L4)
7. Explain the alternative method for describing and manipulating rotations (L4).
8. Design a procedure and code to translate, rotate and scale objects in 2D or 3D system.(L6)
9. Apply inverse transformations.(L3)
10. Combine all basic transformation to create an application in computer graphics.(L6)

Unit 3

Topic: Implementation & input and interaction

Learning Objectives:

1. Explain the basic algorithms that are used to implement the rendering pipeline used by OpenGL (L2&L4).
2. Define the need of Clipping, Rasterization and hidden surface removal (L1).
3. Describe the line clipping algorithms (L3).
4. Apply the polygon clipping on various geometrical primitives (L3).
5. Design rasterization using Bresenham's algorithm (L6).
6. Understand the interaction with the window system (L1)..
7. Define display list and its architecture (L1).
8. Create menus and to use a picking technique in programs(L6).
9. Write a simple CAD program (L6).
10. Write and build interactive programs and models (L6)

Unit 4

Topic: Viewing

Learning Objectives:

1. Discuss the relationship between the classical viewing techniques and computer viewing (L2).
2. Know how viewing is implemented in computer graphics.(L1&L4)
3. Demonstrate how to position the camera in computer graphics using OpenGL functions.(L3)
4. Know how projection is implemented using simple projections (L2).
5. Explain the three types of views (L1).
6. Analyze hidden surface removal algorithm.(L4)
7. Describe the positioning and orientation of the camera (L2).
8. Classify the different type projections to an object using OpenGL viewing functions (L2).
9. Implement meshes in various graphics applications.(L5)
10. Explain how shadows are created in computer graphics.(L1)

Unit 5

Topic: Lighting and shading, curves and surfaces

Learning Objectives:

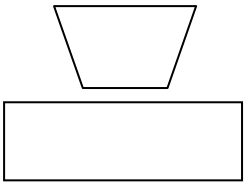
1. Define light and matter and interaction between light and material (L1).
2. Describe and use the different type of light sources for rendering most simple scenes (L2).
3. Explain the Phong lighting model which is basis for lighting and shading in graphics API's (L2).
4. Describe how to apply different shading techniques on a polygonal mesh (L2).

5. Discuss a powerful technique for generating approximation to curves and surfaces to any desired level of accuracy (L2).
6. Identify different material properties for the front and back faces of a surface (L2).
7. Write the program to shade the sphere with different shading techniques (L6).
8. Express how curves and surfaces are represented.(L2)
9. Design Hermite and Bezier curves.(L6)
10. Develop a program to create Utah Teapot.(L6)

Review Questions

Unit – 1

1. Briefly explain the four major areas for the applications of computer graphics.
2. Describe the concept of pinhole camera. Derive the expressions for projection of a point(x, y, z) and angle of view. Also indicate the advantages and disadvantages of pinhole camera.
3. List the major graphics architecture. Discuss any one architecture with a neat diagram.
4. Discuss the various components of a graphics system with the help of a neat diagram.
5. Write a fragment of a simple code in pen plotter model that would generate the output shown in fig a. (assume your own coordinates)



6. Explain four types of camera specification with a neat diagram.
7. Identify major groups of graphics functions and explain.
8. Describe the two major color models with color related API's.
9. Write a complete OpenGL program for creating 3D Sierpinski gasket by subdivision of tetrahedron.
10. Identify the different control functions in OpenGL.

Unit – 2

1. What is transformation? Describe basic two dimensional transformations.
2. Define Reflection and Shear.
3. Give the OpenGL functions for raster transformations.
4. Interpret how quaternion's are applied in creation of images in computer graphics.
5. Discuss affine transformations.
6. Write a OpenGL code to implement the basic 2D transformations.
7. Describe composite three dimensional transformations.
8. Show with a neat diagram the sequence of transformations for rotating an object about an axis that is parallel to the x-axis.
9. Explain two dimensional rigid body transformation.
10. Develop a procedure to translate, rotate and scale a triangular object in two dimensional coordinate system.

Unit – 3

1. Discuss the Bresenham's rasterization algorithm. How it is advantageous when compared to other existing methods? Describe.
2. Explain Cohen Sutherland line clipping algorithm in detail.
3. Implement Liang barsky line Clipping Algorithm with an example.

4. Discuss the Sutherland –Hodgman Polygon clipping algorithm in detail.
5. Demonstrate the scan line area filling algorithm with example.
6. What are major characteristics that describe the logical behavior of an input device?
7. List and explain various classes of logical input devices.
8. What is double buffering? How it is implemented in OpenGL.
9. Describe how an even driven input can be programmed for mouse and window events.
10. Explain the picking operation in OpenGL with an example.

Unit – 4

1. What are simple projections?
2. Derive a 4x4 matrix representing perspective projections.
3. How shadows can be created and projected in OpenGL. Explain.
4. Discuss the classical viewing with a neat diagram.
5. Derive view orientation matrix using rotation and translation in homogeneous co-ordinates.
6. Define hidden surface removal.
7. Illustrate the use of gluLookAt() function with an example.
8. Differentiate between object space algorithm and image space algorithm
9. What is culling? Give the function for culling.
10. Demonstrate the use of meshes in computer graphics with example.

Unit – 5

1. Describe different light sources.
2. Discuss phong lighting model .
3. List different method for shading a polygon . Explain any one.
4. With syntax write the functions used to specify the different material properties for front and back face of a surface.
5. Write a note on flat and gourmand shading.
6. How to represent the curves and surfaces by the explicit representation method.
7. Define Hermite curves?
8. With equations explain how to construct Bezier curves
9. Write a OpenGL program to render a teapot.
10. Discuss cubic B-spline curves.

Lesson Plan

Unit – 1

1. **INTRODUCTION:** Applications of computer graphics, , Imaging Systems, The synthetic camera model
2. A graphics system, Images: Physical and synthetic
3. The programmer's interface, Graphics architectures
4. Graphics Programming, The OpenGL: The OpenGL API
5. The Gasket program.
6. Programmable Pipelines, Performance Characteristics
7. Primitives and attributes.
8. Color, Viewing.
9. Control functions, Polygons and recursion.
10. The three-dimensional gasket.

Unit – 2

1. **GEOMETRIC TRANSFORMATIONS:** Scalars, vectors and points and Frames in OpenGL
2. Basic Two-Dimensional Geometric Transformations
3. Matrix Representations and Homogeneous Coordinates
4. Transformations, Two-Dimensional Composite Transformations
5. Other Two-dimensional Transformations, Raster Methods for Geometric Transformations
6. OpenGL Raster Transformations
7. Transformations Between Two Dimensional Coordinate Systems
8. Geometric transformations in Three Dimensional Space, Three Dimensional Translation, Three Dimensional Rotation, Three Dimensional Scaling,
9. Composite Three Dimensional Transformations, Other Three Dimensional Transformations
10. Transformations Between Three Dimensional Coordinate Systems,
11. Affine Transformations, OpenGL Geometric Transformation Functions.

Unit – 3

1. **IMPLEMENTATION:** Coordinate Reference Frames, Line Drawing Clipping Algorithms
2. Circle Generating Algorithms, Fill-Area Primitives
3. Polygon Fill Areas, OpenGL Polygon Fill Area Functions
4. Two-Dimensional Point Clipping, Two Dimensional Line Clipping
5. Cohen Sutherland Line Clipping, Liang Barsky Line Clipping
6. Polygon Fill Area Clipping: Sutherland –Hodgman Polygon Clipping
7. **INPUT AND INTERACTION:** Interaction, Input devices, Clients and Servers, Display Lists
8. Display Lists and Modelling, Programming Event Driven Input
9. Menus, Picking,
10. Building Interactive Models, Animating Interactive Programs
11. Design of Interactive Programs, Logic Operations

Unit – 4

1. **VIEWING :** Classical and computer viewing
2. Viewing with a Computer, Positioning of the camera
3. Simple projections
4. Projections in OpenGL
5. Hidden-surface removal,
6. Interactive Mesh Displays,
7. Parallel-projection matrices
8. Perspective-projection matrices
9. Perspective-projection matrices
10. Projections and Shadows

Unit – 5

1. **LIGHTING AND SHADING:** Light and matter, Light
2. The Phong lighting model, Computation of vector sources
3. Polygonal shading, Approximation of a sphere by recursive subdivisions
4. Light sources in OpenGL
5. Specification of materials in OpenGL
6. Shading of the sphere model, Global illumination
7. **CURVES AND SURFACES:** Representation of Curves and surfaces,
8. Hermite Curves and Surfaces, Bezier curves and Surfaces
9. Cubic B-Splines
10. The Utah Teapot

B. Course Articulation Matrix (CAM)														
Course Outcome (CO)				**Program Outcome (ABET/NBA-(3a-k))**										
		a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Explain the basics of Application Programming Interface (API) implementation based on graphics pipeline approach.	**L2**	M	H	-	-	M	-	-	-	-	-	M	L	L
Develop mathematical transformations and vector techniques in the production of computer graphics.	**L6**	H	**-**	M	-	M	-	-	-	-	-	M	L	M
Analyze line drawing, clipping algorithms and rasterization techniques and interaction with input devices.	**L4, L1**	M	H	M	-	M	-	-	-	-	-	M	M	M
Identify and apply different types of viewing, lighting, shading and projections to create 2D or 3D images in OpenGL	**L1, L4**	H	M	L	-	H	-	-	-	-	-	M	L	M
Design and create graphics applications in 2D or 3D using OpenGL	**L6**	H	M	M	M	M	-	-	-	-	-	H	M	H
1 – Low, 2 – Moderate and 3 – High														

C. Course Assessment Matrix (CaM)														
Course Outcome (CO)				**Program Outcome (ABET/NBA-(3a-k))**										
		a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Explain the basics of Application Programming Interface (API) implementation based on graphics pipeline approach.	**L2**	2	3	-	-	2	-	-	-	-	-	2	1	1
Develop mathematical transformations and vector techniques in the production of computer graphics.	**L6**	3	**-**	2	-	2	-	-	-	-	-	2	1	2
Analyze line drawing, clipping algorithms and rasterization techniques and interaction with input devices.	**L4, L1**	2	3	2	-	2	-	-	-	-	-	2	2	2
Identify and apply different types of viewing, lighting, shading and projections to create 2D or 3D images in OpenGL	**L1, L4**	3	2	1	-	3	-	-	-	-	-	2	1	2
Design and create graphics applications in 2D or 3D using OpenGL	**L6**	3	2	2	2	2	-	-	-	-	-	3	2	3
1 – Low, 2 – Moderate and 3 – High														

Course Title : Management, Entrepreneurship and IPR			
Course Code: P13CS56	**Semester : V**	**L-T-P-H: 2 -2- 0-4**	**Credits: 3**
Contact Period : Lecture :52 Hrs., Exam: 3Hrs.		**Weightage :CIE:50% SEE:50%**	

Prerequisites

- Knowledge of English language

Course Learning Objectives

The students should be able to

1. **Understand** the nature , importance of management and functions of a manager
2. **Explain** the process of planning and gain familiarity about the different organization structures.
3. **Explain** the requirements of direction and supervision and **explain** the methods of establishing control.
4. **Identify** the role of entrepreneurs in the economic development of the nation and recognize the barriers of entrepreneurship.
5. **Understand** the importance of Intellectual property protection.

Course content

Unit –1

Management : importance of management, definition, management functions, roles of a manager, levels of management ,managerial skills, management and administration , management –a science or art ,management – A profession , professional management v/s family management.

Development of management thought: Early classical approaches, Neo classical approaches, modern approaches. 10 Hours

Unit – 2

Planning : Nature, Importance of planning, forms ,types of plans , steps in planning , limitations of planning, making planning effective , planning skills, strategic planning in Indian industry.

Organisation Meaning, process of organizing, span of management, principles of organizing, Departmentation, organization structure, committees, teams 10 Hours

Unit – 3

Direction and supervision: Requirements of effective direction, giving orders, motivation, job satisfaction, morale, organizational commitment, first level supervision or front line supervision.

Controlling: Meaning and steps in controlling, Essentials of a sound control system, Methods of establishing control. 10 Hours

Unit – 4

Entrepreneurship: Meaning of Entrepreneur; Evolution of the Concept, Functions of an Entrepreneur, Types of Entrepreneur, Entrepreneur – an emerging Class. Concept of Entrepreneurship – Evolution of Entrepreneurship, Development of Entrepreneurship, Stages in entrepreneurial process; Role of entrepreneurs in Economic Development; Entrepreneurship in India; Entrepreneurship – its Barriers. 11 Hours

Unit – 5

Intellectual Property Rights: Introduction to IPR, origin and concepts of IPR, Concept of property, Forms of IP protection: Patents, copyrights, trademarks, designs, Trade secrets, Traditional knowledge, Geographical indications. Basic concepts and historical background of patent system and law- National and international scenario (American & European Patent Regimes). International Treaties/Conventions on IPR: Paris Convention, Berne convention, Madrid agreement, Rome convention, World Intellectual Property Organization (WIPO), World Trade Organization, TRIPS Agreement, Patent Co-operation Treaty. 11 Hours

Text Books:

1. **Management and Entrepreneurship ,** N V R Naidu ,T Krishna Rao 4th reprint.
2. **Intellectual Property Law Handbook**, B.L.Wadhera, Universal Law Publishing, 2002

Reference Books:

1 **Principles of Management**, P C Tripathi, P N Reddy, 5th edition, TataMcGraw Hill, 2012

2 **Dynamics of Entrepreneurial Development & Management**, Vasant Desai, Himalaya publishing house, 2007

Course Outcome

Upon completion of this course, students will be able to

1. **Describe** the importance of management and functions of a manager.
2. **Explain** the process of planning and principles of organizing.
3. **Identify** the role of entrepreneurs in the economic development of the nation.
4. **Explain** the requirements of direction and supervision and **Explain** the methods of establishing control.
5. **List** and **Explain** the different forms of Intellectual property protection.

Topic Learning Objectives

Unit 1

Topic: Management**,** importance of management, definition, management functions, roles of a manager, levels of management ,managerial skills, management and administration , management –a science or art ,management – A profession , professional management v/s family management.

Development of management thought Early classical approaches, Neo classical approaches, modern approaches.

Learning Objective:

1. **Define** management **L1**
2. **Describe** the roles of a manager **L2**
3. **List** the various levels of management **L3**
4. **Describe** the advantages of human relations movement **L2**
5. **Explain** the principles of administrative management **L2**

Unit 2

Topic: Planning**,** Nature, Importance of planning, forms ,types of plans , steps in planning , limitations of planning, making planning effective , planning skills, strategic planning in Indian industry.

Organization Meaning, process of organizing, span of management , principles of organizing, Departmentation, organization structure, committees, teams

Learning Objective :

1. **Give** suggestions to make committees effective **L4**
2. **Differentiate** between strategic planning and tactical planning **L1**
3. **Explain** the principles of organizing **L2**
4. **Define** span of management **L1**
5. **Explain** the importance of planning **L2**

Unit 3

Topic: Direction and supervision**,** Requirements of effective direction, giving orders, motivation , job satisfaction, morale ,organizational commitment, first level supervision or front line supervision.

Controlling, Meaning and steps in controlling, Essential of a sound control system, Methods of establishing control

Learning Objective:

1. **Explain** the requirements of effective direction **L2**
2. **Explain** the importance of job satisfaction **L2**
3. **Describe** the methods of establishing control **L3**
4. **Differentiate** between direction and supervision **L4**
5. **Compare** the different motivation theories **L5**

Unit 4

Topic: Entrepreneurship, Meaning of Entrepreneur; Evolution of the Concept, Functions of an Entrepreneur, Types of Entrepreneur, Entrepreneur – an emerging Class. Concept of Entrepreneurship – Evolution of Entrepreneurship, Development of Entrepreneurship, Stages in entrepreneurial process; Role of entrepreneurs in Economic Development; Entrepreneurship in India; Entrepreneurship – its Barriers

Learning Objective:

1. **Differentiate** between Intrapreneur and Entrepreneur **L4**
2. **Explain** the stages involved in Entrepreneurial process **L2**
3. **Categorize** the different types of Entrepreneurs **L4**
4. **Explain** the barriers of entrepreneurship **L2**
5. **Explain** the role of entrepreneurs in the economic development of a nation. **L2**

Unit 5

Topic: Intellectual Property Rights**,** Introduction to IPR, origin and concepts of IPR, Concept of property, Forms of IP protection: Patents, copyrights, trademarks, designs, Trade secrets, Traditional knowledge, Geographical indications. Basic concepts and historical background of patent system and law- National and international scenario (American & European Patent Regimes). International Treaties/Conventions on IPR: Paris Convention, Berne convention, Madrid agreement, Rome convention, World Intellectual Property Organization (WIPO), World Trade Organization, TRIPS Agreement, Patent Co-operation Treaty

Learning Objective:

1. **Describe** the role of WTO in IPR **L1**
2. **Differentiate** between patents and copyrights **L4**
3. **Give example** for trademarks **L2**
4. **Explain** the patent cooperation treaty **L2**
5. **List** the forms of IP protection **L3**

Review Questions

Unit - 1

1. **Define** management
2. **Explain** the importance of management
3. **Describe** the roles of a manager
4. **Differentiate** between management and administration
5. **List** the various levels of management
6. **Describe** the skills of a manager
7. **Explain** the principles of administrative management
8. **Explain** bureaucratic management.
9. **Describe** the advantages of human relations movement.
10. **Explain** the systems approach

Unit – 2

1. **List** the types of plans
2. **Explain** the importance of planning
3. **Describe** the stages of planning
4. **Differentiate** between strategic planning and tactical planning
5. **Explain** the limitations of planning
6. **Explain** the principles of organizing
7. **Describe** the various types of departmentation

8. **Define** span of management
9. **Differentiate** between the different organization structure
10. **Give** suggestions to make committees effective

Unit – 3

1. **Explain** the requirements of effective direction
2. **Compare** the different motivation theories
3. **Explain** the importance of job satisfaction
4. **Describe** the process of giving orders
5. **Differentiate** between direction and supervision
6. **Explain** the importance of organizational commitment
7. **Explain** the first level supervision
8. **Describe** the steps in control process
9. **Explain** the essentials of sound control system
10. **Describe** the methods of establishing control

Unit – 4

1. **Define** Entrepreneurship
2. **Differentiate** between Intrapreneur and Entrepreneur
3. **Explain** the stages involved in Entrepreneurial process.
4. **Describe** the characteristics of an Entrepreneur.
5. **Explain** the role of entrepreneurs in the economic development of a nation.
6. **Categorize** the different types of Entrepreneurs
7. **Describe** the functions of an Entrepreneur
8. **Explain** entrepreneurship in India
9. **Explain** the barriers of entrepreneurship
10. **Describe** the evolution of entrepreneurship

Unit - 5

1. **Explain** the importance of intellectual property rights
2. **List** the forms of IP protection
3. **Explain** the historical background of patent system and law in national and international scenario
4. **Differentiate** between patents and copyrights.
5. **Give example** for trademarks
6. **Describe** the various conventions on IPR
7. **Explain** the World Intellectual Property Organization (WIPO)
8. **Explain** the TRIPS Agreement
9. **Describe** the role of WTO in IPR
10. **Explain** the patent cooperation treaty

Lesson Plan

Unit – 1

1. Importance of management, definition
2. Management functions, roles of a manager,
3. Levels of management
3. Managerial skills, management and administration
4. Management –a science or art
5. Management – A profession
6. Professional management v/s family management.
7. Early classical approaches,
8. Neo classical approaches
9. Modern approaches

Unit – 2

1. Nature, Importance of planning
2. Forms ,types of plans

3. Steps in planning , limitations of planning
4. Making planning effective , planning skills
5. Strategic planning in Indian industry.
6. Meaning, process of organizing
7. Span of management , principles of organizing ,
8. Departmentation
9. Organization structure
10. Committees, teams

Unit – 3

1. Requirements of effective direction
2. Giving orders
3. Motivation
4. Job satisfaction
5. Morale
6. Organizational commitment
7. First level supervision or front line supervision
8. Meaning and steps in controlling
9. Essential of a sound control system
10. Methods of establishing control

Unit – 4

1. Meaning of Entrepreneur
2. Evolution of the Concept
3. Functions of an Entrepreneur
4. Types of Entrepreneur
5. Entrepreneur – an emerging Class
6. Concept of Entrepreneurship
7. Evolution of Entrepreneurship
8. Development of Entrepreneurship
9. Stages in entrepreneurial process
10. Entrepreneurship in India
11. Entrepreneurship – its Barriers

Unit – 5

1. Introduction to IPR
2. Origin and concepts of IPR
3. Concept of property
4. Forms of IP protection: Patents, copyrights, trademarks,
5. designs, Trade secrets, Traditional knowledge, Geographical indications
6. Basic concepts and historical background of patent system and law- National and international scenario (American & European Patent Regimes)
7. International Treaties/Conventions on IPR: Paris Convention
8. Berne convention, Madrid agreement, Rome convention
9. World Intellectual Property Organization (WIPO)
10. TRIPS Agreement
11. Patent Co-operation Treaty

Course Articulation Matrix (CAM)														
Course Outcome (CO)	**Program Outcome (ABET/NBA-(3a-k))**													
		a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Describe the importance of management and functions of a manager.	L2	-	-	-	-	-	**H**	**H**	**H**	**L**	**H**	**-**	-	-
Explain the process of planning and principles of organizing.	L2	-	L	-	-	-	-	M	M	-	H	**L**	-	-
Identify the role of entrepreneurs in the economic development of the nation	L2	-	L	-	-	-	H	**H**	**H**	L	H	-	-	-
Explain the requirements of direction and supervision and **Explain** the methods of establishing control	L2	-	-	-	-	-	L	L	H	L	M	**L**	-	-
List and **Explain** the different forms of Intellectual property protection.	L3	**-**	**-**	**-**	**-**	**-**	**H**	**M**	**L**	L	H	**-**	-	L
L- Low, M- Moderate, H-High														
Illustration of CAM of Management, Entrepreneurship and IPR														
A. Course Assessment Matrix (CAM)														
Course Outcome (CO)		**Program Outcome (ABET/NBA-(3a-k))**												
		a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Describe the importance of management and functions of a manager	L2	-	-	-	-	-	3	3	**3**	**1**	3	-	-	-
Explain the process of planning and Principles of organizing.	L2	-	1	-	-	-	-	2	2	-	1	3	-	-
Identify the role of entrepreneurs in the economic development of the nation	L2	-	1		-	3	3	3	1	3	-	-	-	-
Explain the requirements of direction and supervision and **Explain** the methods of establishing control	L2	-	**-**	**-**	-	-	**1**	**1**	**3**	**1**	2	1	-	-
List and **Explain** the different forms of Intellectual property protection	L3	-	-	-	-	-	3	2	1	1	3	-	-	1
1 – Low, 2 – Moderate and 3 – High														
Illustration of CaM of Management, Entrepreneurship and IPR														

Course Title : MICROPROCESSOR LAB			
Course Code: P13CSL57	**Semester : V**	**L-T-P-H: 0-1-2-3**	**Credits:1.5**
Contact Period : Lecture :36 Hrs., Exam: 3Hrs.		**Weightage :CIE:50% SEE:50%**	

Prerequisites

- assembly language of 8086 processor

Course objectives

This course makes students focus on and should be able,

- To learn assembly level programs of 8086 processor using different addressing modes and instructions.
- To learn interfacing of external devices to 8086.
- To learn the usage of DOS and BIOS interrupts.
- Exposed to tools used to debug the program.

1.) a.) Search a key element in a list of 'n'16-bit numbers using the *Binary Search algorithm*
b.) Interface a logic controller to perform the conversion from hexadecimal to octal. Accept the number from the i/p port of logic controller and display output on the o/p port.

2.) a) Write ALP macros :
i) To read a character from the key board in the module (1) (in a different file.)
ii) To display character in module (2) (from different file)
iii) Use the above two modules to read a string of characters from the key board terminated by the carriage return and print the string in the display in the next line.
b) Write an assembly level program to accept three numbers from the logic controller and find the largest among them and display it on the logic controller.

3) a) Write an assembly level program to generate N prime numbers. Store the numbers starting from location 2000h.
b) Write an assembly level program to perform BCD up-down counter (00-255).

4**)** a) Read your name from the keyboard and display it at a specified location on the screen in front of the message "What is your name?" You must clear the entire screen before display.
b.) Interface seven segment display and display the given message from right to left.

5) a) Write an assembly level program to read the string and convert lowercase to uppercase, uppercase to lowercase. Also find the frequency of occurance of a given character in that string.
b) Display messages 1234 and 5678 alternately with flickering effects on a 7-segment display interface for a suitable period of time.

6**)** a) Read two strings, store them in locations STR1 of data segment and STR2 of extra segment. Check whether they are equal or not and display appropriated messages. Also display the length of the stored strings.
b.) Interface seven segment display and display the given message from left to right.

7) a.) Write an assembly level program to multiply two matrices.
b.) Drive a Stepper Motor interface to rotate the motor both in *clockwise direction and in anti clockwise direction* by N steps (N is specified through the keyboard). Introduce suitable delay between successive steps (Any arbitrary value for the delay may be assumed by the student).

8) a.) Write an assembly level program to sort the elements by using bubble-sort method.
b) S*can* a 8x3 keypad *for key closure* and to *store the code of the key pressed* in a memory location or display on screen. Also display key pressed value, row and column numbers.

9) a) Compute *nCr* using recursive procedure. Assume that 'n' and 'r' are non- negative integers.
b.) Drive an Elevator Interface in the following way :
i) Initially the elevator should be in the ground floor, with all requests in OFF state
ii) When a request is made from a floor, the elevator should move to that floor, service the request, wait there for a couples of seconds, and then come down to ground floor and stop. If some request occur during up or coming down they should be ignored.

10) a) Write an assembly level program to add N bytes of packed BCD numbers. Accept the BCD numbers from the key board. Display the result in BCD form.
b) Write a program to accept two digit numbers from the keypad interface. Perform division/ multiplication.

11) a) Generate the first 'n' *Fibonacci* numbers and store all the Fibonacci numbers starting at even address.
b) Interface DAC to 8086 and display sine waveform on CRO.

12) a) Write an assembly level program to multiply two 2 digit unpacked BCD number.
b) Interface DAC to 8086 and display triangular waveform on CRO.

Course outcomes

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

- Select appropriate instructions and addressing modes to solve problems
- Program external devices connected to CPU.
- Develop simple project..

Note:

a) In the examination each student picks one question from a lot of 12 questions.
b) In case of change of experiment the student have to take change for the whole question (both software and hardware programs)
c) Only one change will be given.

Course Title : SYSTEM SOFTWARE LAB			
Course Code: P13CSL58	**Semester : V**	**L-T-P-H: 0-1-2-3**	**Credits:1.5**
Contact Period : Lecture :36 Hrs., Exam: 3Hrs.		**Weightage :CIE:50% SEE:50%**	

Prerequisites:

- System software
- Unix programming environment

Course objectives

1. Work effectively in a UNIX-style environment.
2. Understand the applications of lex and yacc tools for developing lexical analyzers and parsers.
3. Write programs that use the UNIX system call interface.
4. Write shell scripts.
5. Interacting directly with the operating system by making system calls for file management, file execution and process control.

Lex and Yacc Programs

Execute the following programs using LEX

1. a) Program to count the number of characters, words, spaces and lines in a given input file.
   b) Program to count the number of comment lines in a given C program. Also eliminate them and copy that program into separate file.

2. a) Program to recognize a valid arithmetic expression and identity the identifiers and operators present. Print them separately.
   b) Program to recognize whether a given sentence is simple or compound.

Execute the following programs using YACC

3. a) Program to recognize a valid arithmetic expression that uses operators +, -, * and /.
   b) Program to recognize a valid variable, which starts with a letter, followed by any number of letters or digits.

4. a)Program to evaluate an arithmetic expression involving operators +, -, * and /.
   b)Program to recognize strings ‘aaab’, ‘abbb’, ‘ab’ and ‘a’ using the grammar ($a^n b^n$ n>=0).

UNIX and Shell programming

5. a) Non-recursive shell script that accepts any number of argument and prints them in the Reverse order, (For example, if the script is named rargs, then executing rargs A B C should produce C B A on the standard output.
   b) C program that creates a child process to read commands from the standard input and execute them (a minimal implementation of a shell – like program). You can assume that no arguments will be passed to the commands to be executed

6. a) Write a shell script that displays a list of all the files in the current directory which has the user read, write and execute permissions, and count the same.

b) C Program to implement cp, mv and ls commands.

7. a) Write a shell script that ask for a word and a file name, and then count how many times that word is occurred in the file.
   b) Write a C program that takes one or more file or directory names as command line input and reports the following information on the file:
      i) File type
      ii) Number of links
      iii) Read, write and execute permissions.
      iv) Time of last access

8. a) Write a shell script which receives two file names as arguments. It should check whether the two file contents are same or not. If they are same then second file should be deleted.
   b) Consider the last 100 bytes of a file as a region. Write a C program to check whether the region is locked or not. If the region is locked, print PID of the process which has locked. If the region is not locked, lock the first 50 bytes of the region with an exclusive lock, read the last 50 bytes and unlock the region.

9. a) Shell script that accepts two file names as arguments, checks if the permissions for these files are identical and if the permissions are identical, outputs the common permissions, otherwise outputs each file name followed by its permissions.
   b) C program to create a file with 16 bytes of arbitrary data from the beginning and another 16 bytes of arbitrary data from an offset of 48. Display the file contents to demonstrate how the hole in file is handled.

10. a) Write a shell script to create a directory, write contents on that and Copy to a suitable location in your home directory.
    b) C program to do the following: Using fork( ) create a child process. The child process prints its own process-id and id of its parent and then exits. The parent process waits for its child to finish (by executing the wait( )) and prints its own process-id and the id of its child process and then exits.

11. a) Write a shell script to search an element from an array using binary searching.
    b) Write a C program that creates a zombie and then calls system to execute the ps command to verify that the process is zombie.

12. a) Write a shell script that determines the period for which a specified user is working on the system.
    b) Write a C program to avoid zombie process by forking twice.

Course outcomes

Upon completion of this lab students should be able to

1. Use the powerful compiler generation tools such as Lex and YACC.
2. Writing programs in a scripting language.
3. Implement linux commands.
4. Implement programs for file management.
5. Use system calls for managing processes.

Course Title : WEB TECHNOLOGIES			
Course Code: P13CS61	**Semester : VI**	**L-T-P-H: 3-0-2-5**	**Credit: 4**
Contact Period : Lecture :52 Hrs., Exam: 3Hrs.		**Weightage :CIE:50% SEE:50%**	

Prerequisites

- Basic concepts of programming language
- Client server communications
- Computer networks

Course Learning Objectives

1. This course is intended to provide an exposure to fundamental concepts of WWW, Internet, Browsers, Servers, URL, MIME, HTTP
2. To present competent technologies for the design of Web using XHTML and CSS.
3. To provide knowledge of scripting languages such as JavaScript and design dynamic XHTML documents using DOM and JavaScript
4. To create XML documents using DTD/ XML schema and XSLT style sheets and create cookies using PHP , Implement session tracking using PHP
5. To develop a Rails application using Ajax

Course content

Unit - 1

Fundamentals of Web: Internet, WWW, Web Browsers and Web Servers, URLs, MIME, HTTP, Security, The Web Programmers Toolbox.

Introduction to XHTML: Origins and evolution of HTML and XHTML, Basic syntax, Standard XHTML document structure, Basic text markup, Images, Hypertext Links, Lists, Tables, Forms, Frames, Syntactic differences between HTML and XHTML. 10 Hours

Unit - 2

CSS: Introduction, Levels of style sheets, Style specification formats, Selector forms, Property value forms, Font properties, List properties, Color, Alignment of text, The Box model, Background images, The and tags

JAVASCRIPT: Overview of JavaScript, Object orientation and JavaScript, General syntactic characteristics, Primitives, operations, and expressions, Screen output and keyboard input, Control statements, Object creation and modification, Arrays, Functions, Constructor, Pattern matching using regular expressions, Errors in scripts, Examples. 10 Hours

Unit - 3

JAVASCRIPT AND HTML DOCUMENTS: The JavaScript execution environment; The Document Object Model; Element access in JavaScript; Events and event handling; Handling events from the Body elements, Button elements, Text box and Password elements

DYNAMIC DOCUMENTS WITH JAVASCRIPT: Introduction to dynamic documents; Positioning elements; Moving elements; Element visibility; Changing colors and fonts; Dynamic content; Stacking elements; Locating the mouse cursor; Reacting to a mouse click; Slow movement of elements; Dragging and dropping elements. 11 Hours

Unit-4

XML: Introduction; Syntax; Document structure; Document Type definitions; Namespaces; XML schemas; Displaying raw XML documents; Displaying XML documents with CSS; XSLT style sheets; XML processors; Web services.

PHP: Origins and uses of PHP, Overview of PHP, General syntactic characteristics, Primitives, Operations and expressions, Output, Control statements, Arrays, Functions, Pattern matching, Form handling, Files, Cookies, Session tracking . 11 Hours

Unit-5

Introduction to Rails : Overview of Rails, Document requests, Processing forms, Rails applications with databases, **Introduction to Ajax**Overview of Ajax, Basics of Ajax, Rails with Ajax. 10 Hours

TEXT BOOK:

1. **Programming the World Wide Web** – Robert W. Sebesta, 4th Edition, Pearson Ed. 2008.

REFERENCE BOOKS:

1. Internet & World Wide Web How to program – M. Deitel, P.J Deitel, A. B. Goldberg, 3rd Edition, Pearson Education / PHI, 2004.

2. Web Programming Building Internet Applications – Chris Bates, 3rd Edition, Wiley India, 2006.

3. The Web Warrior Guide to Web Programming – Xue Bai et al,

Course Outcomes:

At the end of the course the student should be able to

1. **Explain** the fundamental concepts WWW, Web Browsers and Web Servers, URLs, MIME, HTTP, and **Create** XHTML documents using XHTML tags
2. **Develop** XHTML documents with CSS properties
3. **Design** dynamic XHTML documents using DOM and JavaScript.
4. **Create** XML documents using DTD/ XML schema and XSLT style sheets and create cookies using PHP , Implement session tracking using PHP
5. **Develop** a Rails application that uses Ajax.

Topic Learning Objectives

Unit-1

Topic :

Fundamentals of Web**,** Internet, WWW, Web Browsers and Web Servers, URLs, MIME, HTTP, Security, The Web Programmers Toolbox.

Introduction to XHTM, Origins and evolution of HTML and XHTML, Basic syntax, Standard XHTML document structure, Basic text markup, Images, Hypertext Links, Lists, Tables, Forms, Frames, Syntactic differences between HTML and XHTML.

At the end of this chapter student should be able to:

1. **Explain** internet, WWW, Browsers, Servers, URL, MIME, HTTP (L2).
2. **Describe** basic XHML tags <p>,< img>, Hypertext links.(L2)
3. **Create** an XHTML documents using <p>,< img>, Hypertext links.(L3).
4. **Create** and validate an XHTML document (L3).
5. **Develop** an XHTML document to describe ordered, unordered and definition list (L3).
6. **Develop** an XHTML document using <table> and various attributes (L3).
7. **Create** an XHTML document using <form> and widgets (L3).
8. **Develop** an XHTML document using <frames> (L3).

Unit-2

Topic :

CSS: Introduction, Levels of style sheets, Style specification formats, Selector forms, Property value forms, Font properties, List properties, Color, Alignment of text, The Box model, Background images, The and tags

JAVASCRIPT: Overview of JavaScript, Object orientation and JavaScript, General syntactic characteristics, Primitives, operations, and expressions, Screen output and keyboard input, Control statements, Object creation and modification, Arrays, Functions, Constructor, Pattern matching using regular expressions, Errors in scripts, Examples.

At the end this chapter student should be able to:

1. **Explain** the different levels of style sheets (L2).
2. **Develop** an XHTML document using font, list, image and color properties (L3).
3. **Develop** an XHTML document using cascading style sheets (L3).
4. **Understand** the general syntactic characteristics, primitives, operation, expressions, control statements of JavaScript (L1).
5. **Write** a simple JavaScript script using alert( ), prompt( ) and write( ) methods (L3).
6. **Create** objects and object modification in JavaScript (L3).
7. **Write** an XHTML document and JavaScript script using user defined functions (L3).
8. **Write** an XHTML document and JavaScript script using pattern matching methods and regular expressions (L3).

Unit-3

Topic :

JAVASCRIPT AND HTML DOCUMENTS: The JavaScript execution environment; The Document Object Model; Element access in JavaScript; Events and event handling; Handling events from the Body elements, Button elements, Text box and Password elements

DYNAMIC DOCUMENTS WITH JAVASCRIPT: Introduction to dynamic documents; Positioning elements; Moving elements; Element visibility; Changing colors and fonts; Dynamic content; Stacking elements; Locating the mouse cursor; Reacting to a mouse click; Slow movement of elements; Dragging and dropping elements.

At the end this chapter student should be able to:

1. **Develop** an XHTML document to handle events from body elements and button elements using JavaScript (L3).
2. **Create** an XHTML document to handle events from text box and password elements using JavaScript (L3).
3. **Develop** an XHTML document for validating form input using JavaScript (L3).
4. **Explain** dynamic XHTML, positioning of elements using CSS properties (L1).
5. **Explain** informal guidelines for the database design. (L2).
6. **Create** an XHTML document to move an element from its original position to its new position (L3).
7. **Develop** an XHTML document to illustrate the visibility control of elements using visibility property (L3).

Unit-4

Topic :

XML: Introduction, Syntax, Document structure, Document Type definitions, Namespaces, XML schemas, Displaying raw XML documents, Displaying XML documents with CSS, XSLT style sheets, XML processors, Web services

PHP: Origins and uses of PHP, Overview of PHP, General syntactic characteristics, Primitives, Operations and expressions, Output, Control statements, Arrays, Functions, Pattern matching, Form handling, Files, Cookies, Session tracking

At the end this chapter student should be able to:

1. **Explain** about XML, the syntax of XML, XML document structure (L2).
2. **Explain** DTD, structure of DTD and DTD declarations (L2).
3. **Write** a DTD for a given application and an XML document that is valid for the DTD (L3).
4. **Explain** XML schemas, defining a schema and defining a schema instance (L2).
5. **Write** a XML schema for a given application and an XML document that is valid for the XML schema (L3).
6. **Write** an XML document and display with CSS (L3).
7. **Write** XSLT style sheet using templates (L3).
8. **Learn** the fundamentals of PHP (L1).
9. **Write** PHP script for form handling(L3)
10. **Create** cookies using PHP (L3).

11. **Implement** session tracking in PHP (L3).

Unit-5

Topic :

Introduction to Rails

Overview of Rails, Document requests, Processing forms, Rails applications with databases,

Introduction to Ajax

Overview of Ajax, Basics of Ajax, Rails with Ajax

At the end this chapter student should be able to:

1. **Explain** MVC, Document request, static documents, dynamic documents(L2)
2. **Develop** a simple application using Rails(L6)
3. **Design** Rails application with databases(L6)
4. **Explain** the concept of Ajax(L2)

Review questions

Unit 1

1. What is the task of a DNS server? Explain with a diagram Domain name conversion.
2. Briefly explain the two phases of HTTP protocol.
3. Distinguish between HTML and XHTML.
4. Describe the purpose of the five most commonly used HTTTP methods.
5. Explain the working of a web server with the help of a digram.
6. Explain the following tags i) <.meta> ii) <img> iii) <a> iv)<input>
7. Describe the standard HTML document structure with an example.
8. Create an XHTML document that has a form with check boxes, radio buttons, menus
9. Write an XHTML document to describe an ordered list of your five favorite movies. Each element of the list must have a nested list of two actors in your favorite movies.
10. What is the difference between cell padding and cell spacing attributes.

Unit 2

1. What is CSS? Describe the different levels of style sheets and their precedence.
2. List and explain the various selector forms with example.
3. Explain the box model in CSS.
4. Create an XHTML document that describes an unordered list of atleast five popular books. Use style sheet to define defferent list items to have different bullet types.
5. Explain <Span> tag and >div> tag.
6. How JavaScript script is embedded in an XHTML document? Briefly explain three major uses of JavaScript on client side.
7. Explain the different types of variables in JavaScript
8. Describe the two ways an array object can be created?
9. Write an XHTML document that includes JavaScript, parameter: Number, Output: Number with its digits in the reverse order.
10. Write an XHTML document that includes JavaScript to display a table from 5 to 15 and their squares and cubes using alert.

Unit 3

1. What is DOM? Describe the DOM structure for a simple document.
2. What is an event, event handler, event registration. List any five most commonly used attributes related to events
3. Describe the different ways of accessing the elements of an XHTML document in JavaScript.
4. Develop an XHTML document that collects your USN and an event handler to validate your USN on clicking the submit button using JavaScript. Display the error messge using ‘alert’.
5. What is a dynamic XHTML document? How XHTML elements can be placed on the display of a document.
6. Explain the different properties of positioning elements with example.

7. Write an XHTML document and JavaScript script to illustrate the stacking of images.
8. Write an XHTML document and JavaScript script to illustrate the movement of an element from its original position to its new position.
9. Write an XHTML document and JavaScript script to illustrate visibility property of elements.

Unit 4

1. What is XML. Explain the syntax of a simple XML document.
2. What is DTD? Explain how elements, attributes and entities are declared in DTD with example.
3. What are XML namespaces? Explain with an example.
4. What is XML schema? How a schema and a schema instance is defined?
5. Explain the transformation process carried out by XSLT processor.
6. What is PHP? How PHP scripts are embedded in XHTML documents.
7. What are cookies? How cookies are set in PHP?
8. What is a session? Write a PHP script to store page view counts in session, to increment the count on each refresh and to show the count on web page.
9. List and give the definitions of different string functions used in PHP.
10. Explain the different file operations used in PHP with example.

Unit 5

1. What is Rails? Explain MVC architecture.
2. Explain how Rails react to a request for static document with an example.
3. Write a Rails application that accepts two integer values and produces the product of the two values and returns it to the client.
4. How are simple form data gotten by a form processing action method.
5. Explain with a neat diagram the traditional and Ajax client/server interactions.
6. Describe the request phase and receiver phase in Ajax.
7. How are parameters passed in a GET Ajax request?
8. Write a note on Rails that uses Ajax.

Lesson Plan

Unit-1

1. Internet, WWW, Web Browsers and Web Servers,
2. URLs, MIME,
3. HTTP, Security,
4. The Web Programmers Toolbox.
5. Origins and evolution of HTML and XHTML, Basic syntax,
6. Standard XHTML document structure,
7. Basic text markup,
8. Images, Hypertext Links, Lists,
9. Tables, Forms,
10. Frames, Syntactic differences between HTML and XHTML.

Unit-2

1. Introduction, Levels of style sheets, Style specification formats,
2. Selector forms, Property value forms,
3. Font properties, List properties, Color, Alignment of text,
4. The Box model, Background images, The and tags
5. Overview of JavaScript, Object orientation and JavaScript,
6. General syntactic characteristics, Primitives, operations, and expressions,
7. Screen output and keyboard input,
8. Control statements, Object creation and modification,
9. Arrays, characteristics of array objects, Array methods
10. Functions, Constructor,

11. Pattern matching using regular expressions, Errors in scripts, Examples.

Unit-3

1. The JavaScript execution environment; The Document Object Model;
2. Element access in JavaScript
3. Events and event handling
4. Handling events from the Body elements,
5. Handling events from Button elements, Text box and Password elements
6. Introduction to dynamic documents; Positioning elements
7. Moving elements; Element visibility;
8. Changing colors and fonts; Dynamic content;
9. Stacking elements; Locating the mouse cursor; Reacting to a mouse click
10. Slow movement of elements; Dragging and dropping elements.

Unit-4

1. Introduction to XML, Syntax, Document structure
2. Document Type definitions
3. Namespaces, XML schemas
4. Displaying raw XML documents, Displaying XML documents with CSS
5. XSLT style sheets, XML processors, Web services
6. Origins and uses of PHP, Overview of PHP,
7. General syntactic characteristics, Primitives, Operations and expressions,
8. Output, Control statements,
9. Arrays, Functions,
10. Pattern matching, Form handling, Files,
11. Cookies, Session tracking

Unit-5

1. Overview of Rails, Document requests, static documents, dynamic documents
2. Processing forms, setting up the application,
3. The controller and the view
4. Rails applications with databases, building the databases
5. Building the application, layouts
6. Overview of Ajax,
7. Basics of Ajax, the application, the form document
8. The request phase, response document
9. Receiver phase, cross browser support
10. Rails with Ajax, initial form document, triggering Ajax

Course Articulation Matrix (CAM)														
Course Outcome (CO)		**Program Outcome (ABET/NBA-(3a-k))**												
		a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Explain the fundamental concepts WWW, Web Browsers and Web Servers, URLs, MIME, HTTP, and **Create** XHTML documents using XHTML tags	**L2**	M	M	M	-	H	-	-	-	-	-	M	M	M
Develop XHTML documents with CSS properties	**L6**	M	M	M	-	H	-	-	-	-	-	M	M	M
Design dynamic XHTML documents using DOM and JavaScript.	**L6**	M	M	H	-	H	-	-	-	-	-	M	M	M
Create XML documents using DTD/ XML schema and XSLT style sheets and create cookies using PHP , Implement session tracking using PHP	**L6**	M	M	M	-	H	-	-	-	-	-	M	M	M
Build a Rails application that uses Ajax.	**L6**	M	M	M	-	H						M	M	M
L- Low, M- Moderate, H-High														

D. Course Assessment Matrix (CaM)														
Course Outcome (CO)		**Program Outcome (ABET/NBA-(3a-k))**												
		a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Explain the fundamental concepts WWW, Web Browsers and Web Servers, URLs, MIME, HTTP, and **Create** XHTML documents using XHTML tags	**L2**	2	2	2	-	3	-	2	-	1	-	2	2	2
Develop XHTML documents with CSS properties	**L6**	-	2	2	-	3	-	2	-	3	-	2	2	2
Design dynamic XHTML documents using DOM and JavaScript.	**L6**	2	2	2	-	3	-	-	-	2	-	2	2	2
Create XML documents using DTD/ XML schema and XSLT style sheets and create cookies using PHP , Implement session tracking using PHP	**L6**	2	2	2	-	3	-	-	-	-	-	2	2	2
Develop a Rails application that uses Ajax.	**L6**	2	2	2	-	3	-	-	-	-	-	2	2	2
1 – Low, 2 – Moderate and 3 – High														

Course Title : Computer Networks			
Course Code: P13CS62	**Semester : VI**	**L-T-P-H: 3 -2-0-5**	**Credits:4**
Contact Period : Lecture :52 Hrs., Exam: 3Hrs.		**Weightage :CIE:50% SEE:50%**	

Prerequisite

- Data communication

Course Learning Objectives

1. **Understand** and recognize the importance of network layer and its functionalities.
2. **Analyze** various routing algorithms and the need of upgrading to IPv6 protocol.
3. **Differentiate** between TCP & UDP, and understand the services provided by them like flow control, congestion control etc.
4. **Understand** and recognize the importance of application layer and the working of protocols like HTTP, FTP, DNS etc
5. **Analyze** the different types of quality of service and understand some of the factors driving the need for network and Internet security.

Course content

Unit – 1

Network layer: Network-layer services, packet switching, network-layer performance, IPv4 addresses, forwarding of IP packets.
Network-layer protocols: Internet protocol (IP), ICMPv4, mobile ip 11 hours

Unit – 2

Unicast routing: Introduction, routing algorithms, unicast routing protocols.
Multicast routing: Introduction, multicasting basics, intradomain multicast protocols, interdomain multicast protocols, IGMP.
Next generation ip: IPv6 addressing, the IPv6 protocol, the ICMPv6 protocol, transition from ipv4 to ipv6. 10 hours

Unit – 3

Transport layer: Introduction, transport-layer protocols.
Transport-layer protocols: Introduction, user datagram protocol, transmission control protocol, SCTP. 10 hours

Unit – 4

Application layer: Introduction, client-server programming, iterative programming in c.
Standard client-server protocols : World wide web and HTTP, FTP, electronic mail, telnet, secure shell (ssh), domain name system (dns).
Network management: Introduction, SNMP, asn.1. 10 hours

Unit – 5

Quality of service: Data-flow characteristics, flow control to improve qos, integrated services (intserv), differentiated services (dffserv).
Cryptography and network security: Introduction, confidentiality, other aspects of security.
Internet security: Network-layer security, transport-layer security, application-layer security, firewalls. 11 hours

Text Book:

1. Behrouz A. Forouzan: Data communication and Networking, 5th edition, Tata McGraw-Hill, 2012.

Reference Books:

1. Larry L. Peterson and Bruce S Davie: Computer Networks: A Systems Approach, Fifth Edition, Elsevier, 2011.

2. William Stallings: Data and Computer Communications, 8th Edition, Pearson Education, 2012.
3. James F. Kurose and Keith W. Ross: Computer Networking: A Top-Down Approach, 6th edition, Addison-Wesley, 2009.
4. Tanenbaum: Computer Networks, 5th Ed, Pearson Education/PHI, 2011.

Course outcomes

1. Discuss IPv4 protocols and its functions provided at networks layer.
2. Analyze various routing algorithms like distance vector, link state, hierarchical & multicast routing, and understand the concept of fragmentation.
3. Differentiate between TCP, UDP & SCTP and understand the services provided by them like flow control, congestion control etc.
4. Understand and recognize the importance of application layer and the working of protocols like HTTP, FTP, DNS etc
5. Analyze the different types of quality of service and understand the concept of Network & Internet security.

Topic Learning Objectives

Unit 1

Topic: Network layer: Network-layer services, packet switching, network-layer performance, ipv4 addresses, forwarding of ip packets.

Network-layer protocols

Internet protocol (ip), icmpv4, mobile ip

Learning Objective

1. To introduce switching and in particular packet switching as the mechanism of data delivery in the network layer. L4
2. To discuss two distinct types of services a packet-switch network can provide: connectionless service and connection-oriented service.L6
3. To evaluates how routers forward packets in a connectionless packet switch network using the destination address of the packet and a routing table.L6
4. To discuss how routers forward packets in a connection-oriented packet-switch network using the label on the packet and a routing table. L6
5. To introduce the concept of an address space in general and the address space of IPv4 in particular. L4
6. To discuss the classful architecture, classes in this model, and the blocks of addresses available in each class. L6
7. To discuss NAT technology and show how it can be used to improve the shortage in number of addresses in IPv4. L6 L2.
8. To explain the general idea behind the IP protocol and show the position of IP in relation to other protocols in TCP/IP protocol suite. L6
9. To show the general format of an IPv4 datagram and list the fields in the header.L2
10. To show how a checksum is calculated for the header of an IPv4 datagram at the sending site and how the checksum is checked at the receiver site.L2
11. To show how the address resolution protocol (ARP) is used to dynamically map a logical address to a physical address.L2

Unit 2

Topic: Unicast routing Introduction, routing algorithms, unicast routing protocols.

Multicast routing Introduction, multicasting basics, intradomain multicast protocols, interdomain multicast protocols, igmp.

Next generation ip Ipv6 addressing, the ipv6 protocol, the icmpv6 protocol, transition from ipv4 to ipv6.

Learning Objective

1. To discuss how the Routing Information Protocol (RIP) is used to implement the idea of distance vector routing in the Internet.L6, L4
2. To discuss how Open Shortest Path First (OSPF) is used to implement the idea of link state routing in the Internet. L6, L4
3. To discuss how Border Gateway Protocol (BGP) is used to implement the idea of path vector routing in the Internet. L6, L4
4. To discuss and differentiate between virtual circuit and datagram network.L6
5. Analyze the link state routing algorithms.L4
6. Discuss and apply distance vector routing and hierarchical algorithm for finding feasible path.L6,L3
7. Differentiate between broadcast and multicast routing.L4
8. To use the format of an IPv6 datagram discuss composed of a base header and a payload.L2,L6
9. To discuss different fields used in an IPv6 datagram based header and compare them with the fields in IPv4 datagram. L6

Unit 3

Topic: Transport layer Introduction, transport-layer protocols.

Transport-layer protocols Introduction, user datagram protocol, transmission control protocol, sctp.

Learning Objective

1. To define process-to-process communication at the transport layer and compare it with host-to-host communication at the network layer. L1, L4
2. To discuss the addressing mechanism at the transport layer, to discuss port numbers, and to define the range port numbers used for different purposes.L6,L1
3. To illustrates UDP services and show its relationship to other protocols in the TCP/IP protocol suite.L2, L4
4. To explain the format of a UDP packet, which is called a user datagram, and discuss the use of each field in the header. L6
5. To discuss TCP as a protocol that provides reliable stream delivery service. L6
6. To define TCP features and compare them with UDP features.L1, L4
7. To define the format of a TCP segment and its fields.L1
8. To show how TCP provides a connection-oriented service, and show the segments exchanged during connection establishment and connection termination phases.L2
9. To discuss the state transition diagram for TCP and discuss some scenarios.L6
10. To explain the flow control, error control and congestion control mechanism in TCP
11. To discuss SCTP services and compare them with TCP.L6,L4
12. To compare and contrast the state transition diagram of SCTP with the corresponding diagram of TCP.L4
13. To explain the flow control, error control and congestion control mechanism in SCTP and discuss the behavior of the sender site and the receiver site.L6

Unit 4

Topic: Application layer : Introduction, client-server programming, iterative programming in c.

Standard client-server protocols World wide web and http, ftp, electronic mail, telnet, secure shell (ssh), domain name system (dns).

Network management Introduction, snmp, asn.1.

Learning Objective

1. To introduce the Client-Server Programming and show how it implements in networks.L4 L2
2. To describe the purpose of DNS.L6

3. To introduce the TELNET protocol and show how it implements local and remote login using the concept of network virtual terminal.L4 L2
4. To show how different components of SSH are combined to provide a secure connection over an insecure TCP connection.L2
5. To discuss FTP and two connections used in this protocol: control connection and data connection.L6
6. To explain three types of file transfer transferred by FTP.L6
7. To discuss five types of TFTP messages and their applications.
8. To discuss the architecture of WWW and describe the concepts of hypertext and hypermedia.L6
9. To introduce three different Web documents: static document, dynamic document, and active document.L5
10. To introduce cookies and their applications in HTTP.L4
11. To explain the mechanism of sending and receiving e-mails.L6
12. To demonstrates e-mail transfer phases. L3
13. To discuss two message access agents (MAAs): POP and IMAP.L6
14. To show how SNMP can be used for managing network over the Internet.L2

Unit 5

Topic: Quality of service Data-flow characteristics, flow control to improve qos, integrated services (intserv), differentiated services (dffserv).

Cryptography and network security Introduction, confidentiality, other aspects of security.

Internet security Network-layer security, transport-layer security, application-layer security, firewalls.

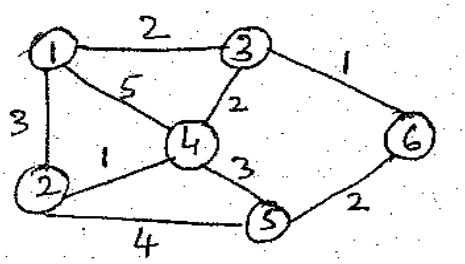
Learning Objective

1. To discuss the general idea of improving QoS over the internet.L6
2. To introduce security goals and to discuss the types of attacks that threatens these goals.L4
3. To introduce traditional ciphers as symmetric-key ciphers to create the background for understanding modern symmetric-key ciphers.L4
4. To introduce the elements of modern block ciphers and show an example of a modern block cipher in which these elements are used.L4
5. To discuss the general idea behind asymmetric-key ciphers and introduce one common cipher in this category.L6
6. To show how secret keys in symmetric-key cryptography and how public keys in asymmetric-key cryptography can be distributed and managed using KDCs or certificate authorities (CAs).L2
7. To introduce the idea of Internet security at the network layer and the IPSec protocol that implements that idea in two modes: transport and tunnel.L4
8. To discuss two protocols in IPSec, AH and ESP, and explain the security services each provide.L6
9. To introduce security association and its implementation in IPSec.L4

Review Questions

1. Find the class of each Address.
   a) 11000010 10000011 00011011 11111111
   b) 241.5.15.111
2. A block of addresses is granted to a small organization. We know that one of the addresses is 205.16.37.39/26. Find the number of addresses.
3. Calculate the HLEN (in IPv4) value if the total length is 2400 bytes, 2380 of which is data from the upper layer.
4. Change the multicast IP address 220.40.12.7 to an Ethernet multicast physical address.
5. How does the hop count limit alleviate RIP's problems?

6. An ISP is granted a block of addresses starting with 150.80.0.0/16. The ISP wants to distribute these blocks to 2600 customers as follows.
   a. The first group has 200 medium-size businesses; each needs 128 addresses.
   b. The second group has 400 small businesses; each needs 16 addresses.
   c. The third group has 2000 households; each needs 4 addresses.
   Design the subblocks and give the slash notation for each subblock. Find out how many addresses are still available after these allocations
7. What is NAT? How can NAT help in address depletion?
8. How can we distinguish a multicast address in IPv4 addressing? How can we do so in IPv6 addressing?
9. Define fragmentation and explain why the IPv4 and IPv6 protocols need to fragment some packets. Is there any difference between the two protocols in this matter?
10. Compare and contrast the options in IPv4 and the extension headers in IPv6. Make a table that shows the presence or absence of each.
11. List three transition strategies to move from IPv4 to IPv6. Explain the difference between tunneling and dual stack strategies during the transition period. When is each strategy used?
12. An IPv4 datagram has arrived with the following information **in** the header (in hexadecimal):
    Ox45 00 00 54 00 03 58 50 20 06 00 00 7C 4E 03 02 B4 OE OF 02
    a. Is the packet corrupted?
    b. Are there any options?
    c. Is the packet fragmented?
    d. What is the size of the data?
    e. How many more routers can the packet travel to?
    f. What is the identification number of the packet?
    g. What is the type of service?
13. What is ARP? Explain its functions on a typical internet.
14. Consider the network in the following figure Q14: Using Dijkstra's algorithm finds the routing table for node A.



15. Contrast and compare distance vector routing with link state routing..
16. In cases where reliability is not of primary importance, UDP would make a good transport protocol. Give examples of specific cases.
17. What is the maximum size of the TCP header? What is the minimum size of the TCP header?
18. What are the three domains of the domain name space?
19. Why was there a need for DDNS?
20. Compare the TCP header and the UDP header. List the fields in the TCP header that are missing from UDP header. Give the reason for their absence.
21. The following is a dump of a UDP header in hexadecimal format.
    0632000DOO lCE217
    a. What is the source port number?
    b. What is the destination port number?
    c. What is the total length of the user datagram?
    d. What is the length of the data?

e. Is the packet directed from a client to a server or vice versa?
f. What is the client process?

22. A window holds bytes 2001 to 5000. The next byte to be sent is 3001. Draw a figure to show the situation of the window after the following two events.
    a) An ACK segment with the acknowledgment number 2500 and window size advertisement 4000 is received
    b) A segment carrying 1000 bytes is sent.
23. Explain the three way handshake for establishing a TCP connection & Termination.
24. In SCTP, the state of a sender is as follows:
    a. The sending queue has chunks 18 to 23.
    b. The value of *cumTSN* is 20.
    c. The value of the window size is 2000 bytes.
    d. The value of *inTransit* is 200.

    If each data chunk contains 100 bytes of data, how many DATA chunks can be sent now? What is the next DATA chunk to be sent?
25. Do you think a recursive resolution is normally faster than an interactive one? Explain.
26. Explain the types of DNS messages.
27. Explain the architecture of an Email?
28. Explain in brief the data transfer between the two processes.
29. In RSA, given n = 12091, e = 13, and d = 3653 encrypt the message "THIS IS TOUGH" using the 00 to 26 encoding scheme. Decrypt the ciphertext to find the original message. Use 4-digit plaintext cipher text blocks.
30. In RSA, why can't Bob choose 1 as the public key e?
31. How are congestion control and quality of service related?
32. What is the difference between open-loop congestion control and closed-loop congestion control?
33. What is the major difference between Integrated Services and Differentiated Services?
34. In regard to quality of service, how do user-related attributes differ from networkr elated attributes in ATM?
35. An ATM network has lost 5 cells out of 10,000 and 2 are in error. What is the CLR? What is the CER?
36. List five functions of network management.
37. To make Diffie-Helman algorithm more robust, one uses cookies. Do some research and find out about the use of cookies in the Diffie-Helman algorithm.
38. What is the difference between a session and a connection in SSL?
39. Does the PGP Protocol need the services of a KDC? Explain your answer.
40. Are there any cipher suites in IPSec? Explain your answer.
41. Are there any cipher suites in PGP? Explain your answer.
42. What is a VPN and why is it needed?

Week 1: **Network layer**
Network-layer services, packet switching, network-layer performance. Ipv4 addresses, forwarding of ip packets.
Week 2: **Network-layer protocols**
Internet protocol (ip), icmpv4, mobile ip
Week 3: **Unicast routing**
Introduction, routing algorithms, unicast routing protocols.
Week 4: **Multicast routing**
Introduction, multicasting basics, intradomain multicast protocols, interdomain multicast protocols, igmp.
Week 5: **Next generation ip**
Ipv6 addressing, the ipv6 protocol, the icmpv6 protocol, transition from ipv4 to ipv6.

Week 6: **Transport layer**
Introduction, transport-layer protocols.
Week 7: **Transport-layer protocols**
Introduction, user datagram protocol, transmission control protocol, sctp.
Week 8: **Application layer**
Introduction, client-server programming, iterative programming in c.
Week 9: **Standard client-server protocols**
World wide web and http,
Week 10: **Standard client-server protocols**
Ftp, electronic mail, telnet, secure shell (ssh), domain name system (dns).
Week 11: **Network management**
Introduction, snmp, asn.1.
Week 12: **Quality of service**
Data-flow characteristics, flow control to improve qos, integrated services (intserv), differentiated services (dffserv).
Week 13: **Cryptography and network security**
Introduction, confidentiality, other aspects of security.
Week 14: **Internet security**
Network-layer security, transport-layer security, application-layer security, firewalls.

E. Course Articulation Matrix (CAM)												
Course Outcome (CO)		**Program Outcome (ABET/NBA-(3a-k))**										
		a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
Discuss IPv4 protocols and its functions provided at networks layer.	**L2**	M	M		L	M	L		H			
Analyze various routing algorithms like distance vector, link state, hierarchical & multicast routing, and understand the concept of fragmentation.	**L2, L4**	H	H			M	L		M			
Differentiate between TCP, UDP & SCTP and understand the services provided by them like flow control, congestion control etc.	**L2**	L	H			M	L		M			
Understand and recognize the importance of application layer and the working of protocols like HTTP, FTP, DNS etc	**L2**	L	H		L	M	M		M			
Analyze the different types of quality of service and understand the concept of Network & Internet security.	**L2, L4**	H	H		L	M	H		M			
L- Low, M- Moderate, H-High												
Fig 3. Illustration of CAM of Digital Circuits Design												

F. Course Assessment Matrix (CaM)												
Course Outcome (CO)		**Program Outcome (ABET/NBA-(3a-k))**										
		a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
Discuss IPv4 protocols and its functions provided at networks layer.	**L2**	2	2		1	2	1		3			
Analyze various routing algorithms like distance vector, link state, hierarchical & multicast routing, and understand the concept of fragmentation.	**L2 , L4**	3	3			2	1		2			
Differentiate between TCP, UDP & SCTP and understand the services provided by them like flow control, congestion control etc.	**L2**	1	3			2	1		2			
Understand and recognize the importance of application layer and the working of protocols like HTTP, FTP, DNS etc	**L2**	1	3		1	2	2		2			
Analyze the different types of quality of service and understand the concept of Network & Internet security.	**L2 , L4**	2	3		1	2	3		2			

Course Title : COMPILER DESIGN			
Course Code: P13CS63	**Semester : VI**	**L-T-P-H: 4- 0-0-4**	**Credits:4**
Contact Period : Lecture :52 Hrs., Exam: 3Hrs.		**Weightage :CIE:50% SEE:50%**	

Prerequisites

- Finite automata and formal languages
- System software

Course Learning Objectives

1. The main objective of this course is to gain in-depth knowledge in **understanding the** compilation process
2. **Understand** the phases of the compilation process and **Know** about the compiler generation tools, role of lexical analyzer for designing a compiler.
3. **Learn top down** parsing techniques.
4. **Learn Bottom up** parsing techniques and **analysis** of ambiguous grammar in the specification and implementation of languages.
5. **Know** how dependency graph is used in evaluation of SDD's ,**Learn** role of a semantic analyzer and type checking, how allocation and deallocation can be done during run time.
6. **Learn** intermediate machine representations and understand the concept of code generation.

Course content

Unit – 1

Introduction, Lexical analysis, Syntax analysis: Various phases of a compiler ,Grouping of phases; Compiler-Construction tools; Lexical analysis: The Role of Lexical Analyzer; Input Buffering; Specifications of Tokens; Recognition of Tokens. 10 Hours

Unit – 2

Syntax Analysis–: Role of parser; Context-free grammars; Top-down Parsing. 11 Hours

Unit – 3

Syntax Analysis : Bottom-up Parsing, LR parsers, Using ambiguous grammars. 10 Hours

Unit – 4

Syntax-Directed Translation: Syntax-directed definitions; Construction of syntax tree ;Evaluation orders for SDDs; Syntax-directed translation schemes. **Type checking**-Type Systems; Specification of a simple type checker; Equivalence of type expression; Type conversions.

Run-Time Environments: Source language issues; Storage Organization; Storage allocation strategies; parameter passing; Symbol tables; dynamic storage allocation techniques.

11 Hours

Unit – 5

Intermediate Code Generation: Intermediate languages; declaration; Assignment statements; Boolean expressions; Case statements; Back patching ;Procedure calls.

Code Generation: Issues in the design of Code Generator; basic blocks and flow graphs; A simple code generation; Register allocation and assignment; DAG representation of basic blocks. 10 Hours

Text Book:

1. Alfred V Aho, Monica S.Lam, Ravi Sethi, Jeffrey D Ullman: Compilers- Principles, Techniques and Tools, 2nd Edition, Pearson Education, 2007.

Reference Books:

1. Charles N. Fischer, Richard J. leBlanc, Jr.: Crafting a Compiler with C, Pearson Education, 1991.
2. Andrew W Apple: Modern Compiler Implementation in C, Cambridge University Press, 1997.
3. Kenneth C Louden: Compiler Construction Principles & Practice, Cengage Learning, 1997.

Course Outcomes

At the end of the course the student should be able:

1. **Describe** the various phases of a compiler, software tools to build compiler and implement lexical analyzer.
2. **Apply** an algorithm for top down parsing.
3. **Apply** an algorithm for bottom-up parsing .
4. **Write** a syntax-directed definition and **Explain** the role of a semantic analyzer , type checking , allocation and deallocation.
5. **Apply** intermediate machine representations and **Analyze** various issues in allocation strategies and design code generator.

Topic Learning Objectives

Unit 1

Topic : Introduction, Lexical analysis, syntax analysis

Learning Objective

1. Explain the phases of the compilation process.
2. To describe the purpose and implementation approach of each phase.
3. To learn software tool used in compiler construction such as lexical analyzer generator.
4. Understand the tokens
5. Describe regular expression for tokens
6. Explain secondary task performed by LA such as removal of comments etc.

Unit 2

Topic: Syntax Analysis

Learning Objective

1. Write grammar to c statements.
2. Explain the role of parser in the design of compiler.
3. Write CFG for parsers.
4. Learn the techniques used in Recursive descent parser for parsing the input string.
5. Write FIRST and FOLLOW from the given grammar.
6. Learn the techniques used in the Construction of predictive parser.
7. Explain the Error recovery in predictive parser.

Unit 3

Topic: Syntax Analysis

Learning Objective

1. What is compiler?
2. Define Handle Pruning
3. Explain the algorithm of Shift Reduce Parser and its conflicts.
4. Explain an algorithm for Item constructions and goto function in LR
5. Use Top-down Parsing or Bottom-up Parsing in designing parser. To learn the use of Ambiguous grammar in specification and implementation of lang.

Unit 4

Topic: SDT Run-Time Environments

Learning Objective

1. To define Syntax-directed definitions and its forms.
2. To design dependency graph for a given grammar.
3. To explain how topological sort decides the evaluation order.
4. To explain bottom up and top down evaluation of SDD. To explain the need of semantic analysis.
5. To define type checking.
6. To define the specification of a simple type checker.
7. Explain structural and name Equivalence of type expressions.
8. To explain about how compiler convert one data type another data type. To learn benefits of intermediate code generation and its representation.
9. To explain the Translation of expression.
10. To learn about the data structure used for storage during the execution of procedure.
11. To learn about the storage allocation strategies.
12. Definition of symbol table and to explain the construction of symbol table.

Unit 5

Topic: Intermediate Code Generation Code Generation

Learning Objective

1. To discuss some common issues in the design of code generator.
2. To explain the graphical representation of three address statements.
3. To explain implementation of three address statements.
4. To write an algorithm to partition the three address statements into basic blocks.
5. To explain an algorithm for code generation.
6. To learn the data structure used for implementing basic blocks.
7. Examples of translating the statements into syntax tree, postfix notation, three address code.

Review questions

1. What is a compiler?
2. With block diagram explain various phases of a compiler.
3. What is the use of symbol table in the process of compilation?
4. Explain different compiler construction tools.
5. Show how the input pos=20+mid*5 got processed in compiler. Show the output at each phases of compiler.
6. Construct the transition diagram to recognize the following tokens
   i. Identifiers ii. Signed and unsigned numbers. iii. Relational operator.
7. Explain the role of lexical analyzer?
8. Explain the terms tokens, patterns and lexeme.
9. Define the role of input buffer in lexical analysis.
10. Explain the role of parser in compiler model.
11. Differentiate between top down and bottom up parser.
12. What do you mean by left recursion and how it is eliminated?
13. Write and explain rules for finding FIRST and FOLLOW sets.
14. Explain the shift-reduce parsing? Explain with an example.
15. Compare SLR,LALR and LR parser.
16. Consider the following grammar
    E→ E+T | T T→ T*F | F F→ (E) | id Construct LR parser.
17. Explain the algorithm of goto function.
18. Construct a canonical LR(1) parsing for the following grammar.
    E→ T+E | T T→ V*T | T V→ id

19. Show that the following grammar is not SLR(1).
    S→Aa | bAc | dc | bda A→ d
20. Construct SLR table for the following grammar
    S→ CC T→ cC C→ id
21. Explain the concept of syntax directed definition.
22. Explain synthesized attribute and inherited attribute with suitable example.
23. Construct annotated parse tree for the expression (7+5)*2
24. Construct DAG for a*(a+a)+(b*c)+d
25. Explain with an example ordering of the evaluation of attributes.
26. What is type checking? Explain its different rules.
27. What do you mean by type conversion? Explain conversion between primitive data types.
28. Differentiate between static and dynamic type checking.
29. Give an algorithm for structural equivalence of type checking.
30. Explain with an appropriate example how to perform bottom up evaluation of inherited attribute.
31. What do you mean by intermediate code generation? Mention its advantage.
32. Explain the following: i. three address code ii. Quadruples iii. Triples
33. Translate the following into three address code
    int a,b; a=3; b=a+7;
34. Give an intermediate code for procedure call.
35. Generate the three address code for the following code segment.
    for (j=1;j<=10;j++)
      if(x==y) a=b+c;
36. Explain the storage organization with diagram.
37. What is stack allocation of space? Explain activation tree of quick sort program.
38. What are the different storage allocation strategies? Explain briefly.
39. Explain the general activation record structure with its field.
40. Describe the different data structures for symbol table implementation and compare them.
41. What are the common issues in design of code generators?
42. Explain basic block in detail.
43. Write an algorithm for partitioning into blocks.
44. Explain the various issues in designing code generator.
45. Generate code for the following C program
    main()
    { int i,a[10];
      While(i<=10)
        a[i]=0;}
46. Give an intermediate code for procedure call.
47. Consider the following three address code segment
    1. PROD:=0 2. I:=1 3. T4:=4*I 4.T2=addr(A)-4 5. T3:=T2[T1]
    6. T4:=addr(B)-4 7.. T5:=T4[T1] 8. T6:=T3*T5 9. PROD:=PROD+T6 10.I=I+1
    11. if i<=20 goto 3
    Find the basic blocks and flow graph of above sequence.
48. Write an algorithm to construct DAG from given basic block..
49. Write an algorithm to partition a sequence of three address statements into basic blocks.
50. List and explain the structure preserving transformation on basic blocks along with an example.

Lesson Plan

Unit 1

1. Explain the phases of the compilation process.
2. Compilation process contd.. with example
3. Grouping of phases, compiler construction tools .
4. Role of lexical analyzer, tokens, patterns, lexeme, Attributes for tokens Input Buffering.
5. Specification of tokens-strings and languages, operation on languages, regular expression.
6. Regular definition, Notational standards, Recognition of tokens.
7. Writing regular expression for tokens.
8. Examples of writing regular expression.
9. Constructing transition diagram.
10. Implementing transition diagram.

Unit 2

1. The Role of parser- syntax error handling, error recovery strategies, context free grammar.
2. Parse tree derivations, RE vs CFG, elimination of left recursion.
3. Examples for left recursion elimination, algorithm, left factoring.
4. Examples for left factoring, Top Down parsing-Recursive descent parser.
5. Predictive parser- transition diagrams.
6. Nonrecursive predictive parser-algorithm, tracing it with an example.
7. Finding FIRST and FOLLOW, example for the same.
8. Construction of predictive parsing table .
9. Examples for construction of predictive parsing table.
10. Error recovery in predictive parser.

Unit 3

1. Bottom up parsing-example, Handle, handle pruning.
2. Stack implementation of shift reduce parsing.
3. Example of shift reduce parser.
4. Viable prefixes, conflicts during shift reduce parsing.
5. LR parser with algorithm
6. Example for LR parser
7. LR grammars, constructing SLR parsing tables
8. Algorithm for SLR parsing table ,example
9. Construction of canonical LR and LALR parsing table
10. Example of LR and LALR parser, ambiguous grammar.

Unit 4

1. Form of a syntax directed definition, inherited attributes, dependency graph, evaluation order
2. Construction of syntax tree ,Evaluation order of sdd, L -attributed definition.
3. Translation schemes.
4. Type systems, Type expression, error recovery
5. Specification of a simple type checker, equivalence of type expression.
6. Type conversions.
7. Activation tree, control stack, scope of declaration with bindings.
8. Storage organization with activation record.
9. Storage allocation strategies with parameter passing.
10. Symbol tables, dynamic storage allocation.

Unit 5

1. Intermediate languages, declation, assignment statements.
2. Boolean expression, case statement
3. Back patching

4. Procedure calls
5. Issues in design of code generator.
6. Basic blocks
7. Flow graph
8. Simple code generator.
9. Register allocation and assignment.
10. DAG representation of basic block.

G. Course Articulation Matrix (CAM)

Course Outcome (CO)	Program Outcome (ABET/NBA-(3a-m))													
		a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Describe the various phases of a compiler, software tools to build compiler and implement lexical analyzer.	L1,L2	M	L	M	-	-	-	-	-	L	-	M	M	M
Apply an algorithm for top down parsing.	L5	H	H	H	-	-	-	-	-	M	-	M	M	M
Apply an algorithm for bottom-up parsing	L5	H	H	H	-	-	-	-	-	M	-	M	M	M
Write a syntax-directed definition and **Explain** the role of a semantic analyzer , type checking , allocation and deallocation	L2,L6	L	M	M	-	-	-	-	-	M	-	M	M	M
Apply intermediate machine representations and **Analyze** various issues in allocation strategies and design of code generator.	L4,L5	M	H	H	-	-	-	-	-	M	-	M	M	M
L- Low, M- Moderate, H-High														

B. Course Assessment Matrix (CAM)

Course Outcome (CO)	Program Outcome (ABET/NBA-(3a-m))													
		a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Describe the various phases of a compiler, software tools to build compiler and implement lexical analyzer.	L1,L2	2	1	2	-	-	-	-	-	1	-	2	2	2
Apply an algorithm for top down parsing.	L5	3	3	3	-	-	-	-	-	2	-	2	2	2
Apply an algorithm for bottom-up parsing	L5	3	3	3	-	-	-	-	-	2	-	2	2	2
Write a syntax-directed definition and **Explain** the role of a semantic analyzer , type checking , allocation and deallocation	L2,L6	1	2	2	-	-	-	-	-	2	-	2	2	2
Apply intermediate machine representations and **Analyze** various issues in allocation strategies and design of code generator.	L4,L5	2	3	3	-	-	-	-	-	2	-	2	2	2
1- Low, 2- Moderate, 3-High														

Course Title: ADVANCED COMPUTER ARCHITECTURE			
Course Code:P13CS64	**Sem:VI**	**L-T-P-H : 4-0-0-4**	**Credits:4**
Contact Period: Lecture: 52 Hrs., Exam: 3 Hrs.		**Weightage: CIE:50%; SEE:50%**	

Prerequisites:

- Microprocessor.
- Computer Organization.
- Operating System.

Course learning objectives

In this course students should be able to,

1. **Understand** the evolution of computers, choosing the parameters needed to evaluate the performance of architectures, classification of computers to perform multiprocessing, fundamental properties of how parallelism can be introduced in program.
2. **Discuss** the present modern processor technology and the supporting memory hierarchy, Bus for interconnection between different processor, how shared memory concept is used in multiprocessor.
3. **Examine** the basic properties of pipelining, classification of pipeline processors, **plan** solutions for the pipeline processors.
4. **Understand** System architectures of multiprocessor and multicomputer, various cache coherence protocols, synchronization methods, other important concepts involved in building a multicomputer and message passing mechanisms.
5. **Understand** how to perform parallelization of computations of data and acquiring knowledge about scalable multiprocessor systems and different scaling methods.

Course Content

Unit –1

Parallel Computer Models: The State of Computing, Multiprocessor and Multicomputer, Multivector and SIMD Computers.

Program and Network Properties: Conditions of Parallelism, Partitioning and Scheduling, Program flow Mechanisms, System Interconnect Architecture. 10 Hours

Unit – 2

Processor and Memory Hierarchy: Advanced Processor Technology, Design space of processors, Instruction Set Architectures, CISC Scalar Processor (exclude CISC Microprocessor Families) RISC Scalar Processor (exclude Sun Microsystems SPARC Architectures) Superscalar and Vector Processor, Superscalar Processor(exclude IBM Rs/6000 Architecture),VLIW Architecture.

Bus and Shared Memory: Bus Systems, Shared – Memory Organization, Interleaved Memory Organization, Bandwidth and fault Tolerance, Memory Allocation Schemes(exclude swapping in Unix, Demand Paging system and Hybrid Paging system). 10 Hours

Unit – 3

Pipelining and Superscalar Techniques: Linear Pipeline Processors: Asynchronous and Synchronous Models, Clocking and Timing Control, Speed up, Efficiency and Throughput, Non-linear Pipeline Processors: Reservation and Latency Analysis, Collision free scheduling, Pipeline schedule optimizations, Instruction pipeline design: Instruction Execution Phases, Mechanism for Instruction Pipelining, Dynamic Instruction Scheduling, Branch handling Techniques, Arithmetic Pipeline Design: Computer Arithmetic Principles, Static Arithmetic Pipeline, Multifunctional Arithmetic Pipelines (exclude IMB360 Floating Point unit)

11 Hours

Unit – 4

Multiprocessor and Multi-computers: Multiprocessor system Interconnects, Hierarchical Bus Systems, Crossbar Switch and Multiport Memory, Multistage and Combing Networks, Cache Coherence and Synchronization Mechanisms, The Cache Coherence Problem, Snoopy Bus Protocol, Directory based Protocols, Hardware Synchronization Mechanisms, Message Passing mechanisms: Message Routing Schemes, Deadlock and Virtual Channels, Flow Control Strategies**.** 10 Hours

Unit – 5

Parallel Programs: Parallel Application Case Studies: Simulating ocean Currents , Simulating the Evolution of Galaxies, Visualization Complex Scenes using Ray Tracing, Mining Data for Associations, The Parallelization Process: Steps in The Process, Parallelization Computation Versus Data, Goals of the Parallelization Process, Parallelization of an Example Program: The Equation Solver Kernel, Decomposition, Assignment, Orchestration under the Shared address Space Model , Orchestration under the Message – Passing Model.

Scalable Multiprocessors: Scalability, Bandwidth scaling, Latency scaling, Cost Scaling, Physical Scaling, Realizing Programming Model: Primitive Network Transaction, Shared address Space, Message Passing. 11Hours

Text Books:

1. Kai Hwang & Naresh Jotwani, ”Advanced Computer Architecture”, Parallelism, Scalability, Programmability 2nd edition McGraw Hill 2012.
2. David E Culler Jaswinder Pal Singh with Anoop Gupta, “Parallel Computer Architecture” A Hardware/Software Approach, Morgan Kaufmann Publications Elsevier 2012.

Reference Books:

1. John P Hayes, Computer Architecture and Organization 3rd Edition McGraw Hill 1998.
2. V.Rajaraman, C.Siva Ram Murthy, Parallel Computers - Architecture and Programming PHI, 2000.

Course Outcomes

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

1. **Describe** the evolution of computers, classification of computers for multiprocessing, properties to maintain parallelism in a program, **choose** various attributes on which performance of system is measured.
2. **Determine** the present modern processor technology and supporting memory hierarchy, Bus for interconnection between processor, shared memory concept for multiprocessors.
3. **Analyze** the basic properties of pipelining, classification of pipeline processors, **Design** solutions for the linear and non-linear pipeline processors.
4. **Explain** system architectures of multiprocessor and multicomputer, need of cache coherence protocols, synchronization methods and message passing mechanisms.
5. **Describe** the steps to perform parallelization of computations of data and **adapt** knowledge about scalable multiprocessor systems and different scaling methods

Topic Learning Objectives

Unit1

Topic: Parallel Computer Models & Program and Network Properties

Learning Objective

1. Explain the elements of modern computers?(L2)
2. Describe the evolution of computer architecture.(L2)

3. Explain the Flynn's classification of computer architecture.(L2)
4. Explain models of shared memory multiprocessors.(L2)
5. Choose system attributes to performance to find effective CPI, MIPS rate and execution time of processors. (L3)
6. List and explain different types of dependencies.(L1,L2)
7. Analyze Bernstein's conditions to detect parallelism.(L4)
8. Describe partitioning and scheduling of instructions(L2).
9. Differentiate b/w program flow mechanisms.(L4)
10. Explain static and dynamic interconnection networks. (L4)

Unit 2

Topic: Processor and Memory Hierarchy.& Bus and Shared Memory

Learning Objective

1. Analyze design space of processors.(L4)
2. Explain the concept of instruction pipelining.(L2)
3. List and explain the differences between CISC and RISC processors. (L1,L4)
4. Explain the pipeline execution in superscalar processor.(L2)
5. Explain data path architecture and control unit of a base scalar processor.(L4)
6. Describe the architecture and pipeline execution in VLIW processor.(L2)
7. Differentiate b/w VLIW and superscalar processors.(L4)
8. Describe backplane bus specifications.(L2)
9. Analyze two types of interleaved memory organization.(L4)
10. Discuss memory allocation schemes.(L2)

Unit 3

Topic: Pipelining and Superscalar Techniques

Learning Objective

1. Distinguish between linear and non-linear pipeline processors.(L2)
2. Explain the asynchronous and synchronous models of linear pipeline processors.(L2)
3. Describe clocking and timing control.(L2)
4. Write the equations for speed up, efficiency and throughput.(L6)
5. Consider the reservation table with stages and clock cycles find,
   i) Forbidden latencies and permissible latencies.
   ii) Initial collision vector.
   iii) Draw the state transition diagram
   iv) Find the latency, greedy and simple cycles.
   v) Find the minimum average latency.(L5)
6. Explain the mechanisms for instruction pipelining.(L2)
7. Illustrate the impact of branch handling techniques.(L3)
8. Explain arithmetic pipeline design with example.(L4)
9. Distinguish b/w carry-propagate adder (CPA) and a carry-save adder (CSA).(L4)
10. Explain dynamic instruction scheduling using Tomasulo's approach.(L4)

Unit 4

Topic: Multiprocessor and Multi-computers

Learning Objective

1. Explain hierarchical bus systems.(L2)
2. Explain the designing of crossbar switch and multiport memory.(L4)
3. Construct omega network to route the data from source to destination and also Explain the Routing in Butterfly Networks.(L3,L4)
4. Define cache coherence problem? Explain the 3 factors causing cache coherence.(L1,L2)
5. Describe snoopy based cache coherence protocol.(L2)
6. Describe directory-based cache coherence protocol.(L2)

7. Explain the hardware synchronization mechanisms.(L4)
8. Explain store-forward and wormhole message routing schemes.(L4)
9. Explain deadlock and virtual channels.(L4)
10. Analyze flow control strategies& describe dimension-order routing methods.(L4,L2)

Unit 5

Topic: Parallel Programs & Scalable Multiprocessors

Learning Objective

1. List and explain different steps in the parallelization process.(L1,L2)
2. List and explain the goals of parallelization process.(L1,L2)
3. Explain simulating ocean currents and simulating the evolution of galaxies.(L4)
4. Analyze visualization complex scenes using Ray Tracing and Describe mining data for associations.(L4,L2)
5. Develop parallelization of an example program the equation solver kernel.(L6)
6. Explain orchestration under the shared address space model and message passing model.(L4)
7. Define scalability and Explain different scaling methods.(L1,L4)
8. Describe Primitive Network Transactions?(L2)
9. Analyze the shared address space.(L4)
10. Explain synchronous and asynchronous message passing methods. (L4)

Review Questions

Unit - 1

1. Explain the elements of modern computers?
2. Describe the evolution of computer architecture.
3. Explain the Flynn's classification of computer architecture.
4. Explain models of shared memory multiprocessors.
5. Explain system attributes to performance to find effective CPI, MIPS rate and execution time of processors.
6. List and explain different types of dependencies.
7. Explain Bernstein's conditions to detect parallelism.
8. Describe partitioning and scheduling of instructions.
9. Differentiate b/w program flow mechanisms.
10. Explain static and dynamic interconnection networks.

Unit – 2

1. Explain design space of processors.
2. Explain the concept of instruction pipelining.
3. List and explain the differences between CISC and RISC processors.
4. Explain the pipeline execution in superscalar processor.
5. Explain data path architecture and control unit of a base scalar processor.
6. Explain the architecture and pipeline execution in VLIW processor.
7. Explain the differences b/w VLIW and superscalar processors.
8. Describe backplane bus specifications.
9. Explain two types of interleaved memory organization.
10. Discuss memory allocation schemes.

Unit – 3

1. Explain the difference between linear and non- linear pipeline processors.
2. Explain the asynchronous and synchronous models of linear pipeline processors.
3. Describe clocking and timing control.
4. Derive the equations for speed up, efficiency and throughput.
5. Given the reservation table with stages and clock cycles find,
   i.)Forbidden latencies and permissible latencies.
   ii.)Initial collision vector.

iii.)Draw the state transition diagram
iv.)Find the latency, greedy and simple cycles.
v.) Find the minimum average latency.

6. Explain the mechanisms for instruction pipelining.
7. Explain branch handling techniques.
8. Explain arithmetic pipeline design with example.
9. Distinguish b/w carry-propagate adder(CPA) and a carry-save adder(CSA).
10. Explain dynamic instruction scheduling using Tomasulo's approach.

Unit – 4

1. Explain hierarchical bus systems.
2. Explain the designing of crossbar switch and multiport memory.
3. Draw omega network to route the data from source to destination and also Explain the Routing in Butterfly Networks.
4. What is cache coherence problem? Explain the 3 factors causing cache coherence.
5. Explain snoopy based cache coherence protocol.
6. Explain directory-based cache coherence protocol.
7. Explain the hardware synchronization mechanisms.
8. Explain store-forward and wormhole message routing schemes.
9. Explain deadlock and virtual channels.
10. Explain flow control strategies and dimension-order routing methods.

Unit - 5

1. Steps in the parallelization process.
2. Goals of parallelization process.
3. Simulating ocean currents and simulating the evolution of galaxies.
4. Visualization complex scenes using Ray Tracing mining data for associations.
5. Mining data for associations
6. Parallelization of an example program the equation solver kernel.
7. Orchestration under the shared address space model and message passing model.
8. Concept of scaling and different scaling methods.
9. Primitive Network Transactions.
10. Concept of shared address space.
11. Synchronous and asynchronous message passing methods.

Lesson Plan

Unit 1

1. Understanding elements of modern computers.
2. Describe the evolution of computer architecture.
3. Flynn's classification of computer architecture.
4. Models of shared memory multiprocessors.
5. System attributes to performance to find effective CPI, MIPS rate and execution time of processors.
6. Different types of dependencies.
7. Bernstein's conditions to detect parallelism.
8. Partitioning and scheduling of instructions.
9. Differentiate b/w program flow mechanisms.
10. Importance of static and dynamic interconnection networks.

Unit 2

1. Introduction to Design space of processors.
2. Concept of instruction pipelining.
3. Differences between CISC and RISC processors.
4. Pipeline execution in superscalar processor.
5. Data path architecture and control unit of a base scalar processor.
6. Understanding architecture and pipeline execution in VLIW processor.

7. Differences b/w VLIW and superscalar processors.
8. Backplane bus specifications.
9. Two types of interleaved memory organization.
10. Memory allocation schemes.

Unit 3

1. Difference between linear and non-linear pipeline processors.
2. Asynchronous and Synchronous models of linear pipeline processors.
3. Clocking and timing control.
4. Derive the equations for speed up, efficiency and throughput.
5. Given the reservation table with stages and clock cycles find,
   i.)Forbidden latencies and permissible latencies.
   ii.)Initial collision vector.
   iii.)Draw the state transition diagram
   iv.)Find the latency, greedy and simple cycles.
   v.) Find the minimum average latency.
6. Mechanisms for instruction pipelining.
7. Importance of Branch handling techniques.
8. Arithmetic pipeline design with example.
9. Distinguish b/w carry-propagate adder(CPA) and a carry-save adder(CSA).
10. Dynamic instruction scheduling using Tomasulo's approach.

Unit 4

1. Hierarchical bus systems.
2. Designing of crossbar switch and multiport memory.
3. Draw omega network to route the data from source to destination.
4. Routing in Butterfly Networks.
5. Cache coherence problem and 3 factors causing cache coherence.
6. Snoopy based cache coherence protocol.
7. Directory-based cache coherence protocol.
8. Hardware synchronization mechanisms.
9. Store-forward and wormhole message routing schemes.
10. Deadlock and virtual channels.
11. Flow control strategies and dimension-order routing methods.

Unit 5

1. Steps in the parallelization process.
2. Goals of parallelization process.
3. Simulating ocean currents and simulating the evolution of galaxies.
4. Visualization complex scenes using Ray Tracing mining data for associations.
5. Mining data for associations
6. Parallelization of an example program the equation solver kernel.
7. Orchestration under the shared address space model and message passing model.
8. Concept of scaling and different scaling methods.
9. Primitive Network Transactions.
10. Concept of shared address space.
11. Synchronous and asynchronous message passing methods.

A. Course Articulation Matrix (CAM)

Course Outcome (CO)	Blooms levels	Program Outcome (ABET/NBA-(3a-m))												
		a	b	c	d	e	f	g	h	i	j	k	l	m
Describe the evolution of computers, classification of computers for multiprocessing, properties to maintain parallelism in a program, **choose** various attributes on which performance of system is measured.	**L2, L5**	H	H	H	-	H	-	-	-	L	-	H	M	L
Determine the present modern processor technology and supporting memory hierarchy, Bus for interconnection between processor, shared memory concept for multiprocessors.	**L5**	H	H	H	-	M	-	-	-	-	-	H	M	-
Analyze the basic properties of pipelining, classification of pipeline processors, **Design** solutions for the linear and non-linear pipeline processors.	**L4, L6**	H	H	H	-	M	-	-	-	-	-	H	M	-
Explain system architectures of multiprocessor and multicomputer, need of cache coherence protocols, synchronization methods and message passing mechanisms.	**L2**	M	H	H	-	H	-	-	-	-	-	H	L	M
Describe the steps to perform parallelization of computations of data and **adapt** knowledge about scalable multiprocessor systems and different scaling methods.	**L6, L3**	H	H	H	-	M	-	-	-	-	-	H	H	-
L- Low, M- Moderate, H-High														

B. Course Assessment Matrix (CaM)

Course Outcome (CO)	Blooms levels	Program Outcome (ABET/NBA-(3a-m))												
		a	b	c	d	e	f	g	h	i	j	k	l	m
Describe the evolution of computers, classification of computers for multiprocessing, properties to maintain parallelism in a program, **choose** various attributes on which performance of system is measured.	**L2, L5**	3	3	3	-	3	-	-	-	1	-	3	2	1
Determine the present modern processor technology and supporting memory hierarchy, Bus for interconnection between processor, shared memory concept for multiprocessors.	**L5**	3	3	3	-	2	-	-	-	-	-	3	2	-
Analyze the basic properties of pipelining, classification of pipeline processors, **Design** solutions for the linear and non-linear pipeline processors.	**L4, L6**	3	3	3	-	2	-	-	-	-	-	3	2	-
Explain system architectures of multiprocessor and multicomputer, need of cache coherence protocols, synchronization methods and message passing mechanisms.	**L2**	2	3	3	-	3	-	-	-	-	-	3	1	2
Describe the steps to perform parallelization of computations of data and **adapt** knowledge about scalable multiprocessor systems and different scaling methods.	**L6, L3**	3	3	3	-	2	-	-	-	-	-	3	3	-
1- Low, 2- Moderate, 3-High														

Course Title : Software Engineering			
Course Code: P13CS65	**Semester : VI**	**L-T-P-H: 2-2-0-4**	**Credit:3**
Contact Period : Lecture :52 Hrs., Exam: 3Hrs.		**Weightage :CIE:50% SEE:50%**	

Prerequisites

- Software development and object oriented concepts

Course Objectives

This course aims to:

1. Introduction to Software Engineering .
2. Describe the process of Software Engineering, the technologies used for Software Engineering, and configuration management of Software Engineering.
3. Apply Architectural Design Architectural design decisions System organization Modular decomposition styles Control styles.
4. Understand what Software Testing is.
5. Explain Project management Risk management, Managing people, Teamwork, Understand Configuration management

Course content

Unit – 1

OVERVIEW: Introduction to Software Engineering ,Introduction, Professional software development ,Software engineering ethics, Case studies.

Software processes: Software process models, Process activities, Coping with change, The Rational Unified Process. 8 Hours

Unit – 2

Agile software development: Agile methods, Plan driven and agile development, Extreme programming, Agile project management, Scaling agile methods

Requirements engineering:

Functional and non-functional requirements, The software requirements document Requirements specification, Requirements engineering processes, Requirements elicitation and analysis, Requirements validation, Requirements management. 12 Hours

Unit – 3

System modelling: Context models, Interaction models, Structural models, Behavioral models, Model-driven engineering.

Architectural design: Architectural design decisions, Architectural views, Architectural patterns, Application architectures. 12 Hours

Unit – 4

Design and Implementation: Object-oriented design using the UML Design patterns, Implementation issues, Open source development

Software testing: Development testing, Test-driven development, Release testing, User testing. 10 Hours

Unit – 5

Project management: Risk management, Managing people, Teamwork.

Configuration management: Change management, Version management System building, Release management. 8 Hours

Text book:

1. Software Engineering – Ian Somerville, 9th Edition, Pearson Education, 2012.

Reference books :

1. Software Engineering: A Practitioners Approach - Roger S. Pressman, 7th Edition, McGraw-Hill, 2007.
2. Software Engineering Theory and Practice - Shari Lawrence Pfleeger, Joanne M. Atlee, 3rd Edition, Pearson Education, 2006.
3. Software Engineering Principles and Practice – Waman S Jawadekar, Tata McGraw Hill, 2004

4. Software Engineering – Pankaj Jalote, Tata McGraw Hill.

Course Outcomes

At the end of the course the student should be able

1. Describe the various types of software process.
2. Know the importance of software development.
3. Explain significance of software engineering.
4. Compare different Software Development methods.
5. Able to identify the different forms of Software Development.

Topic learning objectives

Unit 1

1. Define Software Engineering.L1
2. Define the basics of the various Software process.L1
3. Explain the applications of Software process.L2
4. Describe Agile project management.L2
5. Define Objects of requirement analysis.L1
6. Explain different key requirement engineering process.L2
7. Describe the functional and non functional requirements. L2
8. Describe Rational Unified Process.L1
9. Explain the Process activities in requirement engineering.L2
10. Explain how to Cope with change in requirement.L2

Unit 2

1. Describe Agile software development. L1
2. Explain Plan driven and agile development.L2
3. Explain Extreme programming in Software Engineering .L3
4. Describe some important formats of graphics and images.L2
5. Explain the Scaling in agile methods.L2
6. Describe Functional and non-functional requirements.L2
7. Describe The software requirements document Requirement specification,
8. Requirements engineering processes.L2
9. Explain the basic elicitation and analysis.L2
10. Describe the Requirements validation.L2
11. Explain the Requirements management. L2

Unit 3

1. Understand the Context models and Interaction models. L1
2. Distinguish the Structural models and Behavioural models.L3
3. Describe the Model-driven engineering.L2
4. Explain few basic Architectural design.L2
5. Describe the Architectural design decisions.L2
6. Describe the Architectural views and Architectural patterns. L2
7. Describe the Application architectures.L2
8. Explain the architectural perspectives.L2
9. Explain the Release testing.L2
10. Describe the architecture for the system.L2

Unit 4

1. Describe Object-oriented design using the UML.L1
2. Describe the basic Design in software engineering. L2
3. Explain briefly Design patterns. L2
4. Explain the basic Implementation issues. L2
5. Describe how the Software testing is done. L2
6. Explain Open source development. L2
7. Explain Development testing. L2

8. Explain Test-driven development. L2
9. Explain User testing. L2
10. Compare different testing. L2

Unit 5

1. Distinguish Risk management and Managing people.L2
2. Understand Project management. L3
3. Explain Configuration management. L2
4. Understand Change management. L2
5. Explain Version management. L2
6. Describe System building. L2
7. Explain Release management L2
8. Explain managing the project. L2
9. Describe project analysis. L2
10. Describe system analysis for project. L3

Review questions

1. What is Software Engineering?
2. Discuss the various Software process.
3. Bring out the applications of Software process.
4. Describe in detail agile project management.
5. Explain the Objects of requirement analysis.
6. Explain different key requirement engineering process.
7. Describe in detail the functional and non functional requirements.
8. Explain the Rational Unified Process in detail.
9. Explain the Process activities in requirement engineering.
10. Explain how to cope with change in requirement.
11. Describe agile software development.
12. Explain Plan driven and agile development.
13. Explain Extreme programming in Software Engineering .
14. Describe some important formats of graphics and images.
15. Explain the Scaling in agile methods.
16. Describe Functional and non-functional requirements.
17. Describe The software requirements document Requirement specification,
18. Requirements engineering processes.
19. Explain the basic elicitation and analysis.
20. Describe the Requirements validation.
21. Explain the Requirements management.
22. Differentiate the Context models and Interaction models.
23. Describe the Structural models and Behavioral models.
24. Describe the Model-driven engineering.
25. Explain the basic Architectural design.
26. Describe the Architectural design decisions.
27. Describe the Architectural views and Architectural patterns.
28. Describe the Application architectures.
29. Explain the architectural perspectives.
30. Explain the Release testing.
31. Describe the architecture for the system.
32. Describe Object-oriented design using the UML.
33. Describe the basic Design in software engineering.
34. Explain briefly Design patterns.
35. Explain the basic Implementation issues.
36. Describe how the Software testing is done.
37. Explain Open source development.

38. Explain in detail Development testing.
39. Explain Test-driven development.
40. Explain User testing.
41. Compare different testing.
42. Distinguish Risk management and Managing people.
43. Discuss in detail what Project management is.
44. Explain Configuration management .
45. Explain Change management.
46. Explain Version management.
47. Describe System building.
48. Explain in detail Release management.
49. Explain managing the project in detail.
50. Describe in detail project analysis.
51. Describe system analysis for project.

Lesson plan

Unit 1

1. Overview**,** Introduction to Software Engineering.
2. Professional software development
3. Software engineering ethics
4. Case studies.
5. Software processes, Software process models
6. Process activities
7. Coping with change
8. The Rational Unified Process.

Unit 2

1. Agile software development Agile methods.
2. Plan driven and agile development, Extreme programming
3. Agile project management.
4. Scaling agile methods.
5. Requirements engineering.
6. Functional and non-functional requirements.
7. The software requirements document.
8. Requirements specification , Requirements engineering processes.
9. Requirements elicitation and analysis.
10. Requirements validation.
11. Requirements management.
12. Requirements management.

Unit 3

1. System modelling
2. Context models, Interaction models
3. Structural models, Behavioural models
4. Model-driven engineering.
5. Architectural design
6. Architectural design decisions
7. Architectural design decisions
8. Architectural design decisions
9. Architectural, views
10. Architectural patterns
11. Architectural patterns
12. Application architectures

Unit 4

1. Design and Implementation
2. Object-oriented design using the UML
3. Design patterns
4. Implementation issues
5. Open source development
6. Software testing
7. Development testing,
8. Test-driven development
9. Release testing
10. User testing

Unit 5

1. Project management
2. Risk management
3. Managing people
4. Teamwork
5. Configuration management
6. Change management, Version management
7. System building
8. Release management

Course Articulation Matrix (CAM)														
Course Outcome (CO)	**Program Outcome (ABET/NBA-(3a-m))**													
		a	**b**	**C**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Describe the various types of software process	**L1, L2**		L				M	H	L	M		M		
Know the importance of software development.	**L2**		M	L			M	M	L	M		M		
Explain significance of software engineering.	**L2**		M	M		M	M	H	L	M		M	L	L
Compare different Software Development methods.	**L2**				M		M	M		M				
Able to identify the different forms of Software Development	**L2, L3**		M		M	M	M	M		M		M	M	L
L- Low, M- Moderate, H-High														

Course Assessment Matrix (CAM)														
Course Outcome (CO)	**Program Outcome (ABET/NBA-(3a-m))**													
		a	**b**	**C**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Describe the various types of software process	**L1, L2**		L				M	H	L	M		M		
Know the importance of software development.	**L2**		M	L			M	M	L	M		M		
Explain significance of software engineering.	**L2**		M	M		M	M	H	L	M		M	L	L
Compare different Software Development methods.	**L2**				M		M	M		M				
Able to identify the different forms of Software Development	**L2, L3**		M		M	M	M	M		M		M	M	L
L- Low, M- Moderate, H-High														

Course Title : Computer Graphics and Visualization Lab			
Course Code: P13CSL67	**Semester : VI**	**L-T-P-H: 0-1-2-3**	**Credits:1.5**
Contact Period : Lecture :36 Hrs., Exam: 3Hrs.		**Weightage :CIE:50% SEE:50%**	

Prerequisites

- Computer graphics and visualization

Course Objectives

1. To acquire knowledge about the basic concepts of 2D and 3D graphic functions.
2. To implement various transformation and clipping techniques.
3. Demonstrate menu driven program, mouse and keyboard handling function.
4. To apply lighting and shading techniques in creation of 2D or 3D images.

PART-A

Introduction to OpenGL Programming

1. Write a Hello world application to understand working with IDE.
2. Write a program to initialize OpenGL and display some basic geometric shapes. Example, line, polygon etc.
3. Program to recursively subdivide a tetrahedron to from 3D Sierpinski gasket. The number of recursive steps is to be specified by the user.

Geometric Transformations

4. Program to implement basic 2D transformations.
5. Program to implement basic 3D transformations.
6. Program to implement wireframe drawing of various primitive shapes.

Drawing of line, clipping algorithm

7. Program to draw a line using Bresenham's line drawing algorithm.
8. Program to implement the Cohen-Sutherland line-clipping algorithm. Make provision to specify the input line, window for clipping and view port for displaying the clipped image.
9. Program to clip a polygon using polygon clipping algorithm.
10. Program to fill any given polygon using scan-line area filling algorithm.

Input and Interaction

11. Program to demonstrate the creation of menu, mouse handling and keyboard handling.
12. Program to implement picking using selection mode.

Lighting & shading and curves &surfaces

13. Program to draw the shaded scene.
14. Program to implement the Bezier curves.

PART-B

Develop a suitable simple graphics package using OpenGL to implement the skills learnt in the theory and lab.

Course Outcomes

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

1. Understand and explain the mathematical and theoretical principles of computer graphics eg: To draw basic objects like lines, triangles and polygons using opengl built-in functions
2. Use matrix algebra in computer graphics and implement fundamental algorithms and transformations involved in viewing models.
3. Write complete graphics software systems projection models, illumination models and handling of hidden surfaces and clipping in computer graphics

4. Analyze and evaluate the use of computer graphics methods in practical applications and describe effects such as texture mapping, bump mapping
5. Apply computer graphics techniques to creating aesthetic effect

Note: 1) Program 1 and 2 are only for practice not to be included in the SEE.
2) A report of about 10-12 pages on the package developed in PART-B, duly certified by the guide must be submitted during examination.

Instructions:
In the examination one exercise from PART-A is to be asked for a total of 30 marks. The package developed under PART-B has to be evaluated for total of 20 marks.

Course Title : OPERATING SYSTEM LAB			
Course Code: P13CSL68	**Semester : VI**	**L-T-P-H: 0-1-2-3**	**Credit: 1.5**
Contact Period : Lecture : 36Hrs., Exam: 3Hrs.		**Weightage :CIE:50% SEE:50%**	

Prerequisites

- C or C++ programming language
- Operating System

Course objectives

Students will be able to

1. **Compute** the average waiting time and average turn around time of different processor scheduling algorithms.
2. **Acquire** the knowledge of solving different synchronization problems.
3. **Determine** whether the system is in deadlock or safe state, for the given the number available, allocated and requested resources, number of process.
4. **Select** the best memory allocation algorithms, given the available memory, allocated memory and the processes memory requirement.
5. **Analyze** the storage of data in disks.

Process Scheduling

1. Given the list of processes and their CPU burst times, write a program to display the Gantt chart for FCFS . Also compute and print the average waiting time and average turnaround time.
2. Given the list of processes, their CPU burst times and arrival times, write a program to display the Gantt chart for SJF. Also compute and print the average waiting time and average turnaround time.
3. Given the list of processes, their CPU burst times and time slice, write a program to display the Gantt chart for Round robin scheduling policy. Also compute and print the average waiting time and average turnaround time.
4. Given the list of processes, their CPU burst times and priorities, write a program to display the Gantt chart for Priority scheduling policy. Also compute and print the average waiting time and average turnaround time.

Synchronization

5. Write a program that demonstrates how two processes can share a variable using semaphore.
6. Implement Producer–Consumer problem using semaphores (using UNIX system calls).
7. Implement Dinning Philosopher's problem using monitors.

Deadlock

8. Implement Banker's Deadlock Avoidance algorithm for multiple resources.

Memory management

9. Implement the following memory allocation techniques.
   FIRST FIT
   BEST FIT

WORST FIT

10. Implement the following page replacement algorithms

FIFO page replacement
LRU page replacement
OPT page replacement

Storage management

11. Implement the following Disk scheduling algorithm

FCFS
SSTF
C-SCAN

Course outcomes

Students will be able to

1. **Implement** and **Compare** the different algorithms for CPU Scheduling.
2. **Implement** algorithms for handling synchronization.
3. **Write** the program to check whether the system is in deadlock or safe state.
4. **Implement** memory management algorithms.
5. **Show** the data storage using disk scheduling algorithms.

Course Title : **Data Warehousing and Data Mining**			
Course Code: P13CS661	**Semester : VI**	**L-T-P-H: 4-0-0-4**	**Credits:4**
Contact Period : Lecture :52 Hrs., Exam: 3Hrs.		**Weightage :CIE:50% SEE:50%**	

Prerequisites:

- Data base management system

Course Objective

1. This course is intended to provide an exposure to fundamental concepts of ODS, Data warehouses, OLAP and data cube.
2. The goal is to present fundamental concepts of data mining and data preprocessing techniques
3. To provide knowledge about concepts and algorithms for classification
4. To provide knowledge about algorithms for association analysis
5. To learn concepts and techniques for cluster analysis.

Course content

Unit-1

Data Warehousing: Introduction, Operational Data Strores (ODS), Extraction Transformation Loading(ETL), Data Warehouses, Design issues, Guidelines for Data Warehouse Implementation, Data Warehouse metadata

Online Analytical Processing(OLAP): Introduction, Characteristics of OLAP systems, Multidimensional view and Data cube, Data cube implementations, Data cube operations, Implementation of OLAP 10 Hours

Unit-2

Data Mining. Introduction, Motivating challenges, Origins of data mining, Data mining tasks, Types of data, Data quality, Data preprocessing, Measures of similarity and dissimilarity 10 Hours

Unit-3

Association Analysis: Problem definition, Frequent itemset generation, Rule generation, Compact representation of Frequent itemsets, Alternative method for generating Frequent itemset, FP-growth algorithm 11 Hours

Unit-4

Classification: Basics, General approach to solving a classification problem, Decision tree induction, Rule based classifier, Nearest Neighbor classifiers, Bayesian classifier. 11 Hours

Unit-5

Cluster analysis: Overview, K-means, Agglomerative Hierarchical clustering, DBSCAN
Web data mining : Web terminology and characteristics, Web content mining, Web usage mining, Web structure mining. 10 Hours

TEXT BOOK:

1. Introduction to Data Mining – Pang-Ning Tan, Vipin Kumar, Michael Steinbach, Pearson education
2. Introduction to Data Mining with case studies – G K Guptha, PHI

REFERENCE BOOKS:

1. Data Mining Techniques - Arun K Pujari, Universities press 2009
2. Data mining concepts and techniques – Jiawei and Micheline Kamber, Morgan Kauffmann publisher
3. Data warehousing and data mining , Alex Berson and Stephen J Smith, Mc Grawhill publisher 1997

Course Outcome

1. **Explain** the fundamental concepts of ODS, Data warehouse, ETL, OLAP, Data cube.
2. **Explain** data preprocessing techniques and calculate SMC, J, cosine, correlation for the given data set.
3. **Apply** apriori algorithm and FP-growth algorithm for frequent item set generation.
4. **Apply** classification algorithms to classify a given test record.
5. **Explain** clustering techniques to cluster the data objects.

Topic Learning Objectives

Unit-1

Data Warehousing: Introduction, Operational Data Strores(ODS), Extraction Transformation Loading(ETL), Data Warehouses, Design issues, Guidelines for Data Warehouse Implementation, Data Warehouse metadata
Online Analytical Processing(OLAP): Introduction, Characteristics of OLAP systems, Multidimensional view and Data cube, Data cube implementations, Data cube operations, Implementation of OLAP
At the end of this chapter student should be able to:

1. Define and describe an ODS(L1)
2. Describe the data extraction, transformation and loading process(L2)
3. Define and describe a data warehouse(L1).
4. Describe data warehouse modeling and implementation(L2)
5. Explain the characteristics of OLAP systems(L2)
6. Explain the concept of a data cube(L2).
7. Explain the implementations and operations on data cube.(L2)

Unit-2

Data Mining. Introduction, Motivating challenges, Origins of data mining, Data mining tasks, Types of data, Data quality, Data preprocessing, Measures of similarity and dissimilarity
At the end this chapter student should be able to:

1. Explain the concept of data mining, motivating challenges, data mining tasks with examples(L2)

2. Explain different types of attributes, data sets(L2)
3. Explain different types of data preprocessing techniques(L2)
4. Explain the concept of similarity and dissimilarity measures for different types of attributes(L2)
5. Compute SMC, Jacard coefficient, cosine and correlation for the given data(L3)

Unit – 3

Association Analysis: Problem definition, Frequent itemset generation, Rule generation, Compact representation of Frequent itemsets, Alternative method for generating Frequent itemset, FP-growth algorithm

At the end this chapter student should be able to:

1. Explain the basic concept of Association Analysis(L2)
2. Apply apriori algorithm to generate frequent item sets(L3)
3. Apply apriori algorithm for rule generation(L3)
4. Explain maximal frequent item sets and closed frequent item sets from the given items(L2)
5. Explain the concept of FP-tree(L2)
6. Apply FP-growth algorithm for frequent item set generation(L3)

Unit - 4

Classification: Basics, General approach to solving a classification problem, Decision tree induction, Rule based classifier, Nearest Neighbor classifiers, Bayesian classifier

At the end this chapter student should be able to:

1. Explain the basic concept of classification and explain the general approach to solving a classification problem(L2)
2. Apply Hunt's algorithm for decision tree induction(L3)
3. Explain the characteristics of decision trees(L2)
4. Explain the concept of rule based classifier, accuracy and coverage(L2)
5. Explain how a rule based classifier works(L2)
6. Apply sequential covering algorithm for rule extraction(L3)
7. Compute the evaluation metrics for the given set of rules(L3)
8. Explain K-nearest neighbor classification algorithm (L2)
9. Explain how Bayes theorem is used for classification(L2)
10. Compute the posterior probability to classify a given test record (L3)

Unit – 5

Cluster analysis: Overview, K-means, Agglomerative Hierarchical clustering, DBSCAN

Web data mining : Web terminology and characteristics, Web content mining, Web usage mining, Web structure mining

At the end this chapter student should be able to:

1. Explain cluster analysis, different types of clustering and types of clusters(L2)
2. Explain the basic K-means algorithm for clustering(L2)
3. Explain the basic Agglomerative Hierarchical clustering algorithm(L2)
4. Explain various Hierarchical clustering algorithm(L2)
5. Explain DBSCAN algorithm(L2)
6. Define the relevant web terminology(L2)
7. Explain what web content mining is and how it may be used for discovering useful information(L2)
8. Analyze the structure of the web using the HITS algorithm designed for web structure mining(L2)

Review questions

Unit 1

1. Define ODS. Explain the architecture of ODS with a diagram
2. Define data warehouse. Distinguish between ODS and Data warehouse
3. With the help of a block diagram, explain the architecture of Data warehouse.
4. Explain the problems in building an integrated data base from multiple sources.
5. Explain the implementation steps for a data warehouse
6. What is OLAP? Explain the characteristics of OLAP.
7. Write a note on data cube implementation.
8. Explain different operations applied on datacube.
9. Compare OLTP and OLAP systems.
10. Explain the guidelines for OLAP implementation.

Unit 2

1. What is Data Mining? Explain the challenges that motivated the development of Data Mining
2. What is KDD? Explain the process of knowledge discovery in databases.
3. Explain the different types of data mining tasks with example.
4. What properties of numbers are used to describe the attributes? Explain the different types of attributes based on these properties
5. What are the general characteristics of data sets? Explain any three types of data sets.
6. What is the process of data cleaning? Explain the different types of data errors.
7. List the preprocessing techniques used in Data Mining. Explain any three data preprocessing techniques.
8. What are the different approaches for feature subset selection? Explain the architecture for feature subset selection.
9. For the following vectors X and Y, calculate the indicated similarity measure.
   a. X= (0, 1, 0, 1) Y= (1, 0, 1 0) (Cosine, Correlation)
10. What is variable transformation? Explain the two types of variable transformation

Unit 3

1. What is association analysis? Explain the terms support and confidence.
2. Write the apriori algorithm for frequent item set generation.
3. For the following data set, find all the frequent item sets using apriori principle.

TID	Item sets
T100	K, A, D, B
T200	D, A, C, E, B
T300	C, A, B, E
T400	B, A, D

4. What is support counting? Construct the hash tree for the transtation T={1,3,4,5,8}.
5. Define maximal frequent item sets and closed frequent item sets.
6. What are the two steps in FP-growth algorithm. Construct FP tree for the above data set.

.Unit 4

1. Define classification. Explain the general approach for solving the classification problem with a neat diagram.
2. What is a decision tree? Explain Hunt's algorithm for decision tree induction.
3. Explain the different methods for expressing attribute test conditions in decision tree.
4. Explain the characteristics of decision tree classifier.
5. What is rule based classifier? Write and explain the sequential covering algorithm.
6. Explain the different approaches used for rule evaluation.
7. Explain, how rules are extracted from indirect method with an example.
8. State Bayes theorem. How Bayes theorem is used for classification.

9. Explain K-nearest neighbor classification algorithm.
10. Explain the characteristics of rule based classifier.

Unit 5

1. What is cluster analysis? Explain different types of clustering.
2. Explain different types of clusters.
3. Write and Explain K-means clustering algorithm.
4. Write and Explain basic agglomerative hierarchical technique.
5. Explain the different types of agglomerative hierarchical technique
6. What is Web data mining? Explain the differences between the three types of web data mining.
7. List the major differences between conventional searching and web searching.
8. Explain DIPRE algorithm..
9. How can web pages are clustered. Explain STC algorithm.
10. What data mining techniques can be used for web log data analysis.

Lesson plan

Unit 1

1. Data Warehousing**:** Introduction,
2. Operational Data Strores(ODS)
3. Extraction Transformation Loading(ETL)
4. Data Warehouses, Design issues,
5. Guidelines for Data Warehouse Implementation, Data Warehouse metadata
6. Online Analytical Processing(OLAP): Introduction, Characteristics of OLAP systems,
7. Multidimensional view and Data cube,
8. Data cube implementations,
9. Data cube operations,
10. Implementation of OLAP

Unit 2

1. Data Mining. Introduction, Motivating challenges,
2. Origins of data mining, Data mining tasks,
3. Types of data, attributes and measurement
4. Types of data sets
5. Data quality, Data preprocessing, Aggregation, sampling, dimensionality reduction
6. Feature subset selection, feature creation, discretization and binarization
7. Variable transformation, Measures of similarity and dissimilarity
8. Similarity and dissimilarity between simple attributes
9. Dissimilarity between data objects
10. Examples of proximity measures

Unit 3

1. Association Analysis**:** Problem definition,
2. Frequent itemset generation in the apriori algorithm
3. Solving problems using apriori algorithm
4. Candidate generation and pruning,
5. support counting, Rule generation
6. Compact representation of Frequent itemsets, maximal frequent item sets, closed frequent item sets
7. Alternative method for generating Frequent itemset,
8. FP-growth algorithm, FP-tree representation
9. Frequent itemset generation in FP-growth algorithm
10. Solving problems using FP-growth algorithm

Unit 4

1. Classification: Basics, General approach to solving a classification problem,
2. Decision tree induction, algorithm for decision tree induction

3. Methods for expressing attribute test condition
4. Measures for selecting best split
5. Rule based classifier, rule ordering schemes
6. Direct methods for rule extraction
7. Nearest Neighbor classifiers, algorithm
8. Bayesian classifier, Bayes theorem
9. Using Bayes theorem for classification,
10. Naïve Bayes classifier
11. Bayes error rate, Bayesian Belief networks

Unit 5

1. Cluster analysis: Overview, different types of clusterings, different types of clusters
2. K-means algorithm
3. Bisecting K-means algorithm
4. K-means and different types of clusters
5. Agglomerative Hierarchical clustering,
6. DBSCAN
7. Web data mining **:** Web terminology and characteristics,
8. Web content mining,
9. Web usage mining,
10. Web structure mining

Course Articulation Matrix

H. Course Articulation Matrix (CAM)														
Course Outcome (CO)	**Program Outcome**													
		a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Explain the fundamental concepts of ODS, Data warehouse, ETL, OLAP, Data cube.	**L2**	M	M	M	M	H	-	-	-	-	-	M	M	M
Explain different types of data sets, attributes, preprocessing techniques	**L2**	M	M	M	M	-	-	-	-	-	-	M	M	M
Apply apriori algorithm and FP-growth algorithm for frequent item set generation.	**L3**	M	M	M	M	-	-	-	-	-	-	M	M	M
Apply classification algorithms to classify a given test record	**L3**	M	M	M	M	-	-	-	-	-	-	M	M	M
Explain clustering techniques to cluster the data objects.	**L2**	M	M	M	M	-	-	-	-	-	-	M	M	M
L- Low, M- Moderate, H-High														

Course Assessment Matrix (CaM)														
Course Outcome (CO)	**Program Outcome**													
		a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Explain the fundamental concepts of ODS, Data warehouse, ETL, OLAP, Data cube.	**L2 ,**	2	2	2	2	3	-	-	-	-	-	2	2	2
Explain different types of data sets, attributes, preprocessing techniques	**L2**	-	2	2	2	-	-	-	-	-	-	2	2	2
Apply apriori algorithm and FP-growth algorithm for frequent item set generation.	**L3**	2	2	2	2	-	-	-	-	-	-	2	2	2
Apply classification algorithms to classify a given test record	**L3**	2	2	2	2	-	-	-	-	-	-	2	2	2
Explain clustering techniques to cluster the data objects.	**L2**	2	2	2	2	-	-	-	-	-	-	2	2	2
1- Low, 2- Moderate, 3-High														

Course Title: Multimedia Computing			
Course Code: P13CS662	**Semester : VI**	**L-T-P-H: 4-0-0-4**	**Credits:4**
Contact Period : Lecture :52 Hrs., Exam: 3Hrs.		**Weightage :CIE:50% SEE:50%**	

Prerequisites

- Computer graphics

Course Objective

This course aims to

1. Define multimedia and describe the components of multimedia systems, the technologies used for multimedia system, and storage management of multimedia system
2. Understand applications of multimedia systems and describe the use of sound, music and speech in multimedia.
3. Describe the computerized graphic and images and their respective properties.
4. Understand characteristic and various formats to represent video and animations understand different compression methods used for audio and video signals.
5. Describe the basic optical technology as well as techniques that represent the core of CD-DA, CD-ROM and DVD.
6. Explain different type of currently available content analysis techniques and describe a large number of file and data formats.

Course Content

Unit 1

Introduction, Media and Data Streams, Audio Technology: Introduction: Multimedia Element, Multimedia Applications, Multimedia Systems Architecture, Evolving Technologies for Multimedia Systems, Defining objects for Multimedia systems, Multimedia data Interface standards, the need for data compression, Multimedia Databases.
Media: Perception media, Represent media, Presentation media, Storage media, Characterizing Continuous media data streams.
Sound: Frequency, Amplitude, Sound Perception and Psychoacoustics, Audio representation on Computers, Three dimensional sound projection, Music and MIDI standards, Speech signals, Speech Output, Speech Input, Speech Transmission. 10 hours

Unit 2

Graphics and Images, Video Technology, Computer-Based Animation
Capturing Graphics and Images, Computer Assisted Graphics and Image Processing, Reconstructing Images, Graphics and Image output Options.
Basics, Television System, Digitalization of video Signals, Digital Television.
Basic Concepts, Specification of Animations, Methods of Controlling Animation, Virtual Reality Modeling Language. 11 hours

Unit 3

Data Compression: Storage Space, Coding Requirements, Source, Entropy and Hybrid Coding, Basic Compression Techniques, JPEG: Image Preparation, Lossy Sequential DCT-based Mode, Expanded Lossy DCT-base Mode, Lossless Mode, Hierarchical Mode. H.261(Px64) and H.263: Image Preparation, Coding Algorithm, Data Stream, H.263+ and H.263L, MPEG: Video Encoding, Audio Encoding, Data Stream, MPEG-2,MPEG-4, MPEG-7, Fractal Compression. 10 hours

Unit 4

Optical Storage Media: History of Optical Storage, Basic Technology, Video Discs and Other WORMs, Compact Disc Digital Audio, Compact Disc Read Only Memory, CD-ROM Extended Architecture, Further CD-ROM Based Developments, Compact Disc Recordable, Compact Disc Magneto-Optical, Compact Disc Read/Write, Digital Versatile Disc. 10 hours

Unit 5

Content Analysis, Data and File Format standards: Simple Vs. Complex Features, Analysis of Individual Images, Analysis of Image Sequences, Audio Analysis, Applications. Data and File Format standards: Rich Text Format, TIFF File Format, Resource Interchange File Format [RIFF], MIDI File Format. TWAIN. 11 hours

Text Books:

1. Multimedia Fundamentals: Vol 1- Media Coding and Content Processing – Ralf Steinmetz, Klara Narstedt, 2nd Edition, Pearson Education / PHI, 2003.
2. Multimedia Systems Design - Prabhat K. Andleigh, Kiran Thakrar, PHI, 2003.

Reference Books:

1. Multimedia Communication Systems: Techniques, Standards, and Networks - K.R Rao, Zoran S. Bojkovic and Dragorad A. Milovanovic, Pearson Education, 2002.
2. Multimedia Information Networking - Nalin K Sharad, PHI, 2002.

Course Outcome

After learning all the units of the course, the student is able to

1. Define multimedia and describe the basic requirements of multimedia system.
2. Understand few properties, formats and standards to represent graphics and images, video and animations.
3. Describe different compression techniques used for audio and video.
4. Explain various optical storage media technology.
5. Understand content analysis techniques and various data and file formats standards.

Topic learning objectives

Unit 1

1. Define multimedia.L1
2. Define the basics of the various data elements and application.L1
3. Explain the applications of multimedia.L2
4. Design of multimedia system architecture.L3
5. Define Objects of multimedia system.L1
6. Explain different key attributes of multimedia and characteristics of data streams.L2
7. Describe the storage management of multimedia system. L2
8. Describe how audio material is represented in computers.L1
9. Explain the formats used in audio technology.L2
10. Explain speech signal with their input, output and transmission.L2

Unit 2

1. Describe computerized graphics and images and their respective properties. L1
2. Explain how images can be acquired, manipulated and output on computers.L2
3. Explain how the graphics and images represented and processed on computer.L3
4. Describe some important formats of graphics and images.L2
5. Explain methods used to retransform two-dimensional images into three dimensional spaces.L2
6. Describe how to output graphics and images on output devices.L2
7. Describe video technology and video standards with respect to properties of human perception.L2
8. Explain the basic concepts of computer based animations.L2
9. Describe the specification of animations.L2
10. Explain the method of controlling animation.L2

Unit 3

1. Understand the storage space required and coding requirements of text, audio and video. L1

2. Distinguish the compression techniques into three types of coding.L3
3. Describe the major steps of data compression.L2
4. Explain few basic compression techniques used in multimedia systems.L2
5. Describe the JPEG compression technique for still images.L2
6. Describe the H.263 (H.261 px64) compression technique for low resolution video sequences. L2
7. Describe the MPEG compression technique for audio and video.L2
8. Explain the MPEG-2 compression technique for audio and video.L2
9. Explain the MPEG-4 compression standard.L2
10. Describe the fractal compression techniques for fractal images and videos.L2

Unit 4

1. Describe the history of Optical Storage.L1
2. Describe the basic technology used in optical storage media. L2
3. Explain briefly Video discs and other WORM's. L2
4. Explain the basic technology, Eight-to-Fourteen modulation in CD-DA. L2
5. Describe how the errors are handled in CD-DA. L2
6. Explain CD-ROM and CD-ROM/XA based on CD-DA. L2
7. Explain Compact Disc Recordable. L2
8. Explain Compact Disc Magneto-optical and Compact Disc read/write. L2
9. Explain Digital Versatile Disc (DVD). L2
10. Compare CD and DVD. L2

Unit 5

1. Distinguish b/n simple and complex indicator features for content analysis.L2
2. Understand techniques to analyse individual images and image sequences. L3
3. Explain how audio analysis is performed using syntactic and semantic indicators. L2
4. Understand applications of content analysis. L2
5. Explain Rich Text Format. L2
6. Describe Tagged Image File Format (TIFF). L2
7. Explain Resource Interchange File Format (RIFF). L2
8. Explain MIDI file format. L2
9. Describe TWAIN specification objective. L2
10. Describe Twain architecture. L3

Review questions

1. List any four benefits of multimedia system.
2. Describe the data elements for multimedia system.
3. Explain the multimedia workstation architecture with neat diagram.
4. Explain the issues related to multimedia storage and retrieval.
5. In detail, explain MIDI standards.
6. Explain the three characteristics for continuous media data stream that occur in multimedia systems.
7. Explain the following terms i) sampling rate ii) Quantization
8. With neat diagram, explain the principle of speech recognition.
9. Explain the need for data compression.
10. Explain how full motion video can be used in multimedia applications.
11. Describe the GIF and TIFF image file format.
12. Explain the concept of image segmentation.
13. Explain graphics and image output options.
14. Explain the coding methods used for digitization of video signal.
15. Define the following parameters w.r.t display monitor
    i) Resolution ii) Aspect ratio.
16. Describe various formats to represent video signals.
17. Write a note on VRML with example.

18. The VGA format can present 640x480 pixels with 256 simultaneous colors. The monitor is controlled via an analog RGB output. What is the storage capacity per frame required?
19. List and explain any four techniques for controlling animation.
20. Explain the two approaches for transmission of animations.
21. An uncompressed stereo audio signal of CD quality is sampled at 44.1 kHz and quantized using 16 bits. Calculate the data rate and space required to store **2**min audio clip.
22. The letters A,B,C,D and E are to be encoded and have frequency as follows A= 24, B=12, C=10, D=8, E=8
    Use Huffman coding to derive a code for each letter.
23. Write a note on
    i) Run length coding ii) Vector quantization iii) Predictive or relative coding
24. With a neat diagram, explain the steps of lossy sequential DCT-mode of JPEG image compression.
25. Explain lossless and hierarchical JPEG modes.
26. Explain the two methods of coding in H.261.
27. Explain how the four types of images are coded in MPEG.
28. Explain the MPEG-4 system layer model.
29. Write a note on MPEG-7 and Fractal compression.
30. Explain the different layers of video stream in MPEG.
31. Explain hoe CD-DA works.
32. With neat diagram, explain the working principle and the process of error handling in CD-DA.
33. Describe the component of frames and areas in CD-DA.
34. Explain the mode-1 and mode-2 specification of CD-ROM.
35. In detail, explain Compact Disc Interactive.
36. Explain compact disc magneto-optical.
37. Explain the eight-to-fourteen modulation and process of error handling in CD-DA.
38. In detail, explain CD-R
39. Explain the 6 layers of DVD- video decoder to transfer MPEG data.
40. Describe the structure of DVD sector.
41. With a neat diagram, explain how the text recognition takes place in OCR systems.
42. Explain block and pixel oriented motion vector in image sequence analysis.
43. Explain the analysis of individual images in content analysis.
44. Explain the analysis of digital audio in content analysis.
45. Explain cut detection based on edge extraction.
46. Explain the different methods of controlling animation.
47. Describe the structure of TIFF and their tags.
48. Explain the key format information in RTF document files.
49. What are the TWAIN specification objectives?
50. Describe TWAIN architecture with neat diagram.

Lesson plan

Unit 1

1. Multimedia Element, Multimedia Applications.
2. Multimedia Systems Architecture, Evolving Technologies for Multimedia Systems.
3. Defining objects for Multimedia systems, Multimedia data Interface standards.
4. The need for data compression, Multimedia Databases.
5. Perception media, Represent media, Presentation media, Storage media.
6. Characterizing Continuous media data streams.
7. Frequency, Amplitude, Sound Perception and Psychoacoustics.
8. Audio representation on Computers, Three dimensional sound projection.

9. Music and MIDI standards, Speech signals.
10. Speech Output, Speech Input, Speech Transmission.

Unit 2

1. Introduction, Capturing Graphics and Images.
2. Capturing Graphics and Images.
3. Computer Assisted Graphics and Image Processing.
4. Reconstructing Images, Graphics and Image output Options.
5. Video Technology: Basics.
6. Television System, Digitalization of video Signals, Digital Television.
7. **Computer-Based Animation:** Basic Concepts, Specification of Animations.
8. Methods of Controlling Animation, Virtual Reality Modeling Language

Unit 3

1. Storage Space, Coding Requirements.
2. Source, Entropy and Hybrid Coding, Basic Compression Techniques.
3. Basic Compression Techniques.
4. JPEG: Image Preparation.
5. Lossy Sequential DCT-based Mode.
6. Expanded Lossy DCT-base Mode, Lossless Mode, Hierarchical Mode.
7. H.261(Px64) and H.263: Image Preparation, Coding Algorithm, Data Stream, H.263+ and H.263L
8. MPEG: Video Encoding, Audio Encoding, Data Stream
9. MPEG-2, MPEG-4
10. MPEG-4, MPEG-7,Fractal Compression

Unit 4

1. History of Optical Storage, Basic Technology.
2. Video Discs and Other WORMs, Compact Disc Digital Audio.
3. Compact Disc Digital Audio.
4. Compact Disc Read Only Memory.
5. CD-ROM Extended Architecture.
6. Further CD-ROM Based Developments.
7. Compact Disc Recordable.
8. Compact Disc Magneto-Optical.
9. Compact Disc Read/Write.
10. Digital Versatile Disc.

Unit 5

1. Simple Vs. Complex Features, Analysis of Individual Images.
2. Analysis of Image Sequences.
3. Audio Analysis.
4. Applications.
5. Rich Text Format.
6. TIFF File Format.
7. Resource Interchange File Format [RIFF].
8. MIDI File Format.
9. TWAIN

Course Articulation Matrix (CAM)													
Course Outcome (CO)		**Program Outcome (ABET/NBA-(3a-m))**											
		a	**b**	**C**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Definition of multimedia and describe the basic requirements of multimedia system.	**L1,L2**		L				M	H	L	M		M		
Understand few properties, formats and standards to represent graphics and images, video and animations.	**L2**		M	L			M	M	L	M		M		
Describe different compression techniques used for audio and video.	**L2**		M	M		M	M	H	L	M		M	L	L
Explain various optical storage media technology.	**L2**				M		M	M		M				
Understand content analysis techniques and various data and file formats standards.	**L2,L3**		M		M	M	M	M		M		M	M	L
L- Low, M- Moderate, H-High														

Course Assessment Matrix (CAM)													
Course Outcome (CO)		**Program Outcome (ABET/NBA-(3a-m))**											
		a	**b**	**C**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Definition of multimedia and describe the basic requirements of multimedia system.	**L1, L2**		L				M	H	L	M		M		
Understand few properties, formats and standards to represent graphics and images, video and animations.	**L2**		M	L			M	M	L	M		M		
Describe different compression techniques used for audio and video.	**L2**		M	M		M	M	H	L	M		M	L	L
Explain various optical storage media technology.	**L2**				M		M	M		M				
Understand content analysis techniques and various data and file formats standards.	**L2, L3**		M		M	M	M	M		M		M	M	L
L- Low, M- Moderate, H-High														

Course Title : JAVA and J2EE			
Course Code: P13CS663	**Semester : VI**	**L-T-P-H: 4-0-0-4**	**Credits:4**
Contact Period : Lecture :52 Hrs., Exam: 3Hrs.		**Weightage :CIE:50% SEE:50%**	

Prerequisites

- Concepts of basic programming language
- Object oriented programming

Course Learning Objectives

The objective of this course is to

1. **Adopt** object oriented features to develop java applications, **Handle** the exceptions and **describe** key issues of modern animations.
2. **Apply** the multithreading programming to solve synchronization problems and **develop** generic methods and classes..
3. **Write** the java applications to deal with events using delegation event model , **develop** the applets and **Describe** the use of collection framework.
4. **Describe JDBC process , Use** statement object to manipulate database and **Create** a J2ee component using java servlet technology.
5. **Create** a JSP that can be used as a middle level program between clients and web Services, **Use** Java Remote Method Invocation to invoke Server side objects that are written in Java, **Create** and use JavaBeans.

.

Course content

Unit – 1

I**ntroduction**, Histroy of Java, Java Buzzwords, Java's Bytecode, Java Development Kit (JDK), Object oriented programming, Simple Java programs.

Introducing classes : ClassFundamentals, Declaring Object, Assigning object reference variables, Constructors, This key word, Garbage collection, overloading methods, Acess control, final key word, nested and inner classes.

Inheritance: Simple, multiple, and multilevel inheritance, Super classes, Order of calling constructors, Overriding, Abstract classes, Using final with inheritance.

Interfaces and packages, Exception handling in Java. Enumerations, Autoboxing and Annotations(Metadata): Enumerations, Type Wrappers, Autoboxing , Annotations (Metadata). 9 Hours

Unit – 2

Multi threaded programming: Java's thread model, the main thread, creation of threads, Multiple threads, isAlive() and join(), thread priorities, Synchronization, Interthread communication, suspending, reassuming, and stopping threads.

Networking : Networking Basics, the Networking classes and interfaces, InetAddress, Inet4Addresses and inet6address, TCP/IP Client sockets, URL, URL Coneection, HttpURL connection, the URL Class, Cookies, TCP/IP server Sockets, datagram.

Generics : Introduction, A Simple Generics Example, A Generic Class with Two Type Parameters, The Generic Form of a Generic Class, Bounded Types, Using Wildcard Arguments, Creating a Generic Method, Generic Interfaces, Generic Class Hierarchies, Ambiguity Errors, Some generic Restrictions. 12 Hours

Unit – 3

The Collections Framework : Collections Overview, Recent Changes to Collections, The Collections Interfaces, The Collection Classes, Accessing a Collection via an Iterators, storing user defined classes in collections, The Random Access Interface, Comparators, The

Collection Algorithms, rrays, The Legacy Classes and Interfaces: enumeration, interface, vector, stack.

Applets: Two types of Applets, Applet basics, Applet Architecture, An Applet skeleton, Simple Applet display methods, Requesting repainting, Using the Status Window, The HTML APPLET tag, Passing parameters to Applets, get Documentbase() and get Codebase().

Event Handling: Event handling mechanisms, The delegation event model, Event classes, Sources of events, Event listener interfaces, Using the delegation event model, Adapter Classes, Inner classes. 10 Hours

Unit –4

Java 2 enterprise edition overview, database access: Overview of J2EE and J2SE. The Concept of JDBC, JDBC Driver Types, JDBC Packages, A Brief Overview of the JDBC process, Database Connection, Associating the JDBC/ODBC Bridge with the Database, Statement Objects, ResultSet, Transaction Processing, Metadata, Data types, Exceptions.

Servlets : Background, The Life Cycle of a Servlet, Using Tomcat for Servlet Development, A simple Servlet, The Servlet API, The Javax.servlet Package, Reading Servlet Parameter, The Javax.servlet.http package, Handling HTTP Requests and Responses, Using Cookies, Session Tracking. 11 Hours

Unit – 5

JSP : Java Server Pages (JSP): JSP, JSP Tags. Tomcat, Request String, User Sessions, Cookies, Session Objects.

Java Remote Method Invocation: Remote Method Invocation concept, Server side, Client side.

Enterprise java beans : Enterprise Java Beans, Deployment Descriptors, Session Java Bean, Entity Java Bean, Message-Driven Bean, The JAR File. 10 Hours

Text Books:

2. **Java The Complete Reference** - Herbert Schildt, 7th Edition, Tata McGraw Hill, 2007.
3. **J2EE The Complete Reference** - Jim Keogh, McGraw Hill. 2007.

Reference Books:

1. **Introduction to JAVA Programming** - Y. Daniel Liang, 6th Edition, Pearson Education, 2007.
2. **The J2EE Tutorial**, Stephanie Bodoff et ak, 2nd Edition Pearson Education, 2004.

Course outcomes

1. **Create** Java applications that leverage the object-oriented features.
2. **Implement** multithreaded programs, socket programs and generic classes.
3. **Develop** application framework, applets and handle events using event handling mechanism.
4. **Use java** Data objects to connect to the DBMS and create J2ee componentt using servlets.
5. **Write applications** using JSP and **develop a** J2ee component using EJB technology and also call an EJB from a Java Servlet and JSP.

Topic Learning Objectives

Unit 1

Topic:

Introduction to Java, Introduction to classes , Class Fundamentals, Object Declaration, Assigning object reference variables, Constructors, This and final key words, Garbage collection, overloading methods, nested and inner classes. Inheritance and its types, Super classes, Order of calling constructors, Overriding, Abstract classes, Interfaces and packages, Exception handling in Java. Enumerations, Autoboxing and Annotations.

Learning Objective

1. **Define** the characteristics of java (L1).
2. **Define** the Java Development Kit (JDK) (L1).
3. **Describe** the structure of a java program(L1).
4. **Define** classes and objects, and use classes to model objects(L1).
5. **Apply** the concept of method abstraction in software development (L4).
6. **Define** inheritance and its types (L1).
7. **Define** interfaces and packages (L2).
8. **Handle** the exceptions raised in java programming (L3).
9. **Create** Enumerations (L6).
10. **Explain** key issues of modern animations, such as sound synchronization and moving objects in layers (L1).

Unit 2

Topic:

Java's thread model, the main thread, creation of threads, Multiple threads, thread priorities, Synchronization, Interthread communication, Networking Basics, the Networking classes and interfaces, InetAddress, , TCP/IP Client sockets, URL, Cookies, TCP/IP server Sockets, datagram. Introduction to Generics ,, The Generic Form of a Generic Class, Bounded Types, Using Wildcard Arguments, Generic Interfaces, Generic Class Hierarchies.

Learning Objective

1. **Define** java thread model(L1).
2. **Describe** the life cycle of threads(L2).
3. **Creating and Managing** Threads(L3).
4. **Apply** the multithreading programming to synchronization(L3).
5. **Implement** the program for Interthread communication(L6).
6. **Describe** how to set the priorities of thread (L2).
7. **Write** the java program to access network resouces(L6).
8. **Write the java program to make** Communicate between processes using network sockets(L6).
9. **Define** generics(L1).
10. **Write** a java program that shows creation of a generic method(L6).

Unit 3

Topic:

Introduction to Collections Framework , The Collections Interfaces, and Classes, Comparators, The Collection Algorithms, The Legacy Classes and Interfaces, Introduction to Applets: , Applet Architecture, An Applet skeleton, Simple Applet display methods, The HTML APPLET tag, Event handling mechanisms, The delegation event model, Event classes, Sources of events, Event listener interfaces,

Learning Objective

1. **Explain** how the web browser controls and execute applets(L2).
2. **Explain** the life cycle of the applet(L2).
3. **Combine** applet in web pages(L6).
4. **show** the executin of applets from applet viewer and from web btowser(L6).
5. **Write** a java program that can run as both an application and an applet(L6).
6. **Describe** the delegation event model(L2).
7. **Write** the java program to deal with ActionEvent , MouseEvent, KeyEvent, WindowEvent(L6).
8. **Describe** the use of collection framework(L2).
9. **Use and manipulate** several core data structures like Arrays, linked lists, trees, stacks, and queues using collection framework(L3).
10. **Develop** simple Java user interfaces and identify where data structures are appearing in those user interfaces(L6).

Unit 4

Topic:

Introduction to J2EE and J2SE. The Concept of JDBC, Overview of the JDBC process, Statement Objects, ResultSet, Transaction Processing, The Life Cycle of a Servlet, package, Handling HTTP Requests and Responses, Cookies, Session Tracking.

Learning Objective

1. **Explain** the working of JDBC (L2).
2. **Describe** the different types of JDBC Driver(L2).
3. **Use** statement object to execute SQL Statement (L6).
4. **Use** prepared statement object to execute precompiled SQL statement(L6).
5. **Use** callable statement object to execute stored SQL procedures and functions(L6).
6. **Define** Servlet and Servlet API(L1).
7. **Explain** the life cycle of the Servlet(L2).
8. **Create** and run the servlets(L6).
9. **Use** cookies(L6).
10. **Explain** how to track the sessions(L2).

Unit 5

Topic

JSP Tag, User Sessions, Cookies, Session Objects. Remote Method Invocation concept on Server side and Client side. Introduction to Enterprise Java Beans, Deployment Descriptors, Session Java Bean, Entity Java Bean, Message-Driven Bean.

Learning Objective

1. **Use** different types of JSP tags (L3).
2. **Create** a JSP page (L6).
3. **Use** cookies and sessions(L3).
4. **Describe** RMI invocation concept(L2).
5. **Use** RMI invocation to invoke server side objects that are written in java(L3).
6. **Describe** Enterprise Java Beans(L2).
7. **Explain** the difference between stateless and stateful session beans(L2).
8. **Explain** the difference between session beans and entity beans(L2).
9. **List** the applications of JAR File and Deployment Descriptors(L1).
10. **Create** the Enterprise Java Bean (L6).

Review Questions

Unit – 1

1. Differentiate between java and c++.
2. Describe the charecteristics of java.
3. With an example illustrate method overloading and overriding.
4. Distinguish between abstract and innerclass.
5. Define interface,Give an example ,where interface can be used to support multiple inheritance.
6. Describe the three uses of final keyword ?
7. Write a java program to create a try block that is likely to generate three types of exceptions and must incorporate catch blocks to catch and handle the exceptions and include a finally blocks.
8. Define a nested class? With an example illustrate how inner class members can be accessed by enclosing class.
9. Define type wrappers.explain different types of types wrappers.
10. Write a java program that illustrate the use of marker annotation.

Unit – 2

1. Define java thread model? Explain how threads can be create
2. What is the role of synchronization in threads? Demonstrate the program using synchronized methods.

3. Write a java program, that illustrate the methods defined by thread, to pause and restart the execution of a thread.
4. Define generics? With an example program explain how generics improve type safety.
5. Write a java program that shows creation of a generic method within a non generic class.
6. Explain isAlive() and join() methods.
7. Write a java program to create a thread access the priority and set the priority to the thread.
8. Write a java program that implements client-server communication by writing message typed at server window across the n/w to the client side where they are displayed.
9. List and explain the different constructors used to create a datagram Packet.
10. Write a java program to demonstrate the actions of a bounded wildcard argument.

Unit – 3

1. Define delegation event model? Describe the roles of sources events and listerns in event handling mechanism.
2. Write a java program to demonstrate virtual key words for handling keyboard events.
3. Explain the life cycle of Applets.
4. Write HTML APPLET tag.
5. List the events generated by key board? Write a java program to show the status of keyboard key(excluding special keys) in the status window and which echoes key strokes to the applet window.
6. Distinguish between applet and applications.write an applet to scroll a message "Java is simple" from left to right across the applet window and display the message "Object oriented language" in status window.
7. Define collection frame work? List and explain the goals of collection framework
8. Write a java program to create an array of strings ,then the following operant
    a. display the list.
    b. remove the string.
9. Write a java program to create a linked list using LINKEDLIST class and perform the following operations on the list.
    a) add elements at the beginning of the list.
    b) add elements at the end of the list.
    c) obtain\acess last element.
    d) remove first and last element.
10. Write a java program to find the largest element in a collection using iterator in for each.

Unit – 4

1. Describe the working of JDBC? List and explain the different types of JDBC drivers.
2. Define result set ?write a program to read the data from the result set ? and the detea row in the result set.
3. Write a java program to check whether a driver support a scrollable result set or not.
4. Define save point ?write a program to execute a database transaction using save points.
5. Define transaction processing ?write a program to execute a database transanction.
6. Define servlet ? explain the life cycle of the servlets.
7. Define session? Write a java program to read and modify the session attributes.
8. Write the java code for the following
    a. Write HTML code for read cookie value from user and pase it to the servlet
    b. Write a servlet to read cookie value from web page and add the cookie to the header of HTTP respons
    c. Write a servlet to read cookie from HTTP get request and write thosen to HTTP response
9. Write a java servlet which reads 2 parameters from the webpage which are of type integers and find the sum of those 2 values and return back the result as webpage.
10. Write a hmtl document that call a servlet and that servlet display a getting message, explain the methods used to write servlet.

Unit – 5

1. Define JSP?list and explain the various tags used in JSP.
2. Write a JSP to create a session attribute and read session attributes.
3. Define cookies?write a JSP to create and read cookie named userid that stores the value cs766.
4. Describe the concept of RMI.
5. Write the differences between server side and client side RMI.
6. Write an XHTML document that collect a number from client and call a JSP to display largest digit of a given number on screen.
7. Define deployment descriptor?list the deplaument descriptor for EJB1.1.
8. Define entity java bean?with the skeleton of java entity bean.
9. List and explain the transaction attributes defined in deployment descriptor.
10. Describe the EJB interface and environment with the help of a block diagram.

Lesson Plan

Unit - I

1. Introduction to Java, Java Development Kit (JDK), Simple Java programs.
2. Class Fundamentals, Declaring Object, Assigning object reference variables, Constructors.
3. This key word, Garbage collection, overloading methods, final key word, nested and inner classes
4. Inheritance: Simple, multiple, and multilevel inheritance, Super classes.
5. Order of calling constructors, Overriding, Abstract classes, Using final with inheritance.
6. Interfaces and packages.
7. Exception handling in Java.
8. Autoboxing .
9. Annotations(Metadata).

Unit - 2

1. Java's thread model, the main thread, creation of threads.
2. Multiple threads, isAlive() and join(), thread priorities.
3. Synchronization.
4. Inter thread communication, suspending, reassuming, and stopping threads.
5. Networking Basics, the Networking classes and interfaces, InetAddress.
6. Inet4Addresses and inet6address, TCP/IP Client sockets, URL, URL Connection.
7. HttpURL connection, the URL Class, Cookies, TCP/IP server Sockets, datagram.
8. Introduction to **Generics** , A Simple Generics Example.
9. A Generic Class with Two Type Parameters.
10. The Generic Form of a Generic Class.
11. Bounded Types, Using Wildcard Arguments.
12. Generic Method, Generic Interfaces, Generic Class Hierarchies.

Unit - 3

1. The Collections Framework, Recent Changes to Collections.
2. The Collections Interfaces, The Collection Classes.
3. Accessing a Collection via an Iterators, storing user defined classes in collections.
4. The Random Access Interface, Comparators.
5. The Collection Algorithms, rrays, The Legacy Classes and Interfaces: enumeration, interface, vector, stack.
6. Two types of Applets, Applet basics, Applet Architecture, An Applet skeleton, Simple Applet display methods.
7. Requesting repainting, Using the Status Window ,The HTML APPLET tag, Passing parameters to Applets.
8. getDocumentbase() and getCodebase(), Event Handling mechanisms.

9. The delegation event model, Event classes, Sources of events,Event listener interfaces.
10. Using the delegation event model, Adapter Classes, Inner classes.

Unit - 4

1. Overview of J2EE and J2SE. The Concept of JDBC, JDBC Driver Types.
2. JDBC Packages, A Brief Overview of the JDBC process, Database Connection.
3. Associating the JDBC/ODBC Bridge with the Database, Statement Objects.
4. ResultSet.
5. Transaction Processing.
6. Metadata, Data types, Exceptions.
7. Introduction to servlet The Life Cycle of a Servlet, Using Tomcat for Servlet Development.
8. The Servlet API, The Javax.servlet Package, Reading Servlet Parameter.
9. The Javax. servlet.http package.
10. Handling HTTP Requests and Responses.
11. Using Cookies, Session Tracking.

Unit - 5

1. Java Server Pages overview, JSP Tags.
2. Tomcat, User Sessions.
3. Cookies, Session Objects.
4. Remote Method Invocation concept.
5. Server side, Client side RMI.
6. Enterprise Java Beans.
7. Deployment Descriptors, Session Java Bean.
8. Entity Java Bean.
9. Message-Driven Bean.
10. The JAR File.

I. Course Articulation Matrix (CAM)														
Course Outcome (CO)	**Program Outcome (ABET/NBA-(3a-m))**													
		a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Create Java applications that leverage the object-oriented features.	**L1, L6**	**M**	**H**	**H**	**M**	**M**	**-**	**-**	**-**	**-**	**-**	**H**	**M**	**L**
Implement multithreaded programs , socket programs and generic classes.	L4	**M**	**-**	**M**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**H**	**H**	**-**
Develop application framework , applets and handle events using event handling mechanism.	**L3, L6**	**M**	**L**	**M**	**-**	**M**	**-**	**-**	**-**	**-**	**-**	**H**	**H**	**-**
Use java Data objects to connect to the DBMS and create J2ee componentt using servlets.	**L3**	**-**	**M**	**H**	**-**	**M**	**-**	**-**	**-**	**-**	**-**	**M**	**H**	**-**
Write applications using JSP and **develop** a J2ee component using EJB technology and also call an EJB from a Java Servlet and JSP.	L6	**M**	**-**	**-**	**H**	**-**	**-**	**-**	**-**	**-**	**-**	**H**	**H**	**-**
L- Low, M- Moderate, H-High														

J. Course Assessment Matrix (CaM)														
Course Outcome (CO)	**Program Outcome (ABET/NBA-(3a-m))**													
		a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Create Java applications that leverage the object-oriented features.	**L1,L6**	**2**	**3**	**3**	**2**	**2**	**-**	**-**	**-**	**-**	**-**	**3**	**2**	**1**
Implement multithreaded programs , socket programs and generic classes.	L4	**2**	**-**	**2**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**3**	**3**	**-**
Develop application framework , applets and handle events using event handling mechanism.	**L3,L6**	**2**	**1**	**2**	**-**	**2**	**-**	**-**	**-**	**-**	**-**	**3**	**3**	**-**
Use java Data objects to connect to the DBMS and create J2ee componentt using servlets.	**L3**	**-**	**2**	**3**	**-**	**2**	**-**	**-**	**-**	**-**	**-**	**2**	**3**	**-**
Write applications using JSP and **develop** a J2ee component using EJB technology and also call an EJB from a Java Servlet and JSP.	L6	**2**	**-**	**-**	**3**	**-**	**-**	**-**	**-**	**-**	**-**	**3**	**3**	**-**
L- Low, M- Moderate, H-High														

Course Title : Data Compression			
Course Code: P13CS664	**Semester : VI**	**L-T-P-H: 4-0-0-4**	**Credits: 4**
Contact Period : Lecture :52 Hrs., Exam: 3Hrs.		**Weightage :CIE:50% SEE:50%**	

Prerequisites

- Basic data structures and algorithms
- Fundamental concepts of computer architecture

Course Learning Objectives

1. Understand the essential ideas underlying data compression.
2. Become familiar with the different types of compression algorithms.
3. Be able to describe the most popular data compression algorithms in use today and know the applications for which each is suitable.
4. Understand the Applications of Huffman coding.
5. Able to describe the LZ77Approach,The LZ78 Approach

Course Content

Unit –1

Introduction ,Compression Techniques: Loss less compression , Lossy Compression , Measures of prefonnance , Modeling and coding , Mathematical Preliminaries for Lossless compression: Models- Physical models,probability models, Markov models,composite source model Coding: uniquely decodable codes , Prefix codes. 10 Hours

Unit – 2

Huffman coding :The Huffman coding algorithm: Design of a Huffman code ,Minimum variance Huffman codes, canonical Huffman codes. Adaptive Huffman coding: Update procedure , Encoding procedur,Decoding procedure. Golomb codes , Rice codes , Tunstall codes,Applications of Huffman coding: Loss less image compression , Text compression , Audio Compression. 10 Hours

Unit – 3

Arithmetic Coding ,Coding a sequence , Generating a binary code , Comparison of Binary and Huffman coding ,Applications,Dictionary Techniques: Introduction , Static Dictionary: Diagram Coding , Adaptive Dictionary. The LZ77Approach,The LZ78 Approach Applications: File Compression-UNIX compress , Image Compression: The Graphics Interchange Format (GIF) ,compression over Modems: V.42 bits , Predictive Coding: Prediction with Partial match (ppm): The basic algorithm , The ESCAPE SYMBOL , length of context , The Exclusion Principle ,The Burrows-Wheeler Transform: Moveto-front coding, CALIC , JPEG-LS , Multi-resolution Approaches , Facsimile Encoding , Dynamic Markoy Compression. 12 Hours

Unit –4

Mathematical Preliminaries for Lossy Coding ,Distortion criteria , Models , Scalar Ouantization: The Quantization problem , Uniform Quantizer , Adaptive Quantization , Non uniform Quantization. 10 Hours

Unit – 5

Vector Quantization ,Advantages of Vector Quantization over Scalar Quantization , The Linde-Buzo-Gray Algorithm , Tree structured Vector Quantizers , Structured Vector Quantizers.Audio coding: spectral masking,temporal masking,mpeg audio coding layer I coding,layer II coding,layer III coding. 10 Hours

Text Book:

1.Introduction to Data Compression, Khalid Sayood: Morgan Kaufmann Publishers, Inc., Fourth Edition.2013

References Books:

1. Anil K. Jain: Fundamentals of Digital Image Processing, Prentice Hall, 1989.
2. Data Compressio : The Complete Reference David Saloman, Springer

Course Outcomes

On completion of data compression, you should be able to:

1. Outline the brief history of Data compression
2. Explain how to distinguish lossless data compression from lossy data compression
3. Outline the main compression approaches.
4. Measure the effect and efficiency of a data compression algorithm.
5. Identify applications of data compression
6. Describe The Quantization problem
7. Explain Advantages of Vector Quantization over Scalar Quantization

Topic Learning Objectives

Unit 1

1. **Define** Compression technique L1
2. **List** types of compression Algorithms.L1
3. **Differentiate between** Loss less compression , Lossy Compression L4
4. **Describe the** Modeling and coding of data compression algorithm L2
5. **List** different approaches to modeling .L1
6. **Explain** is codingL2
7. **Understand** Physical models and probability models, L2
8. **Design** Markov models and composite source model L6
9. **Describe** uniquely decodable codes L2
10. **Identify** Prefix codes. L1

Unit 2

1. **Design** Huffman code
2. **Understand** Minimum variance Huffman codes
3. **Describe** canonical Huffman codes
4. **Compare** Encoding procedure and Decoding procedure.
5. **Design** Update procedure for the adaptive Huffman coding algorithm
6. **Understand** decoding procedure
7. **Design** Golomb codes for different set of values of m
8. **Define** Rice codes
9. **List** Applications of Huffman coding
10. **Define** Tunstall codes
11. **Explain** Text compression and Audio Compression.

Unit 3

1. **Explain** Coding a sequence
2. **Describe** Generating a binary code
3. **Compare** Binary and Huffman coding
4. **List** Dictionary Techniques
5. **Understand** Static Dictionary
6. **Compare** The LZ77 and LZ78 Approach
7. **Describe** Prediction with Partial match
8. **Explain** Exclusion Principle
9. **Describe** Facsimile Encoding
10. **Define** Dynamic Markoy Compression.

Unit 4

1. **Understand** Distortion criteria
2. **List** properties of human visual and auditory systems
3. **Understand** The Quantization problem
4. **Describe** Uniform Quantizer
5. **Design** midrise quantizer
6. **Understand** Adaptive Quantization
7. **Compare** forward and backward adaptive quantization
8. **Describe** Non uniform Quantization
9. **Explain** pdf-optimized quantization
10. **Define** Companded quantization

Unit 5

1. **List** Advantages of Vector Quantization over Scalar Quantization
2. **Define** Scalar Quantization
3. **Design** The Linde-Buzo-Gray Algorithm
4. **Understand** Tree structured Vector Quantizers
5. **Demonstrate** Structured Vector Quantizers
6. **Define** Audio coding
7. **Compare** spectral masking,temporal masking
8. **List** mpeg audio coding method
9. **Understand** layer I coding,
10. **Explain** layer II coding,layer III coding

Review Questions

Unit 1

1. Define Compression Techniques
2. What is Loss less compression
3. What is Lossy Compression
4. Define Modeling Models- probability models, , Coding:
5. Define coding
6. Explain Mathematical Preliminaries for Lossless compression
7. What is Physical models,
8. Explain Markov models
9. Define composite source model
10. Explain uniquely decodable codes
11. List Prefix codes.

Unit 2

1. Write Huffman coding algorithm
2. **What is** a Huffman code
3. **Explain** Minimum variance Huffman codes
4. **Define** canonical Huffman codes.
5. **Write** Adaptive Huffman coding
6. **What is** Update procedure
7. **Dedifferentiate** Encoding procedur and Decoding procedure
8. **Explain** Rice codes and Tunstall codes,
9. **List** Applications of Huffman coding
10. **What is** Loss less image compression
11. **Define** Audio Compression

Unit 3

1. Explain Coding a sequence for Generating a binary code.
2. **Differentiate** Binary and Huffman coding
3. **List** Dictionary Techniques

4. **Explain** Diagram Coding of static Dictionary
5. Encode the following sequence using LZ77 algorithm: *barrayar~~b~~bar~~b~~by~~b~~barrayar~~b~~bay* assume you have a window size of 30 with a lookahead buffer of size 15. Furthermore , assume that C(a)=1,C(b)=2,C(~~b~~)=3,C(r)=4 and C(y)=5
6. **What is** UNIX compress
7. **Explain** V.42 bits and Predictive Coding
8. **Define** Prediction with Partial match (ppm)
9. **Write** The Burrows-Wheeler Transform
10. Define Facsimile Encoding
11. **Explain** Dynamic Markoy Compression

Unit 4

1. Explain Distortion criteria.
2. **Write** Scalar Ouantization
3. **Explain** The Quantization problem
4. **What is** Uniform Quantizer
5. Explain Adaptive Quantization
6. What is Non uniform Quantization

Unit 5

1. Explain Advantages of Vector Quantization over Scalar Quantization
2. Write The Linde-Buzo-Gray Algorithm
3. What is Tree structured Vector Quantizers
4. Explain Structured Vector Quantizers
5. What is Audio coding
6. Explain spectral masking
7. What is temporal masking
8. List layers of mpeg audio coding
9. Define layer I coding, ,
10. Explain layer II coding
11. What is layer III coding
12. Compare layer I coding and layer II coding
13. Differentiate layer II coding and layer III coding

Lesson Plan

Unit 1

1. Compression Techniques
2. Loss less compression
3. Lossy Compression
4. Measures of prefonnance
5. Mathematical Preliminaries for Lossless compression:
6. Modeling and coding
7. Models- Physical models,probability models
8. Markov models,composite source model
9. uniquely decodable codes
10. Prefix codes

Unit 2

1. The Huffman coding algorithm:,
2. Design of a Huffman code
3. Minimum variance Huffman codes
4. canonical Huffman codes.
5. Adaptive Huffman coding
6. Update procedure , Encoding procedur
7. Decoding procedure.

8. Golomb codes , Rice codes , Tunstall codes
9. Applications of Huffman coding: Loss less image compression
10. Text compression , Audio Compression.

Unit 3

1. Coding a sequence.
2. Generating a binary code , Comparison of Binary and Huffman coding
3. Applications and Dictionary Techniques: Introduction , Static Dictionary
4. Diagram Coding , Adaptive Dictionary
5. The LZ77Approach,The LZ78 Approach
6. Applications: File Compression-UNIX compress , Image Compression
7. The Graphics Interchange Format (GIF) ,compression over Modems: V.42 bits
8. Predictive Coding: Prediction with Partial match (ppm): The basic algorithm
9. The ESCAPE SYMBOL , length of context , The Exclusion Principle
10. Multi-resolution Approaches , Facsimile Encoding , Dynamic Markoy Compression.

Unit 4

1. Distortion criteria
2. Models
3. Scalar Ouantization
4. The Quantization problem
5. Uniform Quantizer
6. Non uniform Quantization
7. Non uniform Quantization
8. Adaptive Quantization
9. Adaptive Quantization
10. Compare Uniform Quantizer and Adaptive Quantization

Unit 5

1. Advantages of Vector Quantization over Scalar Quantization
2. The Linde-Buzo-Gray Algorithm ,
3. Tree structured Vector Quantizers
4. Tree structured Vector Quantizers
5. Audio coding
6. spectral masking
7. temporal masking
8. mpeg audio coding layer I coding
9. layer II coding
10. layer III coding

I. Course Articulation Matrix (CAM)

Course Outcome (CO)	Program Outcome (ABET/NBA-(3a-m))													
		a	**b**	**C**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Define Compression technique and Describe lossless and lossy compression technique (Unit-I)	L1	L	M	L		H	L		L	H		M	L	M
Understand properties and requirement modeling and coding of compression techniques (Unit-II)	L2	M			L			L	L		L	L		
Design Huffman and adaptive Huffman coding methods (Unit-II)	L6		H		H		L			L	M		M	H
Describe different approach LZ77 and LZ78 (Unit-III)	L2	M		M	M	L		H	H	M		H	H	
understand applications of arithmetic coding(unit-III)	L2		L	L				M	M		L	L		
Explain various mathematical preliminaries for loosy coding (Unit-IV)	L2	H	H		H	L	M		M	L	L	M	L	M
Compare advantages of vector quantization and scalar quantization in data compression (Unit-V)	L5	L		L			L	L		L				

L- Low, M- Moderate, H-High

Fig 3. Illustration of CAM of Data Compression

j. Course Assessment Matrix (CaM)

Course Outcome (CO)	Program Outcome (ABET/NBA-(3a-m))													
		a	**b**	**C**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Define Compression technique and Describe lossless and lossy compression technique (Unit-I)	L1	1	2	1		3	1		1	3		2	1	2
Understand properties and requirement modeling and coding of compression techniques (Unit-II)	L2	2			1			1	1		1	1		
Design Huffman and adaptive Huffman coding methods (Unit-II)	L6		3		3		1			1	2		2	3
Describe different approach LZ77 and LZ78 (Unit-III)	L2	2		2	2	1		3	3	2		2	3	
understand applications of arithmetic coding(unit-III)	L2		1	1				2	2		1	1		
Explain various mathematical preliminaries for loosy coding (Unit-IV)	L2	3	3		3	1	2		2	1	1	2	1	2
Compare advantages of vector quantization and scalar quantization in data compression (Unit-V)	L5	1		1			1	1		1				

1 – Low, 2 – Moderate and 3 – High

Fig 5a. Illustration of CAM of Data Compression

Course Title : C# & .NET			
Course Code : P13CS665	**Semester : VI**	**L - T – P-H : 4-0-0-4**	**Credits:4**
Contact Period: Lecture: 52 Hrs., Exam: 3 Hrs.		**Weightage: CIE:50; SEE:50**	

Prerequisites

- Concepts of basic programming language
- Object oriented programming

Course Objectives

This course aims to

1. Identify the basic building blocks of .NET
2. Provide the concepts of fundamentals of Object oriented programming.
3. Implement the C# programming and .NET features.
4. Build C# Applications with and without using IDE.
5. Design the C# applications to solve real world problems.
6. Analyze the real world problems to solve using object-oriented approach.
7. Learn C# Language Fundamentals, Exception handling and life time of the objects
8. Understand Interfaces and Collections in C#
9. Understanding .NET Assembles and Building a Simple File Test Assembly
10. Implement the advanced concepts of C# programming.

Course Content

Unit – 1

THE PHILOSOPHY OF .NET: Understanding the Previous State of Affairs, The .NET Solution, The Building Block of the .NET Platform (CLR, CTS, and CLS), The Role of the .NET Base Class Libraries, What C# Brings to the Table, An Overview of .NET Binaries (aka Assemblies), The Role of the Common Intermediate Language , The Role of .NET Type Metadata, The Role of the Assembly Manifest, Compiling CIL to Platform – Specific Instructions, Understanding the Common Type System, Intrinsic CTS Data Types, Understanding the Common Languages Specification, Understanding the Common Language Runtime, A tour of the .NET Namespaces, Increasing Your Namespace Nomenclature, Deploying the .NET Runtime.

BUILDING C# APPLICATIONS: The Role of the Command Line Complier (csc.exe), Building C # Application using csc.exe, Working with csc.exe Response Files,
Generating Bug Reports , Remaining C# Compiler Options, The Command Line Debugger (cordbg.exe), Using the Visual Studio .NET IDE, Other Key Aspects of the VS.NET IDE, C# "Preprocessor:" Directives, An Interesting Aside: The System.Environment Class.

10 Hours

Unit – 2

C# LANGUAGE FUNDAMENTALS: The Anatomy of a Basic C# Class, Creating objects: Constructor Basics, The Composition of a C# Application, Default Assignment and Variable Scope, The C# Member Initialization Syntax, Basic Input and Output with the Console Class, Understanding Value Types and Reference Types, The Master Node: System.Object, The System Data Types (and C# Aliases), Converting Between Value Types and Reference Types: Boxing and Unboxing, Defining Program Constants, C# Iteration Constructs, C# Controls Flow Constructs, The Complete Set of C# Operators, Defining Custom Class Methods, Understating Static Methods, Methods Parameter Modifies, Array Manipulation in C #, String Manipulation in C#, C# Enumerations, Defining Structures in C#, Defining Custom Namespaces.

OBJECT- ORIENTED PROGRAMMING WITH C#-1: Forms Defining of the C# Class, Definition the "Default Public Interface" of a Type, Recapping the Pillars of OOP, The First Pillars: C#'s Encapsulation Services, Pseudo- Encapsulation: Creating Read-Only Fields. 11 Hrs

Unit – 3

OBJECT- ORIENTED PROGRAMMING WITH C#-2: The Second Pillar: C#'s Inheritance Supports, keeping Family Secrets: The "Protected" Keyword, Nested Type Definitions, The Third Pillar: C #'s Polymorphic Support, Casting Between.

EXCEPTIONS AND OBJECT LIFETIME: Ode to Errors, Bugs, and Exceptions, The Role of .NET Exception Handing, The System. Exception Base Class, Throwing a Generic Exception, Catching Exception, CLR System – Level Exception (System. System Exception), Custom Application- Level Exception (System. System Exception), Handling Multiple Exception, The Family Block, the Last Chance Exception Dynamically Identifying Application – and System Level Exception Debugging System Exception Using VS. NET, Understanding Object Lifetime, the CIT of "new', The Basics of Garbage Collection, Finalization a Type, The Finalization Process, Building an Ad Hoc Destruction Method, Garbage Collection Optimizations, The System. GC Type.

INTERFACES: Defining Interfaces Using C# Invoking Interface Members at the object Level, Exercising the Shapes Hierarchy, Understanding Explicit Interface Implementation, Interfaces As Polymorphic Agents, Building Interface Hierarchies, Implementing, Implementation, Interfaces Using VS .NET, understanding the IConvertible Interface, Building a Custom Enumerator (IEnumerable and Enumerator), Building Cloneable objects (ICloneable), Building Comparable Objects (I Comparable). 11 Hours

Unit – 4

COLLECTIONS: Exploring the System.Collections Namespace, Building a Custom Container (Retrofitting the Cars Type).

CALLBACK INTERFACES, DELEGATES, EVENTS, AND

ADVANCED TECHNIQUES-1: Callback Interfaces, Understanding the .NET Delegate Type, Members of System. Multicast Delegate, The Simplest Possible Delegate Example, Building More a Elaborate Delegate Example, Understanding Asynchronous Delegates, Understanding (and Using) Events., The Advances Keywords of C#, A Catalog of C# Keywords Building a Custom Indexer, A Variation of the Cars Indexer Internal Representation of Type Indexer. Using C# Indexer from VB .NET, Overloading operators, The Internal Representation of Overloading Operators, Interacting with Overload Operator from Overloaded- Operator- Challenged Languages. 10 Hours

Unit – 5

ADVANCED TECHNIQUES-2: Creating Custom Conversion Routines, Defining Implicit Conversion Routines, The Internal Representations of Customs Conversion Routines.

UNDERSTANDING .NET ASSEMBLES: Problems with Classic COM Binaries, An Overview of .NET Assembly, Building a Simple File Test Assembly, A C#. Client Application, A Visual Basic .NET Client Application, Cross Language Inheritance, Exploring the CarLibrary's, Manifest, Exploring the CarLibrary's Types, Building the Multifile Assembly, Using Assembly, Understanding Private Assemblies, Probing for Private Assemblies (The Basics), Private A Assemblies XML Configurations Files, Probing for Private Assemblies (The Details), Understanding Shared Assembly, Understanding Shared Names, Building a Shared Assembly, Understanding Delay Signing, Installing/Removing Shared Assembly, Using a Shared Assembly. 10 Hours

Text Books:

1. Andrew Troelsen: Pro C# with .NET 3.0, Special Edition, Dream tech Press, India 2007. Chapters: 1 to 11.
2. E Balaguruswamy: Programming in C#, 5th Reprint, Tata McGraw Hill.

References:

1. Tom Archer: Inside C#, WP Publishers, 2007.
2. Herbert Schildt: C# The Complete Reference, Tata McGraw Hill.

Course Outcomes

After learning all the units of the course, the student is able to

1. Understand conceptual framework of .Net environment and Create Configuration for a given machine to host the .NET runtime.
2. Design the C# program with well understanding of C# language constructs.
3. Explain and analyze the advanced concepts in .Net such as Structured Exception Handling, interfaces.
4. Construct efficient C# programs for Event handling and callback interfaces.
5. Design .NET assemblies to solve real world problems.

Topic Learning Objectives

After learning all the topics of unit – I, the student is able to

1. Differentiate between C# and Java – L3.
2. Is it possible to have two Main ( ) in a C# code? If so, how is it resolved without using an IDE? Explain with an example – L2.
3. Define assembly. Explain each component of an assembly – L1, L2.
4. Differentiate between single file assembly and multi file assembly – L3.
5. Explain the role of CIL – L2.
6. Explain the following: i) #region, #endregion ii) conditional code compilation –L2.
7. Explain the role of noconfig and debug flag – L2.
8. Explain the role of csc.rsp with an example – L2.
9. Explain the role of cordbg.exe by using its flags with an example –L2.
10. Write a C# program to display the following information using the System environment class: i) Current directory of application ii) Operating System Version – L3.

After learning all the topics of unit – II, the student is able to

1. List and explain any three instance methods and static methods of System.Object – L1, L2.
2. With an example, explain what happens when reference type is passed by value and passed by reference – L2.
3. Explain boxing and unboxing, with an example – L2.
4. Explain four method parameter modifiers, with an example – L2.
5. Write a C# program to arrange 5 names in ascending order. The names are obtained from the command line arguments – L3.
6. Write a program in C# to read a jagged array and display the sum of all the elements of three inner arrays – L3.
7. Differentiate between System.String and System.Text.StringBuilder – L3.
8. Write a C# program to demonstrate use of static and Read- only variables – L3.
9. Illustrate the use of static constructor with a program – L3.
10. Define Properties. Differentiate between Readonly and Writeonly Properties – L1, L3.

After learning all the topics of unit – III, the student is able to

1. Define Inheritance. Give Example. Discuss how the base class can be controlled using base – L1, L2.
2. Define Polymorphism. Explain how the Polymorphism can be enforced using abstract classes and abstract methods with one programming example – L1, L2, and L3.
3. Define Errors, Bugs and Exceptions. Discuss what is the role of .NET Exception Handling? – L1, L2.
4. List and explain with code, the core members of System.Exception type – L1, L2.

5. Define a method that would sort an array of integers. Incorporate exception handling mechanism for "index out of bounds" situation. Develop a main program that employs this method to sort a given set of integers – L3.
6. Describe the role of .NET garbage collection, finalization process and AdHoc destruction method with examples – L2.
7. Explain how garbage collection is optimized in .NET – L2.
8. Explain the need of interfaces – L2.
9. Differentiate between interfaces and abstract class – L3.
10. Explain with an example, explicit interface implemenatation – L2.

After learning all the topics of unit – IV, the student is able to

1. Explain the interfaces of System.Collection – L2.
2. Explain the need of Callback Interfaces – L2.
3. Differentiate between synchronous and asynchronous delegate, with examples – L3.
4. Write a program to illustrate the implementation of multicast delegates – L3.
5. Write a program to illustrate the implementation of event handler – L3.
6. List out the applications of delegates – L1.
7. Explain the need of event handling – L2.
8. Explain the need of operator overloading – L2.
9. List out the operators that can be overloaded – L1.
10. Write a program to illustrate the overloading of +, - and * operators – L3.

After learning all the topics of unit – V, the student is able to

1. Write a program to illustrate the use of Custom Conversion Routines – L3
2. Explain the need of an assembly – L2.
3. List out the problems of Classic COM Binaries – L1.
4. Explain the steps involved in building a single file assembly, with an example – L2.
5. With a program illustrate the C# support for cross language inheritance – L3.
6. Explain the steps involved in building a multi file assembly, with an example – L2.
7. Differentiate between single file assembly and multi file assenbly – L3.
8. Explain the the specific steps involved in binding to a private assembly at runtime – L2
9. Explain the need of a shared sssembly – L2.
10. With a program illustrate the creation of a shared assembly – L3.

Review Questions

1. Differentiate between C++ and C#.
2. Discuss the Core Features of .NET
3. Discuss the role of CLR, CTS, CLS and Base Class Libraries with the help of block diagram.
4. Discuss the Features of C# Language.
5. Define Metadata and Manifest.
6. With a neat diagram explain the working of Microsoft Common Object runtime execution engine (mscoree.dll).
7. Explain how to Create and Compile a C# program in Command Mode and .NET IDE Mode.
8. Explain how to refer external single and Multiple assemblies in command mode and .NET IDE mode.
9. Discuss how to compile the Multiple source files in Command Mode and .NET IDE Mode.
10. Explain the following aspects of VS.NET IDE :
    i) Solution Explorer ii) Running and debugging iii) Documenting code via XML.
11. Give the Anatomy of a Basic C# Class. Discuss the different types of Main method.
12. Discuss the different types of Basic Input and Output with Console Class.
13. Differentiate between the Value types and reference types.

14. Discuss the characteristics of Static Method and Static data with examples.
15. Define Master Node. Explain the members of System.Object with examples.
16. Explain the following : i) for each loop ii) for loop
17. Differentiate between structure and enum.
18. Write a program in C# to sort an array of student objects having rollno, name and marks in two subjects.
    -Display the array sorted on names.
    -Display the array based on average marks.
19. Write a program in C# to accept two strings and perform the following operations:
    i) Copy string 2 to string 3.
    ii) Check string 1 ends with "ENGG" or not. If it is true, search character 'a' in string 3.
20. Discuss with illustration how the Encapsulation can be enforced using Traditional Accessors/Mutators and Class Properties.
21. Discuss the use of sealed classes with example.
22. Illustrate polymorphism with an example.
23. List and explain the core members of System.Exception type. How would you build custom exception?
24. Explain the keywords: i) finally ii) using.
25. Define object lifetime. Explain the Garbage Collection optimization process in C#.
26. Explain the process of finalizing objects in .NET environment.
27. Given the members of System.GC and explain their usage, with examples.
28. Write C# application to illustrate handling multiple exceptions.
29. What is the alternate approach to support multiple inheritance? List its major features.
30. Write a C# program which contains the following:
    • An interface called dimension with the methods length( ) and width( ), which returns length and width in centimeteres.
    • Another interface called metric dimension with the methods lengthinches( ) and widthinches( ),which returns length and width ininches.
    • A class box that implements both the above said interfaces. This class has two data members lengthinches and widthinches.
    Define appropriate constructor for the class box. Write a main program to create an instance of box and to display the box length and width in inches and centimeters by invoking the appropriate methods of two interfaces.
31. With an example, explain any four interfaces of System.Collection.
32. With a program illustrate the use of callback interfaces.
33. Define delegate and explain its use with an example.
34. List out the steps involved in creating and using a delegate.
35. List out the special features of a delegate.
36. Differentiate between synchronous and asynchronous delegate, with examples.
37. Describe the syntax of an event declaration.
38. List out two typical examples where events are used.
39. Explain the need of operator overloading.
40. List out the operators that are not overloadable.
41. Explain the internal representations of customs conversion routines.
42. Explain the need of an assembly.
43. With a neat diagram, explain the physical view and logical view of. NET assemblies.
44. Explain the C# support for cross language inheritance, with examples.
45. Explain the steps involved in building a single file assembly, with an example.
46. Explain the steps involved in building a multi file assembly, with an example.
47. List out the differences between single file assembly and multi file assenbly.
48. Explain the specific steps involved in binding to a private assembly at runtime

49. With a program explain how we can able to build a shared assembly.
50. With a program explain how we can install/remove a shared assembly.

Lesson Plan

Unit 1

1. **THE PHILOSOPHY OF .NET:** Understanding the Previous State of Affairs, The .NET Solution
2. The Building Block of the .NET Platform (CLR, CTS, and CLS), The Role of the .NET Base Class Libraries, What C# Brings to the Table,
3. An Overview of .NET Binaries (aka Assemblies), The Role of the Common Intermediate Language , The Role of .NET Type Metadata
4. The Role of the Assembly Manifest, Compiling CIL to Platform – Specific Instructions, Understanding the Common Type System
5. Intrinsic CTS Data Types, Understanding the Common Languages Specification, Understanding the Common Language Runtime
6. A tour of the .NET Namespaces, Increasing Your Namespace Nomenclature, Deploying the .NET Runtime.
7. **BUILDING C# APPLICATIONS:** The Role of the Command Line Complier (csc.exe), Building C # Application using csc.exe
8. Working with csc.exe Response Files, Generating Bug Reports , Remaining C# Compiler Options,
9. The Command Line Debugger (cordbg.exe) Using the, Visual Studio .NET IDE, Other Key Aspects of the VS.NET IDE
10. C# "Preprocessor:" Directives, An Interesting Aside: The System.Environment Class.

Unit 2

1. **C# LANGUAGE FUNDAMENTALS:** The Anatomy of a Basic C# Class, Creating objects: Constructor Basics
2. The Composition of a C# Application, Default Assignment and Variable Scope, The C# Member Initialization Syntax
3. Basic Input and Output with the Console Class, Understanding Value Types and Reference Types
4. The Master Node: System, Object, The System Data Types (and C# Aliases)
5. Converting Between Value Types and Reference Types: Boxing and Unboxing, Defining Program Constants, C# Iteration Constructs, C# Controls Flow Constructs
6. The Complete Set of C# Operators, Defining Custom Class Methods, Understating Static Methods, Methods Parameter Modifies
7. Array Manipulation in C #, String Manipulation in C#
8. C# Enumerations, Defining Structures in C#, Defining Custom Namespaces.
9. **OBJECT- ORIENTED PROGRAMMING WITH C#-1:** Forms Defining of the C# Class
10. Definition the "Default Public Interface" of a Type, Recapping the Pillars of OOP
11. The First Pillars: C#'s Encapsulation Services, Pseudo- Encapsulation: Creating Read-Only Fields.

Unit 3

1. **OBJECT- ORIENTED PROGRAMMING WITH C#-2:** The Second Pillar: C#'s Inheritance Supports, keeping Family Secrets: The "Protected" Keyword, Nested Type Definitions
2. The Third Pillar: C #'s Polymorphic Support, Casting Between.
3. **EXCEPTIONS AND OBJECT LIFETIME:** Ode to Errors, Bugs, and Exceptions, The Role of .NET Exception Handing
4. The System. Exception Base Class, Throwing a Generic Exception, Catching Exception, CLR System – Level Exception (System. System Exception)

5. Custom Application- Level Exception (System. System Exception), Handling Multiple Exception, The Finally Block
6. The Last Chance Exception Dynamically Identifying Application – and System Level Exception Debugging System Exception Using VS. NET, Understanding Object Lifetime, the CIT of "new'
7. The Basics of Garbage Collection, Finalization a Type, The Finalization Process, Building an Ad Hoc Destruction Method, Garbage Collection Optimizations, The System. GC Type.
8. **INTERFACES:** Defining Interfaces Using C# Invoking Interface Members at the object Level
9. Exercising the Shapes Hierarchy, Understanding Explicit Interface Implementation, Interfaces As Polymorphic Agents
10. Building Interface Hierarchies, Implementing, Implementation, Interfaces Using VS .NET, understanding the IConvertible Interface
11. Building a Custom Enumerator (IEnumerable and Enumerator), Building Cloneable objects (ICloneable), Building Comparable Objects (I Comparable)

Unit 4

1. **COLLECTIONS:** Exploring the system. Collections Namespace Building a Custom Container (Retrofitting the Cars Type).
2. **CALLBACK INTERFACES, DELEGATES, AND EVENTS,**
3. **ADVANCED TECHNIQUES-1:** Callback Interfaces, Understanding the .NET Delegate Type
4. Members of System. Multicast Delegate, The Simplest Possible Delegate Example
5. Building More a Elaborate Delegate Example, Understanding Asynchronous Delegates
6. Understanding (and Using) Events., The Advances Keywords of C#
7. A Catalog of C# Keywords Building a Custom Indexer
8. A Variation of the Cars Indexer Internal Representation of Type Indexer. Using C# Indexer from VB .NET
9. Overloading operators, The Internal Representation of Overloading Operators
10. Interacting with Overload Operator from Overloaded- Operator- Challenged Languages,

Unit 5

1. **ADVANCED TECHNIQUES-2:** Creating Custom Conversion Routines, Defining Implicit Conversion Routines
2. The Internal Representations of Customs Conversion Routines.
3. **UNDERSTANDING .NET ASSEMBLES:** Problems with Classic COM Binaries
4. An Overview of .NET Assembly, Building a Simple File Test Assembly
5. A C#. Client Application, A Visual Basic .NET Client Application
6. Cross Language Inheritance, Exploring the CarLibrary's, Manifest, Exploring the CarLibrary's Types
7. Building the Multifile Assembly, Using Assembly, Understanding Private Assemblies, Probing for Private Assemblies (The Basics)
8. Private A Assemblies XML Configurations Files, Probing for Private Assemblies (The Details)
9. Understanding Shared Assembly, Understanding Shared Names, Building a Shared Assembly
10. Understanding Delay Signing, Installing/Removing Shared Assembly, Using a Shared Assembly.

A. Course Articulation Matrix (CAM)														
Course Outcome (CO)		**Program Outcome (ABET/NBA-(3a-m))**												
		a	**b**	**C**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Understand conceptual framework of .Net environment and Create Configuration for a given machine to host the .NET runtime.	**L2 L6**		M	M	L	L			M			M	M	
Design the C# program with well understanding of C# language constructs	**L2 L6**		M	M	L	L			M			M	M	
Explain and analyze the advanced concepts in .Net such as Structured Exception Handling, interfaces.	**L2 L4**		M			L			H			M	M	
Construct efficient C# programs for Event handling and callback interfaces	**L6**		M	M	L	L			M			M	M	
Design .NET assemblies to solve real world problems.	**L6**		M	M	L	L			M			M	M	
L- Low, M- Moderate, H-High														

A. Course Articulation Matrix (CAM)														
Course Outcome (CO)		**Program Outcome (ABET/NBA-(3a-m))**												
		a	**b**	**C**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
Understand conceptual framework of .Net environment and Create Configuration for a given machine to host the .NET runtime.	**L2 L6**		2	2	1	1			2			2	2	
Design the C# program with well understanding of C# language constructs	**L2 L6**		2	2	1	1			2			2	2	
Explain and analyze the advanced concepts in .Net such as Structured Exception Handling, interfaces.	**L2 L4**		2			1			3			2	2	
Construct efficient C# programs for Event handling and callback interfaces	**L6**		2	2	1	1			2			2	2	
Design .NET assemblies to solve real world problems.	**L6**		2	2	1	1			2			2	2	
L- Low, M- Moderate, H-High														