HINDUSTAN UNIVERSITY

HINDUSTAN INSTITUTE OF TECHNOLOGY & SCIENCE

Department of Computer Science & Engineering

M.Tech. Computer Science & Engineering with Specialization in Pervasive Computing

Curriculum & Syllabus

2014 Regulations

ACADEMIC REGULATIONS
(M.TECH./ M.B.A. / M.C.A.)
(Full - Time / Part – Time)
(Effective 2014-15)

1. Vision, Mission and Objectives

1.1 The Vision of the Institute is "To make every man a success and no man a failure".

In order to progress towards the vision, the Institute has identified itself with a mission to provide every individual with a conducive environment suitable to achieve his / her career goals, with a strong emphasis on personality development, and to offer quality education in all spheres of engineering, technology, applied sciences and management, without compromising on the quality and code of ethics.

1.2 Further, the institute always strives

- To train our students with the latest and the best in the rapidly changing fields of Engineering, Technology, Management, Science & Humanities.
- To develop the students with a global outlook possessing, state of the art skills, capable of taking up challenging responsibilities in the respective fields.
- To mould our students as citizens with moral, ethical and social values so as to fulfill their obligations to the nation and the society.
- To promote research in the field of science, Humanities, Engineering, Technology and allied branches.

1.3 Our aims and objectives are focused on

- Providing world class education in engineering, technology, applied science and management.
- Keeping pace with the ever changing technological scenario to help our students to gain proper direction to emerge as competent professionals fully aware of their commitment to the society and nation.
- To inculcate a flair for research, development and entrepreneurship.

2. Admission

2.1. The admission policy and procedure shall be decided from time to time by the Board of Management (BOM) of the Institute, following guidelines issued by Ministry of Human Resource Development (MHRD), Government of India. The number of seats in each branch of the (M.TECH / M.B.A. / M.C.A.) programme will be decided by BOM as per the directives from Ministry of Human Resource Development (MHRD), Government of India and taking into account the market demands. Some seats for Non Resident Indians and a few seats for foreign nationals shall be made available.

2.2. The selected candidates will be admitted to the (M.TECH / M.B.A. / M.C.A.) programme after he/she fulfills all the admission requirements set by the Institute and after payment of the prescribed fees.

2.3. Candidates for admission to the first semester of the Master's Degree Programme shall be required to have passed an appropriate Degree Examination recognized by Hindustan University.

2.4. In all matters relating to admission to the (M.TECH /M.B.A. / M.C.A.). Programme, the decision of the Institute and its interpretation given by the Chancellor of the Institute shall be final.

2.5. If at any time after admission, it is found that a candidate has not fulfilled any of the requirements stipulated by the Institute, the Institute may revoke the admission of the candidate with information to the Academic Council.

3. Structure of the programme

3.1. The programme of instruction will have the following structure

i) Core courses of Engineering / Technology / Management.

ii) Elective courses for specialization in areas of student's choice.

3.2. The minimum durations of the programmes are as given below:

Program	No. of Semesters
M.Tech.(Full-Time)	4
M.Tech.(Part -Time)	6
M.B.A. (Full - Time)	4
M.B.A. (Part - Time)	6
M.C.A.(Full - Time)	6
M.C.A.(Part -Time)	8

Every (M.TECH / M.B.A. / M.C.A.) programme will have a curriculum and syllabi for the courses approved by the Academic Council.

3.3. Each course is normally assigned certain number of credits. The following norms will generally be followed in assigning credits for courses.

- One credit for each lecture hour per week per semester;
- One credit for each tutorial hour per week per semester;
- One credit for each laboratory practical (drawing) of three (two) hours per week per semester.
- One credit for 4 weeks of industrial training and
- One credit for 2 hours of project per week per semester.

3.4. For the award of degree, a student has to earn certain minimum total number of credits specified in the curriculum of the relevant branch of study. The curriculum of the different programs shall be so designed that the minimum prescribed credits required for the award of the degree shall be within the limits specified below.

Program	Minimum prescribed credit range
M.Tech. (Full time / Part time)	75 - 85
M.B.A. (Full time / Part time)	85 - 95
M.C.A (Full time / Part time)	115 - 125

3.5. The medium of instruction, examination and the language of the project reports will be English.

4. Faculty Advisor

4.1. To help the students in planning their courses of study and for getting general advice on the academic programme, the concerned Department will assign a certain number of students to a Faculty member who will be called their Faculty Advisor.

5. Class Committee

5.1 A Class Committee consisting of the following will be constituted by the Head of the Department for each class:

(i) A Chairman, who is not teaching the class.

(ii) All subject teachers of the class.

(iii) Two students nominated by the department in consultation with the class.

The Class Committee will meet as often as necessary, but not less than three times during a semester.

The functions of the Class Committee will include:

(i) Addressing problems experienced by students in the classroom and the laboratories.

(ii) Analyzing the performance of the students of the class after each test and finding ways and means of addressing problems, if any.

(iii) During the meetings, the student members shall express the opinions and suggestions of the class

students to improve the teaching / learning process.

6. Grading

6.1 A grading system as below will be adhered to.

6.2 GPA & CGPA

GPA is the ratio of the sum of the product of the number of credits C_i of course "i " and the grade points P_i earned for that course taken over all courses "i" registered by the student to the sum of C_i for all "i ". That is,

Range of Marks	Letter Grade	Grade points
95-100	S	10
85 - 94	A	09
75- 84	B	08
65-74	C	07
55-64	D	06
50-54	E	05
< 50	U	00
	I (Incomplete)	--

$$GPA = \frac{\sum_i C_i P_i}{\sum_i C_i}$$

CGPA will be calculated in a similar manner, at any semester, considering all the courses enrolled from first semester onwards.

6.3. For the students with letter grade I in certain subjects, the same will not be included in the computation of GPA and CGPA until after those grades are converted to the regular grades.

6.4 Raw marks will be moderated by a moderation board appointed by the Vice Chancellor of the University. The final marks will be graded using an absolute grading system. The Constitution and composition of the moderation board will be dealt with separately.

7. Registration and Enrollment

7.1 Except for the first semester, registration and enrollment will be done in the beginning of the semester as per the schedule announced by the University.

7.2 A student will be eligible for enrollment only if he/she satisfies regulation 10 (maximum duration of the programme) and will be permitted to enroll if (i) he/she has cleared all dues in the Institute, Hostel & Library up to the end of the previous semester and (ii) he/she is not debarred from enrollment by a disciplinary action of the University.

7.3. Students are required to submit registration form duly filled in.

8. Registration requirement

8.1. (i) A Full time student shall not register for less than 16 credits or more than 26 credits in any given semester.

8.1. (ii) A part time student shall not register for less than 10 credits or more than 20 credits in any given semester.

8.2 If a student finds his/her load heavy in any semester, or for any other valid reason, he/she may withdraw from the courses within three weeks of the commencement of the semester with the written approval of his/her Faculty Advisor and HOD. However the student should ensure that the total number of credits registered for in any semester should enable him/her to earn the minimum number of credits per semester for the completed semesters.

9. Minimum requirement to continue the programme

9.1. For those students who have not earned the minimum required credit prescribed for that particular semester examination, a warning letter to the concerned student and also to his parents regarding the shortage of his credit will be

sent by the HOD after the announcement of the results of the university examinations.

10. Maximum duration of the programme

The minimum and maximum period for the completion of various programs are given below.

Program	Min. No. of Semesters	Max. No. of Semesters
M.Tech (Full - time)	4	8
M.Tech (Part - time)	6	10
M.B.A. (Full Time)	4	8
M.B.A. (Part Time)	6	10
M.C.A. (Full - Time)	6	12
M.C.A (Part –Time)	8	14

11. Temporary discontinuation

11.1. A student may be permitted by the Director(Academic) to discontinue temporarily from the programme for a semester or a longer period for reasons of ill health or other valid reasons. Normally a student will be permitted to discontinue from the programme only for a maximum duration of two semesters.

12. Discipline

12.1. Every student is required to observe discipline and decorum both inside and outside the campus and not to indulge in any activity which will tend to bring down the prestige of the University.

12.2. Any act of indiscipline of a student reported to the Director(Academic) will be referred to a Discipline Committee so constituted. The Committee will enquire into the charges and decide on suitable punishment if the charges are substantiated. The committee will also authorize the Director(Academic) to recommend to the Vice - Chancellor the implementation of the decision. The student concerned may appeal to the Vice Chancellor whose decision will be final. The Director(Academic) will report the action taken at the next meeting of the Council.

12.3. Ragging and harassment of women are strictly prohibited in the University campus and hostels.

13. Attendance

13.1. A student whose attendance is less than 75% is not eligible to appear for the end semester examination for that semester. The details of all students who have attendance less than 75% will be announced by the teacher in the class. These details will be sent to the concerned HODs and Dean.

13.2. Those who have less than 75% attendance will be considered for condonation of shortage of attendance. However a condonation of 10% in attendance will be given on medical reasons. Application for condonation recommended by the Faculty Advisor, concerned faculty member and the HOD is to be submitted to the Director(Academic) who, depending on the merits of the case, may permit the student to appear for the end semester examination. A student will be eligible for this concession at most in two semesters during the entire degree programme. Application for medical leave, supported by medical certificate with endorsement by a Registered Medical Officer, should reach the HOD within seven days after returning from leave or, on or before the last instructional day of the semester, whichever is earlier.

13.3. As an incentive to those students who are involved in extracurricular activities such as representing the University in Sports and Games, Cultural Festivals, and Technical Festivals, NCC/

NSS events, a relaxation of up to 10% attendance will be given subject to the condition that these students take prior approval from the officer –in-charge. All such applications should be recommended by the concerned HOD and forwarded to Director(Academic) within seven instructional days after the programme/activity.

14. Assessment Procedure

14.1. The Academic Council will decide from time to time the system of tests and examinations in each subject in each semester.

14.2. For each theory course, the assessment will be done on a continuous basis as follows:

Test / Exam	Weightage	Duration of Test / Exam
First Periodical Test*	10%	2 Periods
Second Periodical Test*	10%	2 Periods
Model exam	20%	3 hours
Seminar/ Assignments/Quiz	20%	
End – semester examination	50%	3 Hours

*** Best out of the two tests will be considered.**

14.3. For practical courses, the assessment will be done by the subject teachers as below:

(i) Weekly assignment/Observation note book / lab records – weightage 60%.
(ii) End semester examination of 3 hours duration including viva – weightage 40%.

15. Make up Examination/model examination

15.1. Students who miss the end-semester examinations / model examination for valid reasons are eligible for make-up examination /model examination. Those who miss the end-semester examination / model examination should apply to the Head of the Department concerned within five days after he / she missed examination, giving reasons for absence.

15.2 Permission to appear for make-up examination / model exam will be given under exceptional circumstances such as admission to a hospital due to illness. Students should produce a medical certificate issued by a Registered Medical Practitioner certifying that he/she was admitted to hospital during the period of examination / model exam and the same should be duly endorsed by parent / guardian and also by a medical officer of the University within 5 days.

16. Project evaluation

16.1. For Project work, the assessment will be done on a continuous basis as follows:

Review / Examination	Weightage
First Review	10%
Second Review	20%
Third Review	20%
End semester Examination	50%

For end semester exam, the student will submit a Project Report in a format specified by the Director(Academic). The first three reviews will be conducted by a Committee constituted by the Head of the Department. The end – semester examination will be conducted by a Committee constituted by the Controller of Examinations. This will include an external expert.

17. Declaration of results

17.1 A candidate who secures not less than 50% of total marks prescribed for a course with a minimum of 50% of the

marks prescribed for the end semester examination shall be declared to have passed the course and earned the specified credits for the course.

17.2 After the valuation of the answer scripts, the tabulated results are to be scrutinized by the Result Passing Boards of PG programmes constituted by the Vice-Chancellor. The recommendations of the Result Passing Boards will be placed before the Standing Sub Committee of the Academic Council constituted by the Chancellor for scrutiny. The minutes of the Standing Sub Committee along with the results are to be placed before the Vice-Chancellor for approval. After getting the approval of the Vice-Chancellor, the results will be published by the Controller of Examination/Registrar.

17.3 If a candidate fails to secure a pass in a course due to not satisfying the minimum requirement in the end semester examination, he/she shall register and re-appear for the end semester examination during the following semester. However, the sessional marks secured by the candidate will be retained for all such attempts.

17.4 If a candidate fails to secure a pass in a course due to insufficient sessional marks though meeting the minimum requirements of the end semester examination, wishes to improve on his/her sessional marks, he/she will have to register for the particular course and attend the course with permission of the HOD concerned and the Registrar. The sessional and external marks obtained by the candidate in this case will replace the earlier result.

17.5 A candidate can apply for the revaluation of his/her end semester examination answer paper in a theory course within 2 weeks from the declaration of the results, on payment of a prescribed fee through proper application to the Registrar/Controller of Examinations through the Head of the Department. The Registrar/ Controller of Examination will arrange for the revaluation and the results will be intimated to the candidate concerned through the Head of the Department. Revaluation is not permitted for practical courses and for project work.

18. Grade Card

18.1. After results are declared, grade sheet will be issued to each student, which will contain the following details:

(i) Program and branch for which the student has enrolled.
(ii) Semester of registration.
(iii) List of courses registered during the semester and the grade scored.
(iv) Semester Grade Point Average (GPA)
(v) Cumulative Grade Point Average (CGPA).

19. Class / Division

19.1 Classification is based on CGPA and is as follows:
CGPA≥8.0: **First Class with distinction**
6.5 ≤CGPA < 8.0: **First Class**
5.0 ≤CGPA < 6.5: **Second Class.**

19.2 (i) Further, the award of 'First class with distinction' is subject to the candidate becoming eligible for the award of the degree having passed the examination in all the courses in his/her first appearance within the minimum duration of the programme.

(ii) The award of 'First Class' is further subject to the candidate becoming eligible to the award of the degree having passed the examination in all the courses within the below mentioned duration of the programme.

Program	No. of Semesters
M.Tech (Full - time)	5
M.Tech (Part - time)	7
M.B.A. (Full Time)	5
M.B.A. (Part Time)	7
M.C.A. (Full - Time)	7
M.C.A (Part –Time)	9

(iii) The period of authorized discontinuation of the programme (vide clause 11.1) will not be counted for the purpose of the above classification.

20. Transfer of credits

20.1. Within the broad framework of these regulations, the Academic Council, based on the recommendation of the transfer of credits committee so constituted by the Chancellor may permit students to earn part of the credit requirement in other approved institutions of repute and status in the country or abroad.

21. Eligibility for the award of (M.TECH / M.B.A. / M.C.A.) Degree

21.1. A student will be declared to be eligible for the award of the (M.TECH / M.B.A. / M.C.A.) Degree if he/she has

i) registered and successfully credited all the core courses,
ii) successfully acquired the credits in the different categories as specified in the curriculum corresponding to the discipline (branch) of his/her study within the stipulated time,
iii) has no dues to all sections of the Institute including Hostels, and
iv) has no disciplinary action pending against him/her.

The award of the degree must be recommended by the Academic Council and approved by the Board of Management of the University.

22. Power to modify

22.1. Notwithstanding all that has been stated above, the Academic Council has the right to modify any of the above regulations from time to time subject to approval by the Board of Management.

HINDUSTAN INSTITUTE OF TECHNOLOGY AND SCIENCE

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
M.Tech. COMPUTER SCIENCE AND ENGINEERING
with specialisation in PERVASIVE COMPUTING
CURRICULUM 2014-2015

Sl. No	Course Code	Course Title	L	T	P	C	TCH
SEMESTER I							
Theory							
1.	PMA105	Applied Mathematics for Computer Science*	4	0	0	4	4
2.	PCE101	Advanced Data Structures and Algorithms$$	4	0	0	4	4
3	PCE102	Advanced Computer Architecture***	4	0	0	4	4
4.	PCE103	Advanced Data Base Management Systems$$	4	0	0	4	4
5.	PIT101	Advanced Data Communications$$$	4	0	0	4	4
6.	PIT102	Internet Programming**	4	0	0	4	4
Practical							
7.	PCE104	Advanced Data Structures and Algorithms Lab	0	0	3	1	3
8.	PIT103	Internet Programming Lab$	0	0	3	1	3
Total						**26**	**30**
SEMESTER II							
Theory							
1.	PCE201	Advanced System Software	4	0	0	4	4
3.	PCE202	Information Security	4	0	0	4	4
4.	PCE203	Soft Computing	4	0	0	4	4
2.	PIT204	Advanced Operating Systems$	4	0	0	4	4
5.	**	Elective-I	4	0	0	4	4
6.	**	Elective-II	4	0	0	4	4
Practical							
7.	PCE204	Advanced System Software Lab	0	0	3	1	3
8.	PIT206	Advanced Operating Systems Lab	0	0	3	1	3
Total						**26**	**30**
SEMESTER III							
Theory							
1.	**	Elective-III	4	0	0	4	4
2.	**	Elective-IV	4	0	0	4	4
3.	**	Elective-V	4	0	0	4	4
Practical							
4.	PCE301	Project Phase-I	0	0	12	6	12
Total						**18**	**24**
SEMESTER IV							
Practical							
1.	PCE401	Project Phase-II	0	0	24	12	24
Total						**12**	**24**

Total Credits: 82

***- Common to M.Tech. CSE, M.Tech. SE**

****- Common to M.Tech. CSE, M.Tech. CCE, M.Tech. IT**

*****- Common to M.Tech. CSE, M.Tech. CCE, M.Tech. IT, M.Tech. Embedded**

$- **Common to M.Tech. CSE, M.Tech. IT**

$$- **Common to M.Tech. CSE, M.Tech. SE, M.Tech. IT**

$$$- **Common to M.Tech. CSE, M.Tech. SE, M.Tech. CCE, M.Tech. IT, M.Tech. CS**

ELECTIVES

Sl. No	Semester	Course Code	Course Title	L	T	P	C	TCH
1	II	PCE801	Wireless Sensor Networks	4	0	0	4	**4**
2		PCE802	Ubiquitous Computing	4	0	0	4	**4**
3	III	PCE803	Programming with Embedded Systems	4	0	0	4	**4**
4		PCE804	Real Time Systems	4	0	0	4	**4**
5		PCE805	Nano Computing	4	0	0	4	**4**

SEMESTER I

PMA105	APPLIED MATHEMATICS FOR COMPUTER SCIENCE	L T P C 4 0 0 4
GOAL	To create the awareness and comprehensive knowledge in probability functions random processes and queuing theory. Develop the Mathematical models using simulation techniques in practical problems, solve them and physically interpret the results.	

OBJECTIVES	OUTCOMESs
The course should enable the students to: 1. Understand the concept of random variable, moment generating function and their properties. Learns standard distributions in discrete and continuous cases and their applications. 2. Learn classification of random process and Markov chain processes. Understands Wide sense stationary and ergodic and Poisson processes. 3. Learn the different Markovian models with finite and infinite capacity and understands to classify them. 4. Understand simulation, its different types and their limitations. Learns generation of random numbers. 5. Understand the representation of graphs and the different types of algorithm to find the shortest route, minimal spanning trees and search.	The students should be able to: 1. Verify whether a given function is a probability mass or density function. Apply the discrete and continuous distributions for solving practical problems. Evaluates the moments of the distributions using moment generating function. Evaluate the probability using marginal and conditional distributions. 2. Verify whether a process is of first or second order or wide sense or strictly stationary. Classify the properties of Markov processes and verifies the properties of Poisson process. 3. Analyze and classify the models, M / M / 1, M / M / C, finite and infinite capacity and be able solve practical problems related to the queuing models. 4. Apply simulation techniques to generate random numbers and solves queuing problems. 5. Use Kruskal and prim's algorithm to find the minimal spanning tree of a graph. Evaluates the shortest path of a network using Dijskra's and Floyd algorithm. Use search algorithm to find the death and birth.

UNIT I PROBABILITY AND RANDOM VARIABLE 12

Discrete and Continuous random variables – Moments – Moment generating functions - Standard distributions - Binomial, Poisson, Geometric, Negative Binomial, Uniform, Normal ,Exponential, Gamma and Weibull distributions – Two dimensional random variables – Joint, Marginal and Conditional distributions.

UNIT II RANDOM PROCESSES 12

Classification – Stationary Process – Markov process – Markov Chains – Auto correlation – Cross correlation – Wide Sense Stationary – Ergodic process – Transition Probabilities – Limiting distribution – Poisson process – Properties.

UNIT III QUEUING THEORY 12

Markovian models – Birth and death queuing models – Steady state – Single and Multiple servers – M/M/1 – Finite and infinite capacity – M/M/C – finite and infinite capacity.

UNIT IV SIMULATION AND APPLICATIONS 12

Introduction - Types of simulation - Limitations of simulation techniques - Phases of simulation study – Generation of random numbers – Monte Carlo simulation – Application to queuing problems.

UNIT V GRAPH ALGORITHMS 12

Computer Representation of graphs - Basic graph algorithms - Minimal spanning tree algorithm - Kruskal and Prim's algorithm - Shortest path algorithms - Dijsktra's and Floyd algorithms - DFS and BFS algorithms.

Total 60

REFERENCE BOOKS

1) Taha, H.A., "Operations Research - An Introduction ", Prentice Hall of India Ltd., 6th Edition, New Delhi, 1997.
2) Dr.A.Singaravelu, Dr.S.Siva Subramanian and Dr.C.Ramachandran, "Probability and Queuing Theory", Meenakshi agency, 20th edition, January 2013.
3) Narsingh Deo, Graph Theory with Applications to Engineering and Computer Science, PHI, Unit III - Chapter 11.
4) Kapur, J.N. and Saxena, H.C. "Mathematical Statistics ", S. Chand and Co. Ltd., New Delhi, 1997.
5) Bhat, U.N., " Elements of Applied Stochastic Processes ", John Wiley and Sons, 2nd Edition,New York, 1984.
6) Sharma, S.D., "Operations Research ", Kedar Nath, Ram Nath and Co., Meerut, 12th Edition, 1998.
7) T.Veerarajan, "Probability, Statistics and Random Processes", Tata McGraw-Hill, second edition, 2006.

PCE101	ADVANCED DATA STRUCTURES AND ALGORITHMS	L T P C 4 0 0 4
GOAL	To study advanced concepts in Data structures and algorithm design techniques	

OBJECTIVES	OUTCOMES
The course should enable the students to: 1. Learn about the time complexity of algorithms. 2. Understand the representations used in heap data structures. 3. Learn the different types of search structures. 4. Learn various algorithm design techniques. 5. Understand the advanced data structures.	The student should be able to : 1. Calculate the time complexity of algorithms and express it using appropriate notations. 2. Compare and implement the different types of heap structures and search structures. 3. Implement the applications of heap and search structures in computer science. 4. Implement the real life problems using suitable algorithm design technique.

UNIT I INTRODUCTION **12**

Abstract Data Types - Time and Space Analysis of Algorithms - Big Oh and Theta Notations - Average, best and worst case analysis - Simple recurrence relations – Mappings.

UNIT II HEAP STRUCTURES **12**

Min-max heaps - Heaps - Leftist heaps -Binomial heaps - Fibonacci heaps - Skew heaps - Lazy-binomial heaps.

UNIT III SEARCH STRUCTURES **12**

Binary search trees - AVL trees - 2-3 trees - 2-3-4 trees - Red-black trees - B-trees - splay trees - Tries.

UNIT IVALGORITHM DESIGN TECHIQUES **12**

Divide and Conquer and Greedy : Quicksort - Strassen's matrix multiplication - Convex hull - Tree-vertex splitting - Job sequencing with deadlines - Optimal storage on tapes Dynamic Programming and Backtracking: Multistage graphs - 0/1 knapsack - 8- queens problem - graph coloring.

UNIT V ADVANCED ALGORITHMS **12**

Parallel Algorithms: Basic Techniques- Work & Efficiency - Distributed Computation - Heuristic & Approximation Approaches.

Total 60

REFERENCE BOOKS

1. E. Horowitz, S.Sahni and Dinesh Mehta, Fundamentals of Data structures in C++, University Press, 2009.
2. E. Horowitz, S. Sahni and S. Rajasekaran, Computer Algorithms/C++, Second Edition, University Press, 2007.
3. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C", Third Edition, Pearson Education, Asia.2007.
4. Jean-Paul Tremblay, Paul .G. Sorenson, "An Introduction to Data Structures with Applications", Tata McGraw Hill second edition , 1991.
5. Thomas H.Coremen, Charles E. Leiserson, Ronald L.Rivest, Clifford Stein , "Introduction to algorithms", Third edition, MIT press.

PCE102	ADVANCED COMPUTER ARCHITECTURE	L T P C 4 0 0 4
GOAL	To study the design principles of parallel models and parallel computing architectures.	

OBJECTIVES	OUTCOMES
The course should enable the students to: 1. Learn the parallel computer models. 2. Understand the processor technology and memory organizations. 3. Learn pipeline design techniques. 4. Learn the various parallel and scalable architectures. 5. Understand the multithreaded and data flow computing architectures.	The student should be able to 1. Identify and understand the parallel computer models. 2. Have thorough knowledge about the types of processors and memory hierarchy. 3. Design various pipeline architectures. 4. Design parallel and scalable architectures. 5. Analyze and understand multithreading and data flow computers.

UNIT I PARALLEL MODELS **12**

Parallel computer models: Multiprocessors and Multicomputer - Multi vector and SIMD computer, PRAM & VLSI models, conditions of parallelism, System interconnect architectures performance, Metrics and Measures.

UNIT II PROCESSORS AND MEMORY HIERARCHY **12**

Advanced processor technology - Super scalar and vector processors - Memory hierarchy technology, virtual memory technology - cache memory organization - shared – memory organization.

UNIT III PIPELINING AND SUPERSCALAR TECHNIQUES **12**

Linear pipeline processors - Nonlinear pipeline processors - Instruction pipeline design Arithmetic pipeline design - Superscalar pipeline design

UNIT IV PARALLEL AND SCALABLE ARCHITECTURE **12**

Multiprocessor system interconnects - Cache coherence, Vector processing principle Compound Vector processing, SIMD computer organization, multiprocessor operating system, multiprocessor examples

UNIT V SCALABLE, MULTITHREADED & DATA FLOW ARCHITECTURE **12**

Latency - Hiding techniques - Principles of Multithreading , Scalable and Multithreaded architectures. Dataflow computer, static data flow computer , Dynamic data flow compiler

Total 60

REFERENCE BOOKS

1. Kai Hwang, “Advanced Computer Architecture”, Parallelism, Scalability, Programmability, McGraw Hill, 2008.
2. Hwang Briggs, “Computer Architecture and parallel processing”, McGraw Hill, 1984.
3. William Stallings, “Computer Organization and Architecture- Designing for Performance”, Prentice Hall of India, 2009.

PCE103	ADVANCED DATABASE MANAGEMENT SYSTEMS	L T P C 4 0 0 4
GOAL	To provide a comprehensive study of Relational, Distributed and Advanced Database technologies.	

OBJECTIVES	OUTCOMES
The course should enable the students to: 1. Learn ER model and transaction processing system. 2. Understand various storage structures for Database. 3. Learn the distributed and Parallel database processing. 4. Learn Object Oriented database concepts and models. 5. Understand various advancements in database technology.	The student should be able to 1. Construct ER models and explain transaction processing in relational models. 2. Implement database storage models and query optimization techniques. 3. Design a distributed database. 4. Implement the distributed query processing systems.

UNIT I RELATIONAL MODEL ISSUES **12**

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

UNIT II DATABASE STORAGE AND SYSTEM DESIGN **12**

Storage Structures, Indexing and multi dimensional indexes, Query Processing Algorithms, External Sorting, Query Optimization- Heuristic based optimization- cost based optimization, Buffer Management, Concurrency Control, Recovery.

UNIT III DISTRIBUTED DATABASES **12**

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

UNIT IV OBJECT ORIENTED DATABASES **12**

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

UNIT V EMERGING SYSTEMS **12**

Enhanced Data Models - Client/Server Model - Web Databases - Mobile Databases- XML and Web Databases - current issues - Rules - Knowledge Bases - Active and Deductive Databases - Multimedia Databases- - Multimedia Query languages - Spatial Databases.

Total 60

REFERENCE BOOKS

1. Thomas Connolly and CarlolynBegg, “Database Systems, A Practical Approach to Design, Implementation and Management”, Fourth Edition, Pearson Education 2000.
2. R. Elmasri, S.B. Navathe, “Fundamentals of Database Systems”, Fifth Edition, Pearson Education, 2006.
3. Abraham Silberschatz, Henry F. Korth, S. Sudharshan, “Database System Concepts”, Sixth Edition, Tata McGraw Hill, 2006.
4. C.J.Date, A.Kannan, S.Swamynathan, “An Introduction to Database Systems”, Eighth Edition, Pearson Education, 2006.

PIT101	ADVANCED DATA COMMUNICATIONS	L T P C 4 0 0 4
GOAL	To provide wide knowledge in the design and implementation of communication systems.	

OBJECTIVES	OUTCOMES
The course should enable the students to: 1. Understand the concepts of TCP/IP, UDP and OSI architecture. 2. Understand ATM Services and Connection details. 3. Learn congestion control algorithms. 4. Learn graph methods to find the shortest path among different routing protocols. 5. Learn the key concepts of integrated service architecture and label switching.	The student should be able to 1. Compare and understand the different network architectures 2. Implement ATM protocol architecture and services. 3. Design techniques to control the congestion in the network. 4. Apply the different routing protocols to find the shortest path. 5. Design the ISA with the associated protocols.

UNIT I NETWORK ARCHITECTURES **12**

The need for speed and quality of service, Advanced TCP/IP and ATM Networks, The need for a protocol architecture, The TCP/IP protocol architecture, The OSI model, Internetworking, TCP, UDP, Ipv6.

UNIT II ATM NETWORKS **12**

Packet-switching networks, Frame relay networks, ATM protocol architecture, ATM logical connections, ATM cells, ATM service categories, ATM Adaptation Layer (AAL), The emergence of high-speed LANs, Ethernet, Fibre channel, Wireless LANs.

UNIT III TRAFFIC MANAGEMENT **12**

Congestion control in data networks and internets, Effects of congestion, Congestion and control, Traffic management, Congestion control in Packet-Switching networks, Frame relay congestion control, The need for flow and error control, Link control mechanisms, ARQ performance, TCP flow control, TCP congestion control performance of TCP over ATM.

UNIT IV ROUTING PROTOCOLS **12**

Overview of graph theory and least-cost paths, Elementary concepts of graph theory, Shortest path length determination, Internet routing principles, Distance-Vector protocol, RIP, Link-State protocol, OSPF, Path-Vector protocols, BGP and IDRP, Multicasting.

UNIT V ADVANCED NETWORKING CONCEPTS **12**

Integrated Services Architecture (ISA), Queuing discipline, Random early detection, Differentiated services, Real-Time traffic, Resource Reservation : RSVP, Multiprotocol label switching, Real-Time Transport Protocol (RTP).

Total 60

REFERENCE BOOKS

1. Willam Stallings, "High Speed Networks and Internets - Performance and Quality of Service", 2nd Ed., Pearson Education.
2. Andrew S. Tanenbaum, " Computer Networks", 4th Ed., Pearson Education.
3. James F. Kurose, Keith W. Ross, "Computer Networking: A Top-Down Approach featuring the Internet", 3rd edition,Pearson Education, 2005.
4. William Stallings, "Data and Computer Communications", 9th Ed., Pearson Education.

PIT102	INTERNET PROGRAMMING	L T P C 4 0 0 4
GOAL	To provide knowledge about the fundamental concepts in web technology and also about the various server side and client side programming languages.	

OBJECTIVES	OUTCOMES
The course should enable the students to: 1. Learn the basics of Internet architecture and basics of web designing 2. Learn the style sheets and client side scripting. 3. Understand web databases and connectivity. 4. Learn server side programming. 5. Learn PHP programming.	The student should be able to 1. Design simple web pages using HTML. 2. Do styling using CSS and do client side validation using java script. 3. Create XML document, present XML using CSS, connect to database using JDBC. 4. Develop web applications using SERVELETS and JSP using IDE. 5. Develop web applications using PHP.

UNIT I INTRODUCTION **12**

Introduction – Network concepts – Web System Architecture - concepts – Internet addresses – URL- System Overview of HTTP – HTTP Request Response – Generation of dynamic web pages – Cookies- Introduction, Basic HTML tags, Formatting and Fonts, Commenting Code, Anchors, Backgrounds, Images, Hyperlinks, Lists, Tables, Frames, simple HTML Forms, XHTML, Introduction to HTML5.

UNIT II STYLING & CLIENT SIDE SCRIPTING **12**

CSS: Introduction, Basic syntax and structure, using CSS, manipulating text, padding, lists, Positioning using CSS. Object model and Event model- Filters and Transitions-Active X Controls - What is JavaScript? How to develop Java Script Simple JavaScript, variables, functions, condition, JavaScript and Objects, JavaScript's own Objects, the DOM and the Web browser Environment, forms and validation.

UNIT III WEB DATABASES AND CONNECTIVITY **12**

XML – Introduction – Structuring Data – XML Namespaces – XML vocabularies – Presenting XML using CSS - Web server- Database Connectivity - Connecting to Databases – JDBC principles – Database access using javax.sql package.

UNIT IV DATABASE CONNECTIVITY AND SERVER SIDE PROGRAMMING **12**

Common Gateway Interface: Introduction to Java Servlets – overview and architecture – Handling HTTP get & post request – session Tracking – Using Servlet context – Dynamic Content Generation – Servlet Chaining and Communications- Demonstration of Java Servlets using Apache Tomcat Server – Demonstration of web development using IDE - – JSP – Overview – Objects – scripting – Standard Actions–Directives.

UNIT V INTRODUCTION TO PHP **12**

Basics- variables – strings - operators – control statements – Arrays - functions – forms – GET & POST – Date – File - Cookies – Sessions – Emails – Errors & Exceptions. PHP Database: Connecting Databases – Executing Queries – Demonstration of PHP using Xampp/Wamp Servers.

Total 60

REFERENCE BOOKS

1. Deitel, Deitel and Neito,"Internet and World Wide Web–How to program", Pearson, 4th Ed, 2009.
2. Elliotte Rusty Herold, "Java Network Programming", O'Reilly Publications, 3rd Edition, 2004.
3. Jeffy Dwight, Michael Erwin and Robert Nikes "USING CGI", PHI Publications, 1997
4. Jason Hunter, William Crawford "Java Servlet Programming" O'Reilly, 2nd Edition, 2001.
5. Eric Ladd and Jim O'Donnell, et al, "Using HTML 4, XML, and JAVA1.2", Prentice Hall, 2003.
6. Bruce Lawson and Remy Sharp, "Introducing HTML5", New Riders; Second edition, 2012

PCE104	ADVANCED DATA STRUCTURES AND ALGORITHMS LAB	L T P C 0 0 3 1
GOAL	To implement advanced Data structures and applications using algorithms in C++	

OBJECTIVES	OUTCOMES
The course should enable the students to 1. Implement advanced data structures 2. Calculate the time complexity of algorithms and express it using appropriate notations 3. Implement different algorithm design techniques. 4. Implement advanced algorithm concepts.	The student should be able to 1. Use the appropriate data structures for applications in computer science. 2. Understand the applications of heap and search structures in computer science. 3. Calculate the time complexity of algorithms.

LIST OF EXPERIMENTS

1. Implement min - max heap and calculate the efficiency of the algorithms.
2. Implement Heap data structure.
3. Implement splay trees.
4. Implement the insertion of AVL trees with rotations.
5. Implement B-Tree.
6. Implement vertex cover problem using approximate algorithms.
7. Implement and calculate the time complexity of quick sort
8. Implement Convex hull
9. Implement 0/1 Knapsack using Dynamic Programming
10. Implement Graph coloring using backtracking

PIT103	INTERNET PROGRAMMING LAB	L T P C 0 0 3 1
GOAL	To understand various technologies involved in internet	

OBJECTIVES	**OUTCOMESs**
The course should enable the students to: 1. Understand web designing using HTML 2. Understand the usage of CSS 3. Understand the concept of client side validation 4. Understand the concept of database connectivity 5. Understand server side programming	The student should be able to 1. Design a HTML web page using dream weaver 2. Apply a thematic design to all the web pages of a web site 3. Do client side validation using java script 4. Connect a java program to a database using JDBC. 5. Do server side programming using java Servlet and PHP.

LIST OF EXPERIMENTS

1. Design a web for mail authentication using HTML to demonstrate the use of Frames, Forms, Tables and Images.

2. Design a web page for a university which contains links for each department, also design separate web pages of every department. Follow same color theme for all the web pages. Use CSS for styling the web page.

3. Design a web page for an online job portal registration and validate the inputs for mandatory fields including the format of the email id using java script.

4. Write a Servlet program to read the roll no of a student and display all the details along with the photograph of the student, use LONGBLOB data type to store images in database.

5. Write a program using java Servlet to publish university exam results in internet.

6. Write a Program using JSP to register a new customer in your online shopping database.

7. Create a table in PHP and embed it in a web page.

8. Create a used id authentication form using PHP. Use calendar function, time function, session variable and bring the session data in all the pages.

9. Write a program to retrieve the data from database and display it in the table format using PHP

SEMESTER II

PCE201	ADVANCED SYSTEM SOFTWARE	L T P C 4 0 0 4
GOAL	To provide a sound technical exposure to the concepts, principles, methods, and best practices in the design of system software.	

OBJECTIVES	OUTCOMES
The course should enable the students to: 1. Learn the basic concepts of compilers and device drivers. 2. Study symbol table structure and intermediate code optimization techniques. 3. Understand memory optimization and code sharing. 4. Understand virtual machine concept. 5. Learn emulation techniques and binary translations.	The student should be able to 1. Explain fundamental concepts of compilers. 2. Construct symbol table. 3. Implement various optimization techniques. 4. Design Virtual machine. 5. Develop small compiler and device drivers.

UNIT I INTRODUCTION TO ADVANCED SYSTEM SOFTWARE **12**

Basic Compiler Functions – Grammars – Lexical Analysis – Syntactic Analysis – Code Generation – Heap Management – Parameter Passing Methods – Semantics of Calls and Returns – Implementing Subprograms – Stack Dynamic Local Variables – Dynamic binding of method calls to methods – Overview of Memory Management, Virtual Memory, Process Creation – Overview of I/O Systems, Device Drivers, System Boot.

UNIT II SYMBOL TABLE MANGEMENT AND INTERMEDIATE CODE **12**

Introduction and Overview – Symbol table structure – Local and Global Symbol table management Intermediate representation – Issues – High level, medium level, low level intermediate languages – MIR, HIR, LIR – ICAN for Intermediate code – Optimization –Early optimization–loop optimization.

UNIT III CODE OPTIMIZATION **12**

Procedure optimization – in-line expansion – leaf routine optimization and shrink wrapping – register allocation and assignment – graph coloring – data flow analysis – constant propagation – alias analysis – register allocation – global references – Optimization for memory hierarchy - Code Scheduling – Instruction scheduling – Speculative scheduling – Software pipelining – trace scheduling – Run-time support – Register usage – local stack frame – run-time stack – Code sharing – position–independent code.

UNIT IV VIRTUAL MACHINE **12**

Introduction to Virtual Machines (VM) – Pascal P-Code VM – Object-Oriented VMs – Java VM Architecture – Common Language Infrastructure – Dynamic Class Loading – Security – Garbage Collection – Optimization.

UNIT V EMULATION AND BINARY TRANSLATION **12**

Emulation – Interpretation and Binary Translation – Instruction Set Issues – Process Virtual Machines – Profiling – Migration – Grids – Examples of real world implementations of system software.

Total 60

REFERENCE BOOKS

1. Steven S. Muchnick, "Advanced Compiler Design Implementation", Morgan Koffman, 2004.
2. James E Smith and Ravi Nair, "Virtual Machines", Elsevier, 2005.
3. Robert W. Sebesta, "Concepts of Programming Languages", 8th, Pearson Education, 2009.
4. Alfred V Aho, Ravi Sethi, Jeffrey D Ullman, "Compilers: Principles, Techniques and Tools ", 2nd edition, Pearson Education.

PCE202	INFORMATION SECURITY	L T P C 4 0 0 4
GOAL	To study the critical need for ensuring Information Security in Organizations	

OBJECTIVES	OUTCOMESs
The course enable the student to:	The student should be able to
1. Learn the basics of Information Security	1. Understand the basics of information security
2. Understand legal, ethical and professional issues in Information Security	2. Explain the legal, ethical and professional issues in Information Security
3. Understand the aspects of risk management	3. Explain Risk management
4. Learn various security standards.	4. Design the logic of various standards
5. Learn the technological aspects of Information Security	5. Implement Information Security procedures.

UNIT 1 INTRODUCTION **12**

History, What is Information Security?, Critical Characteristics of Information, NSTISSC Security Model, Components of an Information System, Securing the Components, Balancing Security and Access, The SDLC, The Security SDLC.

UNIT II SECURITY INVESTIGATION **12**

Need for Security, Business Needs, Threats, Attacks, Legal, Ethical and Professional Issues.

UNIT III SECURITY ANALYSIS **12**

Risk Management: Identifying and Assessing Risk, Assessing and Controlling Risk.

UNIT IV LOGICAL DESIGN **12**

Blueprint for Security, Information Security Policy, Standards and Practices, ISO 17799/BS 7799, NIST Models, VISA International Security Model, Design of Security Architecture, Planning for Continuity.

UNIT V PHYSICAL DESIGN **12**

Security Technology, IDS, Scanning and Analysis Tools, Cryptography, Access Control Devices, Physical Security, Security and Personnel.

Total 60

REFERENCE BOOKS

1. Michael E Whitman and Herbert J Mattord, "Principles of Information Security", Vikas Publishing House, New Delhi, 2012.
2. Micki Krause, Harold F. Tipton, " Handbook of Information Security Management", Vol 1-3 CRC Press LLC, 2004.
3. Stuart Mc Clure, Joel Scrambray, George Kurtz, "Hacking Exposed", Tata McGraw-Hill, 2003.
4. Matt Bishop, " Computer Security Art and Science", Pearson/PHI, 2002.

PCE203	SOFT COMPUTING	L T P C 4 0 0 4
GOAL	To provide knowledge about soft computing techniques.	

OBJECTIVES	OUTCOMES
The course should enable the students to:	The student should be able to
1. Learn the concepts of fuzzy sets, fuzzy logic and heuristics based on human experience. 2. Understand neural network techniques. 3. Learn the mathematical background for carrying out optimization associated with neural network learning. 4. Learn genetic algorithms and random search procedures. 5. Learn the methodologies to illustrate the intelligent behaviour of programs based on soft computing	1. Apply concepts of fuzzy sets, fuzzy logic and heuristics based systems. 2. Derive appropriate rules for inference systems. 3. Have understood the mathematical background to optimize neural network learning. 4. Implement optimization algorithms and random search procedures useful to seek global optimum in self-learning. 5. Develop case studies to illustrate the intelligent behaviour of programs based on soft computing

UNIT I FUZZY SET THEORY **12**

Introduction to Neuro – Fuzzy and Soft Computing – Fuzzy Sets – Basic Definition and Terminology – Set-theoretic Operations – Member Function Formulation and parameterization – Fuzzy Rules and Fuzzy Reasoning – Extension Principle and Fuzzy Relations – Fuzzy If-Then Rules – Fuzzy Reasoning – Fuzzy Inference Systems – Mamdani Fuzzy Models – Sugeno Fuzzy Models – Tsukamoto Fuzzy Models – Input Space Partitioning and Fuzzy Modeling.

UNIT II OPTIMIZATION **12**

Derivative-based Optimization – Descent Methods – The Method of Steepest Descent – Classical Newton's Method – Step Size Determination – Derivative-free Optimization – Genetic Algorithms – Simulated Annealing – Random Search – Downhill Simplex Search- Particle Swarm Techniques - Ant Colony Optimization.

UNIT III NEURAL NETWORKS **12**

Supervised Learning Neural Networks – Perceptrons - Adaline – Back propagation Mutilayer Perceptrons – Radial Basis Function Networks – Unsupervised Learning Neural Networks – Competitive Learning Networks – Kohonen Self-Organizing Networks – Learning Vector Quantization – Hebbian Learning.

UNIT IV NEURO FUZZY MODELING **12**

Adaptive Neuro-Fuzzy Inference Systems – Architecture – Hybrid Learning Algorithm – Learning Methods that Cross-fertilize ANFIS and RBFN – Coactive Neuro Fuzzy Modeling – Framework Neuron Functions for Adaptive Networks – Neuro Fuzzy Spectrum.

UNIT V APPLICATIONS OF COMPUTATIONAL INTELLIGECE **12**

Printed Character Recognition – Inverse Kinematics Problems – Automobile Fuel Efficiency Prediction – Soft Computing for Color Recipe Prediction.

Total 60

REFERENCE BOOKS

1. J.S.R.Jang, C.T.Sun and E.Mizutani, "Neuro-Fuzzy and Soft Computing", Pearson, 2004.
2. Timothy J.Ross, "Fuzzy Logic with Engineering Applications", McGraw-Hill, 1997.
3. Davis E.Goldberg, "Genetic Algorithms: Search, Optimization and Machine Learning", Addison Wesley, 2009.
4. S. Rajasekaran and G.A.V.Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithms", PHI, 2003.

PIT204	ADVANCED OPERATING SYSTEMS	L	T	P	C
		4	0	0	4
GOAL	To provide a wide knowledge in the design and implementation of distributed operating systems.				

OBJECTIVES	OUTCOMES
The course should enable the students to:	The student should be able to
1. Learn the architecture, theoretical foundations and mutual exclusion concepts.	1. Design distributed operating system.
2. Understand the concept of deadlock and various protocols.	2. Detect, prevent and avoid the deadlocks in distributed environment.
3. Learn load distribution and error recovery.	3. Explain the need for load distribution and the corresponding techniques.
4. Understand fault tolerance and security mechanisms.	4. Design security mechanisms for distributed operating system.
5. Learn multiprocessor architecture and database operating systems.	5. Analyze and find out the requirements to construct a database operating systems

UNIT I DISTRIBUTED OPERATING SYSTEM **12**

Synchronization Mechanisms: Introduction – concept of a process – concurrent process – the critical section problem – Synchronization problems – language mechanisms for synchronization: Monitors. System Architecture types – issues in distributed operating systems – communication networks – communication primitives. Theoretical Foundations: inherent limitations of a distributed system – lamports logical clocks – vector clocks – casual ordering of messages – global state – cuts of a distributed computation – termination detection.

UNIT II DISTRIBUTED DEADLOCK DETECTION **12**

Deadlock handling strategies in distributed systems – issues in deadlock detection and resolution – control organizations for distributed deadlock detection – centralized and distributed deadlock detection algorithms –hierarchical deadlock detection algorithms. Agreement protocols – introduction-the system model, a classification of agreement problems, solutions to the Byzantine agreement problem, applications of agreement algorithms.

UNIT III DISTRIBUTED SHARED MEMORY **12**

Architecture– algorithms for implementing DSM – memory coherence and coherence protocols – design issues. Distributed Scheduling: introduction – issues in load distributing – components of a load distributing algorithm – stability – load distributing algorithm – performance comparison – selecting a suitable load sharing algorithm – requirements for load distributing -task migration and associated issues. Failure Recovery and Fault tolerance: introduction – basic concepts – classification of failures – backward and forward error recovery approaches - recovery in concurrent systems – synchronous and asynchronous check pointing and recovery – check pointing for distributed database systems - recovery in replicated distributed databases systems.

UNIT IV MULTIPROCESSOR OPERATING SYSTEM **12**

Basic multiprocessor system architectures – basic multiprocessor system architecture - inter connection networks for multiprocessor systems – caching – hypercube architecture – structures of multiprocessor operating system - operating system design issues – threads - process synchronization – processor scheduling – Memory management. The mach OS.

UNIT V DATABASE OPERATING SYSTEM **12**

Requirements of a database operating system Concurrency control : theoretical aspects - introduction, database systems - a concurrency control model of database systems- the problem of concurrency control - Serializability theory- distributed database systems, concurrency control algorithms - introduction, basic synchronization primitives, lock based algorithms-timestamp based algorithms, optimistic algorithms - concurrency control algorithms, data replication.

Total 60

REFERENCE BOOKS

1. Mukesh Singhal, NiranjanG.Shivaratri, "Advanced concepts in operating systems: Distributed, Database and multiprocessor operating systems", TMH, 2001
2. Abraham Silberschatz, Peter B. Galvin, G. Gagne, “Operating System Concepts”, Nineth Edition, Addison Wesley Publishing Co., 2013.
3. Andrew S.Tanenbaum, "Modern operating system", PHI, 3rd edition,2008
4. Pradeep K.Sinha, "Distributed operating system-Concepts and design", PHI, 2003.
5. Andrew S.Tanenbaum, "Distributed operating system", Pearson education, 2003

PCE204	ADVANCED SYSTEM SOFTWARE LAB	L T P C 0 0 3 1
GOAL	To implement various phases of Compilers.	

OBJECTIVES	OUTCOMES
The course should enable the students to 1. Implement Lexical analyser with /without using lex tool. 2. Implement the parser with/without using yacc tool. 3. Implement various intermediate representation in ICAN. 4. Implement simple virtual machine.	The student should be able to 1. design lexical analyser with/without using Lex tool. 2. Design parser with/without using the tool Yacc. 3. Understand the various IRs in ICAN. 4. Design a Virtual machine. .

LIST OF EXPERIMENTS

1. Design a simple Lexical analyzer.
2. Implement the lexical analyzer using Lex, flex or other lexical analyzer generating tools.
3. Design predictive parser for the given language
4. Design a LALR bottom up parser for the given language
5. Convert the BNF rules into Yacc form and write code to generate abstract syntax tree.
6. Write a program to represent HIR,MIR and LIR in ICAN.
7. Create a simple Virtual Machine.

SYSTEM REQUIREMENTS

1. Intel based desktop PC of 166MHz or faster processor with at least 64 MB RAM and 100 MB free disk space.
2. C++ compiler and JDK kit.

PIT206	ADVANCED OPERATING SYSTEMS LAB	L T P C 0 0 3 1
GOAL	To design and implement distributed operating systems.	

OBJECTIVES	OUTCOMES
The course should enable the students to 1. Implement Semaphores. 2. Implement Multithreading concepts. 3. Develop Network Operating System 4. Develop Real Time Operating System 5. Develop Database Operating System	The students should be able to 1. Design and implement algorithms used in the OS.

LIST OF EXPERIMENTS

1. Semaphores -Multiprocessor Operating System
2. Multithreading - Multiprocessor Operating System
3. Multiple Sleeping Barbers - Multiprocessor Operating System
4. Network Operating System
5. Real Time Operating System
6. Transactions and Concurrency - Database Operating System
7. Deadlocks in Distributed Operating System

ELECTIVES

PCE801	WIRELESS SENSOR NETWORKS	L T P C 4 0 0 4
GOAL	To learn wireless sensor networks and its infrastructure design	

OBJECTIVES	**OUTCOMESs**
The course should enable the students to 1. Realize the challenges of wireless sensor networks. 2. Learn the architecture of WSN 3. Understand the networking of sensors to form the network. 4. Understand the infrastructure of WSN. 5. Learn programming of wireless sensor networks.	The students should be able to 1. Have an understanding of WSN concepts. 2. Design the WSN architecture. 3. Implement the Layer protocols 4. Establish the complete infrastructure. 5. Work with motes and simulators.

UNIT I OVERVIEW OF WIRELESS SENSOR NETWORKS **12**

Challenges for Wireless Sensor Networks, Enabling Technologies For Wireless Sensor Networks.

UNIT II ARCHITECTURES **12**

Single-Node Architecture - Hardware Components, Energy Consumption of Sensor Nodes , Operating Systems and Execution Environments, Network Architecture - Sensor Network Scenarios, Optimization GOALs and Figures of Merit, Gateway Concepts.

UNIT III NETWORKING SENSORS **12**

Physical Layer and Transceiver Design Considerations, MAC Protocols for Wireless Sensor Networks, Low Duty Cycle Protocols And Wakeup Concepts - S-MAC , The Mediation Device Protocol, Wakeup Radio Concepts, Address and Name Management, Assignment of MAC Addresses, Routing Protocols- Energy-Efficient Routing, Geographic Routing.

UNIT IV INFRASTRUCTURE ESTABLISHMENT **12**

Topology Control , Clustering, Time Synchronization, Localization and Positioning, Sensor Tasking and Control.

UNIT V SENSOR NETWORK PLATFORMS AND TOOLS **12**

Sensor Node Hardware – Berkeley Motes, Programming Challenges, Node-level software platforms, Node-level Simulators, State-centric programming.

TOTAL 60

REFERENCE BOOKS

1. Holger Karl & Andreas Willig, " Protocols And Architectures for Wireless Sensor Networks" , John Wiley, 2005.

2. Feng Zhao & Leonidas J. Guibas, “Wireless Sensor Networks- An Information Processing Approach", Elsevier, 2007.

3. Kazem Sohraby, Daniel Minoli, & Taieb Znati, “Wireless Sensor Networks-Technology, Protocols, And Applications”, John Wiley, 2007.

4. Anna Hac, “Wireless Sensor Network Designs”, John Wiley, 2003.

PCE802	UBIQUITOUS COMPUTING	L T P C 4 0 0 4
GOAL	To study pervasive computing techniques and its application.	

OBJECTIVES	OUTCOMES
The course should enable the students to:	The student should be able to
1. Learn the basic concepts of pervasive computing. 2. Understand various device technologies. 3. Learn device connectivity and Web Application Concepts. 4. Analyze WAP and voice technologies in detail. 5. Understand pervasive web application architecture.	1. Explain the history of pervasive computing and its applications. 2. Implement the hardware, software and interfaces. 3. Develop the web architecture for pervasive computing. 4. Design and develop WAP architecture, infrastructure and the security issues. 5. Have an understanding the PDA device categories, characteristics, software, browsers and various mobile applications.

UNIT I PERVASIVE COMPUTING **12**

Pervasive Computing: Past, Present and Future Pervasive Computing - Pervasive Computing Market - m-Business - Application examples: Retail, Airline check-in and booking - Sales force automation - Health care - Tracking - Car information system - E-mail access via WAP.

UNIT II DEVICE TECHNOLOGY **12**

Device Technology: Hardware - Human Machine Interfaces - Biometrics – Operating Systems - Java for Pervasive devices.

UNIT III DEVICE CONNECTIVITY AND WEB APPLICATION CONCEPTS **12**

Device Connectivity: Protocols - Security - Device Management Web Application Concepts: WWW architecture - Protocols - Transcoding - Client authentication via internet.

UNIT IV WAP AND VOICE TECHNOLOGY **12**

WAP and Beyond: Components of the WAP architecture - WAP infrastructure – WAP security issues - WML - WAP push - Products - i-Mode - Voice Technology: Basics of Speech recognition- Voice Standards - Speech applications - Speech and Pervasive Computing.

UNIT V PDA AND PERVASIVE WEB APPLICATION ARCHITECTURE **12**

PDA: Device Categories - PDA operation Systems - Device Characteristics – Software Components - Standards - Mobile Applications - PDA Browsers Pervasive Web Application architecture: Background - Scalability and availability - Development of Pervasive Computing web applications - Pervasive application architecture

Total 60

REFERENCE BOOKS

1. Jochen Burkhardt, Horst Henn, Stefan Hepper, Thomas Schaech & Klaus Rindtorff “Pervasive Computing, Technology and Architecture of Mobile Internet Applications”, Pearson Education, 2009.
2. Frank Adelstein, Sandeep KS Gupta, Golden Richard III, Loren Schwiebert, “Fundamentals of Mobile and Pervasive Computing”, McGraw Hill edition, 4th ed., 2008.

PCE803	PROGRAMMING WITH EMBEDDED SYSTEMS	L T P C 4 0 0 4
GOAL	To learn the types of programming languages with the hardware for the design and development of embedded system.	

OBJECTIVES	**OUTCOMESs**
The course should enable the student to 1. Expose the students to the fundamentals of embedded Programming. 2. Get introduced the GNU C Programming Tool Chain in Linux. 3. Study the basic concepts of embedded C and Embedded OS 4. Get introduced time driven architecture, Serial Interface with a case study. 5. Get introduced the concept of embedded Java for Web Enabling of systems.	The student should be able to 1. Have learnt C with assembly language programming 2. Work with GNU C programming tool chain 3. Have learnt embedded OS and embedded C programming 4. Do the case study in time driven architecture 5. Do programming in Java for web enabling of systems

UNIT I EMBEDDED PROGRAMMING 12

C and Assembly - Programming Style - Declarations and Expressions - Arrays, Qualifiers and Reading Numbers - Decision and Control Statements - Programming Process - More Control Statements - Variable Scope and Functions - C Preprocessor - Advanced Types – Simple Pointers - Debugging and Optimization – In-line Assembly.

UNIT II C PROGRAMMING TOOLCHAIN IN LINUX 12

C preprocessor - Stages of Compilation - Introduction to GCC - Debugging with GDB – The Make utility - GNU Configure and Build System - GNU Binary utilities - Profiling - using *gprof* Memory Leak Detection with *valgrind* - Introduction to GNU C Library

UNIT III EMBEDDED C AND EMBEDDED OS 12

Adding Structure to 'C' Code: Object oriented programming with C, Header files for Project and Port, Examples. Meeting Real-time constraints: Creating hardware delays - Need for timeout mechanism - Creating loop timeouts - Creating hardware timeouts. Creating embedded operating system: Basis of a simple embedded OS, Introduction to sEOS, Using Timer 0 and Timer 1, Portability issue, Alternative system architecture, Important design considerations when using EOS.

UNIT IV TIME-DRIVEN MULTI-STATE ARCHITECTURE AND HARDWARE 12

Multi-State systems and function sequences: Implementing multi-state (Timed) system - Implementing a Multi-state (Input/Timed) system. Using the Serial Interface: RS232 - The Basic RS-232 Protocol - Asynchronous data transmission and baud rates - Flow control - Software architecture - Using on-chip UART for RS-232 communication - Memory requirements - The serial menu architecture - Examples. Case study: Intruder alarm system.

UNIT V EMBEDDED JAVA 12

Introduction to Embedded Java and J2ME – Smart Card basics – Java card technology overview – Java card objects – Java card applets – working with APDUs – Web Technology for Embedded Systems.

TOTAL 60

REFERENCE BOOKS

1. Steve Oualline, 'Practical C Programming 3rd Edition', O'Reilly Media, Inc, 2006.
2. Stephen Kochan, "Programming in C", 3rd Edition, Sams Publishing, 2009.
3. Michael J Pont, "Embedded C", Pearson Education, 2007.
4. Zhiqun Chen, 'Java Card Technology for Smart Cards: Architecture and Programmer's Guide', Addison-Wesley Professional, 2000.

PCE804	REAL TIME SYSTEMS	L T P C 4 0 0 4
GOAL	To impart knowledge about real time systems that is a narrative technique whereby events are depicted as occurring entirely within the span of and at the same rate as the depiction.	

OBJECTIVES	OUTCOMESs
The course should enable the students to 1. Expose the students to the fundamentals of Real Time systems 2. Learn the fundamentals of Scheduling and features of programming languages 3. Study the data management system for real time 4. Introduce the fundamentals of real time communication 5. Learn the different algorithms and techniques used for real time systems	The students should be able to 1. Have learnt the basics and issues in real time computing 2. Do low level programming for real time systems 3. Design real time databases 4. Have learnt real time communication techniques 5. Have learnt evaluation techniques

UNIT I INTRODUCTION **12**

Introduction – Issues in Real Time Computing – Structure of a Real Time System – Task classes – Performance Measures for Real Time Systems – Estimating Program Run Times – Task Assignment and Scheduling – Classical uniprocessor scheduling algorithms – Uniprocessor scheduling of IRIS tasks – Task assignment – Mode changes and Fault Tolerant
Scheduling.

UNIT II PROGRAMMING LANGUAGES AND TOOLS **12**

Programming Languages and Tools – Desired language characteristics – Data typing – Control structures – Facilitating Hierarchical Decomposition, Packages, Run time (Exception) Error handling – Overloading and Generics – Multitasking – Low level programming – Task Scheduling – Timing Specifications – Programming Environments – Run time support.

UNIT III REAL TIME DATABASES **12**

Real time Databases – Basic Definition, Real time Vs General Purpose Databases, Main Memory Databases, Transaction priorities, Transaction Aborts, Concurrency control issues, Disk Scheduling Algorithms, Two – phase Approach to improve Predictability – Maintaining Serialization Consistency – Databases for Hard Real Time Systems.

UNIT IV COMMUNICATION **12**

Real – Time Communication – Communications media, Network Topologies Protocols, Fault Tolerant Routing. Fault Tolerance Techniques – Fault Types – Fault Detection. Fault Error containment Redundancy – Data Diversity – Reversal Checks – Integrated Failure handling.

UNIT V EVALUATION TECHNIQUES **12**

Reliability Evaluation Techniques – Obtaining parameter values, Reliability models for Hardware Redundancy – Software error models. Clock Synchronization – Clock, A Nonfault – Tolerant

Synchronization Algorithm – Impact of faults – Fault Tolerant Synchronization in Hardware – Fault Tolerant Synchronization in software.

TOTAL 60

REFERENCE BOOKS

1. C.M. Krishna, Kang G. Shin, "Real – Time Systems", McGraw – Hill, 1997.
2. Rajib Mall, "Real-time systems: theory and practice", Pearson Education, 2007
3. Peter D.Lawrence, "Real Time Micro Computer System Design–An Introduction, McGraw Hill, 1988.
4. Stuart Bennett, "Real Time Computer Control – An Introduction", Prentice Hall, 1998.
5. S.T. Allworth and R.N.Zobel, "Introduction to real time software design", Macmillan, 2nd Edition, 1987.
6. R.J.A Buhur, D.L Bailey, "An Introduction to Real – Time Systems", Prentice – Hall International, 1999.
7. Philip.A.Laplante, "Real Time System Design and Analysis", Prentice Hall of India, 3rd Edition, April 2004

PCE805	NANO COMPUTING	L T P C 4 0 0 4
GOAL	To impart knowledge about nano computing which is the latest technology used in all the fields of engineering and technology..	

OBJECTIVES	OUTCOMESs
The course should enable the students to 1. Learn the students to the fundamentals of Nano Computing. 2. Learn the engines of construction. 3. Understand its long life and future. 4. Learn the engines of destruction 5. Know the latest trends in Nano computing	The students should be able to 1. Explain the fundamentals of nano computing 2. Design the engines of construction. 3. Design the engines of destruction. 4. Apply nano computing in real time.

UNIT I ENGINES OF CONSTRUCTION 12

Two styles of technology – Molecular technology today – Existing protein machines – Designing with protein – Second-generation nanotechnology – Universal assemblers – Nailing down conclusions – Nanocomputers – Disassemblers – The world made new – The principles of change – Order from chaos – Evolving molecules – Explaining order – Evolving organisms – Another route back – The rise of the replicators – Evolving technology – The evolution of design – The new replicators – The creatures of the mind – Selecting ideas – Predicting and projecting – Pitfalls of prophecy – Science and natural law – Science vs. technology – The lesson of leonardo – The assembler breakthrough.

UNIT II ENGINES OF ABUNDANCE AND THINKING MACHINES 12

Engines of abundance – Clanking replicators – Molecular replicators – Molecules & skyscrapers – Thinking machines – Machine intelligence – Turing's target – Engines of design – The AI race – Accelerating the technology race – The world beyond earth – The new space program – Space and advanced technology – Abundance – The positive-sum society – Engines of healing – Life – Mind and machines – From drugs to cell repair machines – Cell repair machines – Some cures – Anesthesia plus – From function to structure – From treating disease to establishing health – A disease called "Aging".

UNIT III LONG LIFE IN AN OPEN WORLD AND FUTURE 12

Cell repair machines – Healing and protecting the earth – Long life and population pressure – The effects of anticipation – Progress in life extension. – A door to the future – The requirements for biostasis – Methods of biostasis – Reversing biostasis – Mind – Body and soul – Reactions & arguments – Time – Cost and human action – The limits to growth – The structure of the vacuum – Will physics again be upended – The limits to hardware – Entropy – A limit to energy use – The limits to resources – Malthus – Will someone stop us – Growth within limits – Views of limits.

UNIT IV ENGINES OF DESTRUCTION AND SURVIVAL 12

The threat from the machines – Engines of power –Trustworthy systems – Tactics for the assembler breakthrough – Is success possible – Strategies and survival – Personal restraint – Local suppression – Global suppression agreements – Global suppression by force – Unilateral advance – Balance of power – Cooperative development – A synthesis of strategies – Active shields vs space weapons – Power – Evil – Incompetence and sloth – Finding the facts – A mess of experts – From feuds to due process – An approach – Why not due process – Building due process.

UNIT V THE NETWORK OF KNOWLEDGE AND TIME 12

Magic paper made real – Linking our knowledge – Dangers of hypertext – From desktop to world library – Hypertext and printing press – Worlds enough and time – Nanotechnology and daily life – Other science fiction dreams – Advanced simplicity – Room enough for dreams – Preparations.

Total 60

TEXT BOOK

1. Eric Drexler, "Engines of Creation: The Coming Era of Nanotechnology", Reprint Edition, Anchor, 1987.

REFERENCE BOOKS

1. Mark A Ratner, Daniel Ratner, Mark Ratner, "Nanotechnology: A Gentle Introduction to the Next Big Idea", 1st Edition, Prentice Hall of India, 2002.
2. Eric Drexler, "Nanosystems: Molecular Machinery, Manufacturing, and Computation", 1st Edition, Wiley, 1992.
3. Jack Uldrich and Deb Newberry, "The Next Big Thing Is Really Small: How Nanotechnology Will Change the Future of Your Business",1st Edition, Crown Business, 2003.
4. Douglas Mulhall, "Our Molecular Future: How Nanotechnology, Robotics, Genetics and Artificial Intelligence Will Transform Our World", Prometheus Books, 2002.