

ECE1101 Programing Fundamentals and Problem Solving**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
-	1	2	1		3	2

COURSE AIM

Introducing the basic structured programming and problem solving techniques associated with the C/C++-Language, used to program most nowadays systems. Studying their application to practical problems with emphasis on some practical applications concerning multi- disciplinary fields. The course will cover programming and problem solving techniques, sequence, iterations and multi-disciplinary engineering case-studies, strings ,arrays, functions, and modular programming.

COURSE WEEKLY CONTENTS

- 1 Data type operators and simple functions
- 2 Input/output statements and expressions.
- 3 Arithmetic operations
- 4 Selection structures and switch statements
- 5 Selection structures and switch statements continued
- 6 7th Week Exam
- 7 Iterations and Multi-disciplinary Engineering Case-studies
- 8 Iterations and Multi-disciplinary Engineering Case studies continued.
- 9 Arrays and Application Case-Studies
- 10 Multi-dimensional Arrays and Application Case studies in Engineering Problems.
- 11 12th Week Exam
- 12 String structures and applied applications in Engineering Problems
- 13 Functions and modular programming.
- 14 Functions and modular programming continued
- 15 Data type operators and simple functions

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	←		30	MARKS		→	30
8 to 12	←		20	MARKS		→	20
1 to 15	←		10	MARKS		→	10
16 or 17	40	Final					40

REFERENCES

Textbook Starting Out with C++: From Control Structures through Objects Global Edition, 8th Edition, Tony Gaddis

4.3.2. Computer Engineering (ECE):

ECE2102 Applied Programming

COURSE INFORMATION

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE1101	2	3	2	0	2	3

COURSE AIM

An advanced C/C++-language Programming is provided in this course: arrays(dynamic), searching and sorting algorithms, strings, pointers, recursion, structures, bitwise-operators, as well as text and binary files are covered in details. The course also introduces the basic concepts of classes. Projects are required from students to increase their skills in programming and to increase their ability to work in teams. The course also introduces covers different applications and their link to practical problems with special emphasis on some practical applications concerning different discipline.

COURSE WEEKLY CONTENTS

1	Applications on selection, repetition concepts
2	Applications on arrays and multidimensional arrays
3	Functions & Recursion
4	Functions (Call by reference)
5	Pointers
6	Arrays (Dynamic)
7	7th Week Exam
8	Searching algorithms & complexity analysis
9	Sorting algorithms & complexity analysis
10	Structures
11	Structures with pointers
12	12th Week Exam
13	Files
14	Introduction to Classes
15	Classes and objects
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10 MARKS			→	30
8 to 12	←	15 MARKS		5 MARKS		→	20
14 to 15	←	10 PROJECTS				→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Starting Out with C++: From Control Structures through Objects Global Edition, 8th Edition, Tony Gaddis
Other	C++ How to Program (Early Objects Version), Global Edition (10th Edition), Deitel, Paul/Deitel, Harvey;

ECE2103 Data Structure and Algorithms**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2102	2	4	2	2	0	3

COURSE AIM

- Understand the difference between primitive, derived and user-defined data structure
- Differentiate between static and dynamic memory allocation
- Fully recognize different dynamic data structures with its corresponding algorithms
- Acquire practical programming experience for dynamic structures using C, C++ and Python

COURSE WEEKLY CONTENTS

1	Introduction to the course
2	Memory allocation and introduction to algorithm analysis
3	Linked lists: Single, circular, double and circular doubly linked list
4	Stacks & its applications
5	Queues & its applications
6	Trees: concepts, binary trees & its traversal techniques & searching algorithms
7	7 th week Assessment
8	Graphs: representations, traversal techniques and spanning trees
9	Hashing: Hash function and collision resolution techniques
10	Standard Template Library in C++: Sequence containers (array, vector, list & deque)
11	Standard Template Library in C++: Adaptor containers (stack, queue & priority queue) and associative containers (map & set)
12	12 th week Assessment
13	Basic Data Structures in Python: array, list, dictionary and tuple
14	Advanced Data Structures in Python: linked lists, stacks, queues and binary trees
15	Project submission
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm		10				30
8 to 12	15 EXAM		5				20
13 to 15		5			5		10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Dinesh P. Mehta and Sartaj Sahni "Handbook of Data Structures and Applications", Chapman & Hall/CRC.
Other	• Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, Data Structures and Algorithms in Python, Wiley • Noel Kalicharan, Data Structures in C, CreateSpace Independent Publishing Platform • Reema Thareja, Data Structures using C, Oxford press

ECE2104 Object-oriented Programming**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
, ECE2102	2	4	2	0	2	3

COURSE AIM

In this course, students will develop engineering skills in the design and analysis programming of C# language. The course covers the main features of Object Oriented programming paradigm.

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

- Highlight the main features of object oriented programming and the C# programming language.
- Design and write computer programs for object oriented systems.
- Develop software skills in the design and analysis of C# programming

COURSE WEEKLY CONTENTS

1	An introduction to C# programming language.
2	Input / output statement expressions.
3	Windows application design.
4	Conditional statement and iteration statements.
5	Object and classes.
6	Methods and modular programming.
7	7th week exam.
8	Arrays.
9	Strings object and string methods.
10	Classes and objects (OOP).
11	Inheritance.
12	12th week exam.
13	Polymorphism.
14	Abstract classes and Interfaces.
15	Exceptions and Exception Handling.
16	Final Exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total	
1 to 7	20 Midterm	←	10	MARKS			→	30
To be freely distributed among possible assessments								
8 to 12	←		20	MARKS			→	20
13 to 15	←		10	MARKS			→	10
16 or 17	40	Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100	

REFERENCES

Textbook • Deitel et al, "Visual C# 2010 how to program", Prentice Hall, latest edition..

Other

- *C# in depth: What you need to master C#. John Skeet, Manning Publications Co.*
- *C# in a Nutshell: The Definitive Reference. Joseph Albahari and Eric Johanssen. O'Reilly Publications.*
- *Michael McMillan, "Data structures algorithms and programming style with C#", Cambridge University Press, latest edition.*

ECE2201 Digital Logic Design**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE1101	2	3	2	2	1	3

COURSE AIM

The course introduce the concepts of number systems - binary arithmetic and codes - logic gates - Boolean algebra and logic simplifications - Design and realization of combinational circuits - Functions of combinational circuits logic- Sequential Logic (Flip-Flops, latches, synchronous /Asynchronous counter designs) . Finally, design analysis and realization of counters and shift registers are covered.

COURSE WEEKLY CONTENTS

1	Introduction to digital concepts.
2	Number systems, operations, and codes.
3	Logic gates
4	Boolean algebra and logic simplification
5	(SOP) and (POS) Expressions.
6	Simplification using Karnaugh maps
7	7th Week Exam
8	Functions of combinational logic, adders and sub-tractor design
9	Decoders, encoders and combinational design.
10	MUX, DEMUX and Comparator design.
11	Flip-Flops and related devices.
12	12th Week Exam
13	Asynchronous counter design.
14	Synchronous counter design.
15	Shift registers
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10 MARKS			→	30
8 to 12	20						20
14 to 15	←		10 QUIZZES, AND LAB WORK			→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Thomas L. Floyd, "Digital Fundamentals", Prentice Hall, Eleventh edition.
Other	M. Mano, "Digital Design", Prentice Hall, latest edition. J. P. Hayes, "Introduction to Digital Logic Design", Addison Wesley, latest edition. - David M. Buchla, "Experiments in Digital Fundamentals", Prentice Hall, latest edition. - Texas Instruments Data Sheets, latest version.

ECE2202 Digital Systems Analysis**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2201	2	4	2	1	2	3

COURSE AIM

Students will gain knowledge about methodologies and tools for digital design, optimization, and verification of custom digital systems/hardware. They learn how to design synchronous digital circuits on register transfer level, analyze their timing and implement them in VHDL. The student will be familiar with ASM analysis and design, programmable logic and memory modules. Students will analyze digital circuits and apply the concepts of basic timing issues, including clocking, timing constraints, and propagation delays during the design process

COURSE WEEKLY CONTENTS

1	Digital systems: Views and abstractions in digital hardware systems, formalisms for system description
2	Foundations of synchronous digital design
3	Timing analysis and timing constraints, basic architectural transformations
4	Register-transfer-level (RTL) design
5	Combinatorial and sequential circuits
6	Memory, busses, interfaces
7	Design Examples Digital circuit level, register transfer level
8	7 th Week Exam
9	Programmable devices: PLD, FPGA, PLA, ROM, PAL, CPLD, memory modules.
10	Programmable devices: PLD, FPGA, PLA, ROM, PAL, CPLD, memory modules
11	Introduction to VHDL: overview of VHDL and characteristics, user interface and features.
12	Modeling and Simulation: block diagram development, hierarchical schematic modeling, digital system modeling with VHDL, functional simulation of combinational and sequential circuits, timing models of digital circuit elements
13	12 th Week Exam.
14	Digital systems design: Hierarchical and Modular Design.
15	Algorithmic state machine (ASM) chart.
16	Final exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20	Midterm	←	10	MARKS	→	30
8 to 12	10	←	10	MARKS	→		20
13 to 15	←		10	MARKS	→		10
16 or 17	40	Final					40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook F. P. Processor, D. E. Wiskel, "The art of digital Design and introduction to top-Down Design", Prentice hall, latest edition.

Other

ECE2402 Numerical methods**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
EBA1204 ECE1101	2	3	1		2	2

COURSE AIM

Allow students to master the approximation techniques used in numerical solutions that arise in science and engineering problems. Teach students how numerical methods work, what types and sources of errors to expect and when an application might lead to difficulties, solution of systems of linear equations for direct and indirect methods, Optimization, numerical integration, numerical interpolation, least square error and regression.

COURSE WEEKLY CONTENTS

1	Introduction to Numerical Analysis and course description.
2	Solution of equation: Bisection method, Secant method and False Position.
3	Solution of equation: Successive Approximation and Modified Successive Approximation methods
4	Solution of equation: Newton Raphson method and Berge Vieta method.
5	Error Analysis: Definition of error, Instability, Ill conditioning, Source of errors, Types of errors.
6	Error Analysis: Floating Point Arithmetic & Rounding, Error Propagation, Process Graph.
7	7th week exam.
8	Solution of system of linear equations: (Direct Methods) Gauss Elimination and Gauss Jordan using Gauss Elimination
9	Solution of system of linear equations: Gauss Jordan using Pivots, Gauss Jordan for Integral Matrix and Matrix Inverse
10	Solution of system of linear equations: (Indirect Methods) Jacobi and Gauss Seidel
11	Unconstrained and Constrained Optimization.
12	12th week exam.
13	Numerical Integration: Mid-point, Trapezoidal, Simpson.
14	Numerical Interpolation: Linear, Quadratic and Repeated Linear.
15	Least square error and regression
16	Final exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS	→		30
8 to 12	10 Midterm		10	MARKS		→	20
13 to 15	←		10	MARKS		→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Steven C. Chapra and Raymond P. Canale, "Numerical Methods for Engineers", McGraw Hill, latest edition.
Other	• Faire Burden, "Numerical Analysis", 5/ED. PWS, latest edition. • Earl E. Swartzlander, "Computer Arithmetic-vol 1", IEEE Computer Society Press. • Robert F. Churchhouse, "Numerical methods hand book of applicable mathematics, John Wiley & Sons, latest edition

ECE2401 Discrete Mathematics**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE1101	2	3	1	2	0	2

COURSE AIM

To get the students be able to:

Comprehend mathematical logic, predicates and methods of proof.

Use mathematical induction to prove theorems.

Understand discrete structures like Functions and Relations on Sets

Use techniques of counting and probability theory to solve simple problems

Apply the knowledge of predicate logic and inference rules to build a simple Prolog project

Survey the most important applications of Discrete Mathematics in different Computer Engineering Tracks

COURSE WEEKLY CONTENTS

1. Logical form & logical equivalence.
2. Conditional statement, valid & invalid arguments.
3. Predicates & quantified statements.
4. Number theory and methods of proof.
5. Proof by Mathematical Induction and Inference Rules
6. Functions defined on general sets.
7. 7th week exam.
8. One-to-one, onto, inverse function, composition of functions.
9. Relations on sets.
10. Reflexivity symmetry, transitivity & equivalence relations.
11. Partial order relations and finite state automata.
12. 12th week exam.
13. Counting Techniques (1).
14. Counting Techniques (2).
15. Applications of Discrete Math Presentations and Discussion
16. Final Exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS			30
8 to 12	←		20	MARKS			20
13 to 15	←		10	MARKS			10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook • Kenneth Rosen, "Discrete Mathematic and Application", McGraw Hill, Latest edition.

Other • Sussana Epp, "Discrete Mathematics with Applications", Cengage Learning, Latest edition.

ECE3203 Computer Architecture**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2202	3	5	2	1	2	3

COURSE AIM

To introduce students to the basic concepts of computer architecture and organization through the study of the fundamentals associated with subject matter such as, Basic Computer Organization - Arithmetic Logic Unit - High Speed Arithmetic – The MIPS Architecture – Performance Evaluation Techniques - The Memory Element - Memory Organization - Computer Input/Output Organization – The Hardwired Control Unit - Microprogramming and typical Computer Architectures – CISC and RISC Paradigms.

COURSE WEEKLY CONTENTS

1	Introduction-overview of computer architecture-lecture grading policy.
2	Computer abstraction and technology + term project.
3	The role of performance.
4	MIPS Assembly Language.
5	Instructions for making decisions and procedures calls.
6	Addressing modes.
7	7 th Week Exam
8	The processor: Data path and control.
9	Single-Cycle Data path.
10	Multi-Cycle Data path.
11	Exceptions.
12	12 th Week Exam
13	Pipelining,
14	Hazards and Dependencies in pipelining.
15	Cache memory and Virtual memory.
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS	→		30
8 to 12	10 Midterm		10	MARKS		→	20
13 to 15	←		10	MARKS		→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• David A. Patterson and John I. Hennessy, “Computer Organization and Design: The Hardware/Software Interface”, El Sevier Ltd, latest edition.
Other	• M. Mano, C. Kime, “Computer Design Fundamentals”, Prentice Hall, latest edition. • Michael Flynn, “Computer Architecture” Jones and Bartlett, latest edition.

ECE3204 Microprocessors Systems**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3203	4	6	2	2	1	3

COURSE AIM

Is to have an in-depth knowledge of the architecture and programming of 8-bit and 16-bit Microprocessors. Introducing students to the basic processor components and control unit design. The Intel 8086/8088 Microprocessor is used as case study. The programming and interfacing with memory and various peripheral devices are introduced. The students explore the essentials of memory address decoding, bus timing and I/O interfacing.

COURSE WEEKLY CONTENTS

1	Introduction to Microprocessor and Microcomputers
2	Microprocessor & its architecture
3	Assembly Language Programming – Addressing modes
4	Arithmetic Instructions and Programs – Instruction set
5	Branching and Looping Instructions
6	8086/8088 Hardware Specifications
7	7th week exam.
8	Memory Interface
9	I/O Interfacing
10	Interrupts – Types, Vector tables and Priority Schemes
11	Programmable Peripheral Devices
12	DMA Controllers
13	12 th week Exam
14	Bus Interface
15	Advanced Processors
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10 MARKS			→	30
8 to 12	10 Midterm		10 MARKS			→	20
13 to 15	←		10 MARKS			→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Barry B. Brey, "Intel Microprocessors", Prentice Hall, latest edition.
Other	- Mazidi, Muhammad Ali, "The x86 PC Assembly Language, Design and interfacing", Prentice-Hall, Inc Latest Edition. - Walter A. Triebel, Avtar Singh, "The 8088 and 8086 Microprocessors: Programming, interfacing, hardware", Pearson, latest Edition

ECE3301 Computer Networks**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
EEC2220	3	5	2	2	1	3

COURSE AIM

This course introduces the students to fundamental concepts in data communications and computer networks, including their design, implementation, and simulation. First, the course covers the main principles of data communications which includes the OSI and TCP/IP protocol architectures, data transmission and encoding techniques, multiplexing, and switching techniques. Then, the course will cover in depth network protocols found in application, transport, and network layers. Application layer protocols include HTTP, FTP, SMTP and DNS. Transport layer protocols include UDP, TCP and TCP congestion control. Network layer protocols include forwarding and addressing (IPv4, IPv6). Finally, Software-defined networks is examined. Students are introduced to ns-3 simulation package in addition to Wireshark software. Moreover, student will learn how to implement various network protocols using Java-based Socket Programming and Python-based socket programming.

COURSE WEEKLY CONTENTS

1	Introduction to the course – Protocol Layers and Their Service Models
2	Data Transmission- Encoding Techniques
3	Switching Techniques and Multiplexing
4	What is the Internet? The Network Edge-The Network Core-Delay, Loss, and Throughput in Packet Switched Networks- Networks Under Attack
5	Principles of Network Applications- The Web and HTTP-
6	SMTP: Electronic Mail Protocol in the Internet- DNS: The Internet's Directory Service
7	Transport-Layer Services-Multiplexing and Demultiplexing-Connectionless Transport: UDP
8	7 th week exam
9	Principles of Reliable Data Transfer
10	Connection-Oriented Transport: TCP
11	Principles of Congestion Control- TCP Congestion Control
12	Overview of the Network Layer-What is inside a router?
13	12 th week exam
14	The Internet Protocol: Forwarding and Addressing in the Internet
15	Generalized forwarding and SDN
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS	→		30
8 to 12	←		20	MARKS	→		20
13 to 15	←		10	MARKS	→		10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	James F. Kurose, Keith W. Ross, Computer Networking: A Top-Down Approach, , Pearson, Latest edition
Other	- David J. Wetherall , Andrew S. Tannenbaum, “Computer Networks”, Pearson, Latest edition - William Stallings, Data and Computer Communication, Pearson, Latest edition - Behrouz A Forouzan, Data Communications and Networking, McGraw-Hill Higher Education, Latest edition

ECE3502 Systems Programming**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2104	3	5	1	2	0	2

COURSE AIM

Help students understand the architecture of a hypothetical machine, its assembly language. Getting the student familiar with the structure and design of assemblers, linkers and loaders. Introducing the programming languages development. Explaining the components of compilers

COURSE WEEKLY CONTENTS

1	Introduction to Systems Programming
2	Simplified Instructional Computer (SIC)
3	SIC/XE machine architecture
4	SIC/XE machine dependent features
5	SIC/XE machine independent features (literals, symbols, expression)
6	Machine dependent assembler features
7	7th Week Exam +Revision
8	SIC/XE assembler (Program blocks and control sections)
9	Loader and Linkers (part 1)
10	Loader and Linkers (part 2)
11	Introduction to Compilers
12	12th Week Exam
13	Lexical Analysis
14	Syntax Analysis (part 1)
15	Syntax Analysis (part 2)
16	Final Exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm1	✓				✓	30
8 to 12	20 Midterm2						20
13 to 15		✓		✓	✓	✓	10
16 or 17	40 Final						40
Total	Exams	Assign	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	•Dhotre I and Puntambekar A, System Software, 4th Edition, Technical Publications, 2014
Other	• Aho, A.V., Sethi, R. & Ullman, J.D. (2007). Compilers: principles, techniques and tools. Addison-Wesley (2nd ed.). Mogensen, T.Æ. (2011). Introduction to compiler design. Springer. http://www.diku.dk/~torbenm/Basics . •Leland L. Beck, System Software: An Introduction to Systems Programming, 3rd Edition, Addison Wesley, Longman Inc., 1997.

ECE3503 Operating Systems**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3502	3	6	2	2	1	3

COURSE AIM

The course introduce concepts of managing modern computer resources which is handled by various OS techniques. It also aims at exposing the student to practical OS features and techniques.

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

Discuss main OS concepts.

Solve sample resource management problems

list important OS algorithms

Evaluate OS performance, and be exposed to the Linux OS

COURSE WEEKLY CONTENTS

<i>Week Number 1:</i>	Introduction and Computer System Overview.
<i>Week Number 2:</i>	Computer System Overview (cont.).
<i>Week Number 3:</i>	Operating System Overview.
<i>Week Number 4:</i>	Process Description and Control.
<i>Week Number 5:</i>	Threads.
<i>Week Number 6:</i>	Concurrency: Software Approaches for Mutual Exclusion.
<i>Week Number 7:</i>	7th week exam.
<i>Week Number 8:</i>	Concurrency: Other approaches for mutual exclusion (hardware support and OS support (semaphores)).
<i>Week Number 9:</i>	Concurrency: Deadlock prevention, avoidance, and detection.
<i>Week Number 10:</i>	Memory Management.
<i>Week Number 11:</i>	Virtual Memory: Hardware and Control Structures.
<i>Week Number 12:</i>	12th week exam.
<i>Week Number 13:</i>	Virtual Memory: OS Software.
<i>Week Number 14:</i>	Uniprocessor Scheduling.
<i>Week Number 15:</i>	Disk Scheduling and Revision.
<i>Week Number 16:</i>	Final Exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS	→		30
8 to 12	←		20	MARKS		→	20
13 to 15	←		10	MARKS		→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook • William Stallings, “Operating Systems: Internals and Design Principles”, Prentice Hall, latest edition.

Other

- Harvey M. Deitel, “An introduction to Operating Systems”, Addison Wesley, latest edition
- Andrew S. Tanenbaum and Albert S. Woodhull, Operating Systems Design and Implementation, 3rd Edition, Prentice Hall

ECE3511 Database Systems**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2104	3	6	2	2	1	3

COURSE AIM

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

Understand database design concepts.

Design a database system for a real-world problem.

Develop a database system using a relational DBMS.

Write SQL modules to create and query a database application.

Study Transaction Management and basic Concurrency control and Recovery techniques

Get familiar with current DBMS applications and current models like NOSQL DBMS

COURSE WEEKLY CONTENTS

1. Introductory to database concepts.
2. Relational data model of relational database systems.
3. Relational Integrity rules.
4. Relational algebra.
5. Basic relational analysis and data modeling.
6. Normalization of database tables. (Part 1).
7. 7th week exam.
8. Normalization of database tables. (Part 2).
9. Extended relational analysis and data modeling.
10. Developing Entity / relationship diagram.
11. Developing a database design and Implementation of a real-world problem.
12. 12th week exam.
13. Transaction management and concurrency control.-1
14. Transaction management and concurrency control -2
15. Current Trends in DBMS.
16. Final Exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS			30
8 to 12	←		20	MARKS			20
13 to 15	←		10	MARKS			10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

- | | |
|-----------------|--|
| Textbook | <ul style="list-style-type: none"> Peter Rob and Carlos Coronel, "Database Systems: Design, Implementation, and Management", Course Technology, latest edition. |
| Other | <ul style="list-style-type: none"> Ramez Elmasry & S. Navathe, "Database Management Systems", Pearson Publishers, latest edition. Ramakrishnan. Gehrke, "Database Management systems", Mc-Graw Hill, latest edition. |

ECE3601 Artificial Intelligence**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2104	3	6	2	2	0	3

COURSE AIM

This course will introduce the basic principles in artificial intelligence. It will cover simple representation schemes, problem solving methodology, and search strategies including heuristic search and iterative improvement search. Areas of application such as knowledge representation, expert systems, propositional logic, game playing trees will be explored. Understanding of the basic issues of knowledge representation and blind and heuristic search, as well as an understanding of other topics such as minimax, resolution, etc. that plays an important role in AI programs. Additionally, some natural inspired techniques and genetic algorithms will be introduced. The LISP programming language will also be introduced. The course will enable students to go into industry or research, they will be able to choose the correct AI techniques for the problems which arise.

COURSE WEEKLY CONTENTS

1	Introduction to AI: Definition - History – Goals
2	AI Representation and Search. State Space, problem solving & search strategies.
3	Blind search techniques.
4	Informed (Heuristic) search techniques: Hill Climbing – Best First A* search. Monotonicity – Informedness of a heuristic function and Evaluation of search strategies and complexity
5	Game trees
6	Machine Learning (Supervised and Unsupervised)
7	7th Week Exam +Revision
8	Introduction to artificial neural networks, perceptron networks, and feed forward networks
9	Introduction to generic algorithms and optimization problems.
10	Clustering Techniques
11	Knowledge representation and Propositional logic
12	First Order Logic: Syntax - Semantic - Resolution - Soundness – Completeness.
13	Expert systems and Knowledge-based systems.
14	Continue with expert systems, forward and backward chaining
15	AI Applications & case studies
16	Presentation of projects and Final Exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS			30
8 to 12	←		20	MARKS			20
13 to 15	←		10	MARKS			10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Stuart Russell and Peter Norvig, Artificial Intelligence: A modern Approach, Prentice-Hall, latest edition, available on Kortext Portal.
Other	- Elaine Rich, Kevin Knight, Artificial intelligence, McGrawHill Inc. - Peter Jackson, Introduction to Expert Systems, Addison Wesley, 3rd edition. - Ivan Bratko, Prolog programming for AI, Addison Wesley, 3rd edition

ECE4210 Embedded Systems Design**COURSE INFORMATION**

Prerequisites		Academic Year & Level		Teaching Methods			Credit Hrs.
		Year	Semester	Lecture	Tutorial	Laboratory	
ECE3503	ECE3204	4	8	2	2	1	3

COURSE AIM

The aim of the course is to provide students with a detailed understanding of embedded systems, covers fundamentals of microcontrollers, Interfacing, Memory, and I/O modules. Develop an understanding of the technologies behind Embedded Systems design issues, embedded hardware Building Blocks, embedded processors, memory addressing and interfacing. The course covers Embedded Systems programming, embedded operating systems, design, development, implementation, and testing.

COURSE WEEKLY CONTENTS

1	Introduction to Embedded systems, Microcontrollers, Embedded processors, and Comparison between Embedded systems and conventional computing systems
2	Microcontrollers' architecture, Interfacing, Memory, and I/O modules
3	Embedded Systems Design issues and embedded hardware Building Blocks
4	Embedded Processors
5	Memory Addressing and Interfacing.
6	I/O Interfacing.
7	7th week exam.
8	Embedded Systems Programming
9	Continue with Embedded Systems Programming
10	Embedded Operating Systems
11	Continue with Embedded Operating Systems
12	12th week exam
13	Middleware, application software and embedded systems development.
14	Implementation and Testing
15	Term work Projects
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS	→		30
8 to 12	←		20	MARKS		→	20
13 to 15	←		10	MARKS		→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	- Embedded systems Architecture, Elsevier Inc. Latest edition, ISBN: 0-7506-7792-9, Author: T. Noergaard - Embedded System Design, Springer, 2006, Author: P. Marwedel
Other	- Designing Embedded Hardware, 2005, Author: John Castsoulis, latest Eddtion.. - Computers as Components: Principles of Embedded Computing system Design, latest Edition, Morgan Kaufman Publishers, 2008, Author: W. Wolf, - ARM System Developers' Guide, Authors: A. N. Sloss, D. Symes, and C. Wright

ECE4303 Cybersecurity**COURSE INFORMATION**

Prerequisites		Academic Year & Level		Teaching Methods			Credit Hrs.
		Year	Semester	Lecture	Tutorial	Laboratory	
ECE3301	ECE3503	4	8	2	2	0	3

COURSE AIM

To gain knowledge in the security across different domains such as network, software, operating systems and cloud. The following core competencies and learning outcomes should be achieved by the end of the course: Analyze and resolve security issues in networks and computer systems to secure an IT infrastructure. Design, develop, test and evaluate secure software.

COURSE WEEKLY CONTENTS

1	Introduction to the course – overview of cybersecurity and its related concepts: cyberwarfare, cybercrime. Introduction to security requirements (CIA), breaches consequences
2	Cryptography and Malware- Concept of Encryption, Decryption: Private-key vs Public-key algorithms. Data Integrity: Message Authentication Code(MAC), Hashing, HMAC. Malicious software types: Viruses, Worms, Trojans horses...
3	Local host security: The three layers of security. Defining and securing the outer perimeter and inner perimeter.
4	Local host security: Prevention methods and techniques: OS hardening, Firewalls, Browser security, Intrusion Detection Systems.
5	Network security: Transport level security, Secure Socket Layer (SSL), certificates. HTTPS
6	Network security: Wireless Network Security, Mobile Device Security, IP security
7	7 th week Assessment
8	Network security: Cloud computing security, cloud security as a service, data protection in the cloud
9	Cyber security attacks and vulnerabilities: Identifying, Defending and Preventions
10	Exploiting software vulnerabilities: bad-code practice, code weaknesses and secure coding.
11	Webmail security: Pretty Good Privacy (PGP), S/MIME, DomainKeys Identifies Mail
12	12 th week Assessment
13	Social Engineering cybersecurity
14	Ethical Hacking
15	Big Data security: Protecting Data Moving Through the Internet
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	15 Midterm	←	15 MARKS			→	30
8 to 12	←		20 MARKS			→	20
13 to 15	←		10 MARKS			→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Brooks, C. J., Grow, C., Craig, P., & Short, D. (2018). Cybersecurity essentials. John Wiley & Sons.
Other	Stallings, William. Cryptography and network security, 6/E. Pearson Education, 2006. Singer, Peter W., and Allan Friedman. Cybersecurity: What everyone needs to know. oup usa, 2014.

ECE4317 Advanced Networks**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3301	4	7	2	1	1	3

COURSE AIM

This course covers a set of advanced topics in computer networks. The aim of this course is to get students familiar with principles, architectures and protocols used in Switched Local Area Networks, Wireless, Mobile ad hoc networks, and multimedia networks. Topics include: Data Plane, Generalized forwarding and SDN, Control Plane, Routing Algorithms, LAN Multiple Access Protocols, Link Layer Error Detection and Correction Techniques, Link Layer Addressing and switches- ARP- CDMA- Wi-Fi: architecture, CSMA/CA-Frame Format- Mobility Management: addressing and Routing-Mobile IP-Multimedia Networking Applications, VoIP-RTP-SIP-QoS- Diffserv-IoT. Students are required to build IoT systems as a group project.

COURSE WEEKLY CONTENTS

1	Introduction to the course
2	Network Layer: Data Plane, Generalized forwarding and SDN
3	Network Layer: Control Plane, Routing Algorithms
4	Link Layer: Introduction and Services, Error Detection and Correction Techniques.
5	Multiple Access Links and Protocols
6	Wireless Links and Network Characteristics
7	WiFi: 802.11 Wireless LANs
8	7 th week Assessment
9	Mobility Management: Principles- Mobile IP
10	Multimedia Networking Applications- Streaming Stored Video
11	Voice-over-IP
12	Protocols for Real-Time Conversational Applications
13	12 th week Assessment
14	Network Support for Multimedia
15	Introduction to Internet of Things (IoT)
16	IoT Case Studies
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10 MARKS			→	30
8 to 12	←		20 MARKS			→	20
13 to 15	←		10 MARKS			→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	James F. Kurose, Keith W. Ross, Computer Networking: A Top-Down Approach, latest edition
Other	Ala Al-Fuqaha et al., "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications, IEEE Communication Surveys and tutorials, vol.17, No.4, 2015

ECE4504 Computing Systems**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3503	4	8	2	2	0	3

COURSE AIM

To introduce concepts of high performance computing, advanced computing architectures and multiprocessing requirements. The impact of VLSI on modern computing architectures is emphasized with applications on different architectures. To get the students familiar with ILP, RISC architecture, Memory hierarchy, Pipelining, Vector processing, Array processing, Massively parallel processors, Multiprocessor architecture, Data flow computers. Learn Different Parallel computing models, Shared and Distributed Memory Systems, Analyse the use of virtual and cache memory and evaluate their effects on computer systems.

COURSE WEEKLY CONTENTS

1	Introduction to high performance computing.
2	Parallel Computer Models.
3	Instruction Level Parallelism.
4	Memory Hierarchy.
5	Memory and I/O Subsystem.
6	Virtual Memory.
7	7th week exam.
8	Introduction to Pipeline Design.
9	Superscalar / Super pipeline.
10	Multiprocessor architecture.
11	System Interconnect.
12	12th week exam.
13	Vector Processing.
14	SIMD Organization.
15	New Architecture trends.
16	Final exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS		→	30
To be freely distributed among possible assessments							
8 to 12	10 Midterm		10	MARKS		→	20
13 to 15	←		10	MARKS		→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Michel Dubois, Murali Annavaram, Per Stenström " Parallel Computer Organization and Design ", Cambridge University Press, latest edition.
Other	- John P.Hayes, "Computer Architecture and Organization", McGraw Hill, latest edition. K. Hwang, "Advanced Computer Architecture", McGraw Hill, latest edition. - A.John Anderson, "Multiple processing, A system overview", Prentice Hall, latest edition. - Edward Ritman, "Exploring Parallel Processing", Windcrest, latest edition.

ECE4515 Computing Algorithms**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2104	4	7	1	2	0	2

COURSE AIM

Analyze the asymptotic performance of algorithms. Apply important algorithmic design paradigms and methods of analysis to solve problems. Measure the performance of algorithms and Proof correctness of algorithms. Introduction to parallel Algorithms.

COURSE WEEKLY CONTENTS

1	Introduction to the course objectives and policies.
2	Analysis of Algorithms, Asymptotic Notations.
3	Asymptotic Notations (part 2).
4	Divide and Conquer and Quiz #1.
5	Sorting: Heapsort, Priority Queues and Linear-time Sorting.
6	Greedy Methods.
7	7th week exam.
8	Dynamic Programming Part 1.
9	Dynamic Programming Part 2.
10	Minimum Spanning Trees, Network flow algorithms.
11	Graph representation.
12	Computational Geometry, closest pair problem.
13	Parallel Algorithms Part 1.
14	Parallel Algorithms Part 2.
15	Projects and Assignments discussion and presentations.
16	Final Exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS	→		30
8 to 12	←		20	MARKS	→		20
13 to 15	←		10	MARKS	→		10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, “Introduction to Algorithms”, MIT Press, latest edition.
Other	• Steven S. Skiena, “The Algorithm Design Manual”, Springer-Verlag, latest edition.

ECE4711 Computer Graphics**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2104	4	7	2	2	1	3

COURSE AIM

This course introduces students to the theory and practice of Interactive Computer Graphics. Its principal aim is to teach the fundamental principles of two- and three-dimensional Interactive Computer Graphics. OpenGL is used as the platform for practical C programming exercises, and as an example of a system which incorporates many of the fundamental ideas and algorithms of modern computer graphics.

COURSE WEEKLY CONTENTS

1	Introduction, history and survey of computer graphics applications
2	Overview of graphics systems: raster and random scan display
3	Color models
4	Bresenham's line drawing algorithm
5	Mid-point circle/ ellipse drawing algorithms
6	Drawing free curves: Bezier and spline technologies
7	7th week Exam+ Revision
8	2D transformations
9	2D transformations (cont.)
10	Viewing transformation
11	Line and Polygon clipping algorithm
12	12th week Exam+Revision
13	Filling Algorithms
14	Texture Mapping
15	Revision
16	Presentation of projects and Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS	→		30
8 to 12	←		20	MARKS	→		20
13 to 15	←		10	MARKS	→		10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• D. Hearn and M.P. Baker, Computer Graphics Open GI Version, latest edition, Prentice Hall.
Other	• Francis S. Hill, Jr. Computer Graphics Using OpenGL, latest edition, Prentice Hall. • Foley J., Van Dam, A., Feiner, S., Hughes, J., C Edition, Interactive Computer Graphics: Principles and Practice, latest edition, Addison – Wesley.

4.3.3. Electronics and Communication Engineering (EEC):**EEC2220 Introduction to Communication Systems****COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
EBA1204	2	3	2	2	1	5

COURSE AIM**COURSE WEEKLY CONTENTS**

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	←		30	MARKS		→	30
8 to 12	←		20	MARKS		→	20
1 to 15	←		10	MARKS		→	10
16 or 17	40	Final					40

REFERENCES

Textbook

EEC2320 Electronics I**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
EEE2304	2	4	1	2	0	2

COURSE AIM

Introducing different electronic devices used in constructing modern electronic circuits. Studying semiconductor materials, p-n junction Diodes, Bipolar Junction Transistor (BJT) and Metal Oxide Field Effect Transistor (MOSFET). Also, studying their performance with special emphasis on some practical applications.

COURSE WEEKLY CONTENTS

- 1 Revision on Electric Circuits Theorems
- 2 Thevenin Equivalent Theorem
- 3 Types of solids: conductor, insulator, semiconductor. Covalent bond – Semiconductor types – Doping of semiconductors (P and N).
- 4 Mobility and conductivity in semiconductors (intrinsic and extrinsic) – Hole and electron concentration - Diffusion and drift currents
- 5 Built-in voltage in a p-n junction – Depletion layer in a p-n junction. Forward and reverse bias.
- 6 PN junction as a circuit element, I-V characteristics, and different models.
- 7 Half wave and full wave rectifier - Smoothing circuits - Clipping circuits - Clamping circuits.
- 8 Special diodes: Zener diodes - Light emitting diodes (LEDs) – Photodiodes - Varactor diodes - Solar cells.
- 9 Bipolar Junction Transistor (BJT): construction – types – symbol - energy band diagram– operation - BJT: dc solution and biasing circuits.
- 10 BJT Transistor as a switch- Regions of operation: Cut-off, saturation, reverse- and forward active regions.
- 11 BJT: I-V Characteristics of BJT - BJT: Small signal analysis – ac equivalent circuit.
- 12 Metal oxide semiconductor Field Effect Transistor (MOSFET): construction – symbol – operation.
- 13 I-V Characteristics of MOSFET, Enhancement and depletion modes
- 14 MOSFET as a switch; Complementary MOSFET (CMOS): symbol - operation - Basic Logic gates using CMOS (Inverter, NAND, NOR, AND, OR gates)
- 15 Revision
- 16 Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	15	10		0			25
8 to 15	15	10		0			25
10 to 15		0		10			10
16 or 17	40		0				40
Total	70	20		10			100

REFERENCES

Textbook • Boylestad, Robert and Nashelsky, Louis “**Electronic Devices and Circuit Theory**”. Pearson, 2013.

EEC3321 Electronics II**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
EEC2320	3	5	1	2	0	2

COURSE AIM

The student should be able to analyze MOSFET amplifiers, operational amplifiers, filters and oscillators. Familiarize students with the use of MOSFET in digital circuit design either combination or sequential circuits. Arithmetic Logic Unit (ALU) design

COURSE WEEKLY CONTENTS

1	Revision, MOSFET Transistor theory, course overview
2	MOSFET amplifiers. Voltage gain, current gain, input, and output impedance.
3	Frequency response of MOSFET amplifiers.
4	Sinusoidal oscillators.
5	Square wave oscillators.
6	Differential Amplifier
7	Differential Amplifier applications
8	Operational amplifiers, specifications. Analysis of basic Op-Amp circuits
9	Op-Amp applications – Filter circuits
10	MOSFET as a switch, Analysis, and design of digital CMOS logic gates
11	Transistor level Design of compound logic gates. Optimize the design for delay and area.
12	Analysis and design of clocked storage elements (flip-flops, latches)
13	Analysis and design of clocked storage elements (memory cells)
14	Circuit topologies for arithmetic and logical functional units
15	Revision
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	10	10			0		20
8 to 15	15	5			0		20
10 to 15		0			20		20
16 or 17	40			0			40
Total	65	15			20		100

REFERENCES

Textbook	• Floyd, Thomas and Buchla, David “ Electronics Fundamentals. Circuits, Devices, and Applications ”, Pearson, 2014
----------	---

4.3.4. Electrical Engineering (EEE):

EEE2304 Electrical Circuits I

COURSE INFORMATION

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
EBA1204, EBA1103	2	3	1	2		2

COURSE AIM

- Studying basic theories of circuit analysis with direct current circuits.
- Discussing the alternating current concepts.
- Taking an overview on magnetic circuits.

COURSE WEEKLY CONTENTS

- 1 Introduction to Basic Circuit: Resistance, Voltage, Current
- 2 Ohm's Law
- 3 1st & 2nd kirchoff's laws, Dependent Sources
- 4 Series Parallel Circuits
- 5 Nodal analysis.
- 6 Mesh analysis.
- 7 Source Transformation + Midterm Exam
- 8 Superposition
- 9 Thevenin Equivalent Circuit
- 10 Norton Equivalent Circuit, Maximum Power Transfer
- 11 Alternating current fundamentals and AC generation
- 12 Root Mean Square and Average Value Calculations + 12th Exam
- 13 Relation between current and voltage in resistors, inductors and capacitors
- 14 Introduction of Magnetic Circuits
- 15 Analysis of Magnetic Circuits

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm			10 MARKS			30
8 to 12			To be freely distributed among possible assessments				20
13 to 15				10 MARKS			10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	R.L. Boylestad, "Introductory Circuit Analysis", Merrill, London, 1994
Other	J. Nilson & S.Riedel, "Electrical circuits", Prentice Hall, latest edition

EEE2305 Electrical Circuits II**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
EEE2304	2	4	1	2	0	2

COURSE AIM

- Studying basic theories of circuit analysis with direct current circuits.
- Discussing the alternating current concepts.
- Taking an overview on magnetic circuits

COURSE WEEKLY CONTENTS

- 1 Sinusoidal response of series RLC circuits.
- 2 Sinusoidal Steady State Analysis for Series and Parallel Circuits , Resonance
- 3 Star delta transformation
- 4 Source Transformation for AC circuits
- 5 Thevenin and Norton Equivalent Circuit for AC circuits
- 6 Node Voltage Analysis, Mesh Current Analysis for AC circuits
- 7 Complex Power Calculations + Midterm Exam
- 8 Maximum Power Transfer
- 9 Three Phase Systems and Balanced Y-Y Circuit.
- 10 Y- Δ , Δ -Y, Δ - Δ Three Phase systems
- 11 Power Calculation in three Phase System + 12th Exam
- 12 Unbalanced Three Phase Circuits
- 13 The natural response of RL and RC Circuits
- 14 Step response of RL and RC Circuits
- 15 General Solution for RL and RC Circuits

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	□	To be freely distributed among possible assessments			□	30
8 to 12	□			20 MARKS		□	20
13 to 15	□			10 MARKS		□	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	R.L. Boylestad, "Introductory Circuit Analysis", Merrill, London, 1994
Other	J. Nilson & S.Riedel, "Electrical circuits", Prentice Hall, latest edition

EEE3208 Electrical Power and Machines**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Lab.	
EEE2305	3	5	1	2	0	2

COURSE AIM

- To study the basic concepts and applications of DC and AC electric machines
- To investigate operation and control of small scale motors
- To study the basic concepts and applications of transformers
- To investigate the different stages of power system generation and distribution

COURSE WEEKLY CONTENTS

1	DC Motor (construction, theory of operation, and equivalent circuit)
2	DC motor power flow diagram, torque speed characteristics, speed control and typical fields of applications
3	DC servo motor concept, control circuit and application to robotics
4	DC Generator (theory of operation, and equivalent circuit and typical field of application in small scale wind turbine)
4	Stepper motor construction, theory of operation, types and applications
5	Stepper motor drive circuit and operation control for CNC machines
6	Transformer (construction, theory of operation & application example in mobile charger)
7	Concept of rotating field in AC machines. + 7 th Week Exam
8	Three-phase induction motor (construction, theory of operation, and applications)
9	Induction motor power flow diagram and torque speed characteristics
10	Synchronous generator (construction, theory of operation, and applications)
11	Automated parallel operation of synchronous generators
12	Electric power system components and stages + 12 th Week Exam
13	Protective devices and distribution of electricity in buildings.
14	System protection and automatic power factor correction
15	Application case study :utilization of electric machines within electric vehicles

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	□		10 MARKS		□	30
8 to 12	□			20 MARKS		□	20
13 to 15	□			10 MARKS		□	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Stephan J. Chapman "Electric Machinery Fundamentals" Mcgraw-Hill, 5 th Edition, 2011.
Other	C. Hubert, 'Electric Machines: Theory, Operating Applications, and Control', Maxwell Macmilla, .2nd Edition 2001,.

EEE3108 Applied Control Systems**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
EEE3208	3	6	1	2	0	2

COURSE AIM

- To study controller units, their type analysis and tuning
- Modeling and analysis of simple physical system are investigated
- Stability concept and time domain analysis using time response
- State space method of design for continuous time systems

COURSE WEEKLY CONTENTS

- 1 System Modelling (Electrical Systems)
- 2 System Modelling (Electromechanical Systems)
- 3 Block diagram models using MATLAB
- 4 System Analysis and Classifications first order system
- 5 Second order system
- 6 Steady state error
- 7 Stability concept Routh- Hurwitz stability criterion
- 8 Root locus techniques
- 9 Design Lead Controller
- 10 Design Lag Controller
- 11 Design of PI
- 12 Design PD Controller
- 13 Design of PID Controller
- 14 Bode Plot, phase and gain margin
- 15 Continue Bode Plot, phase and gain margin

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign	Quizzes	Reports	Present	Lab.	Total	
1 to 7	20 Midterm	☐	10 MARKS To be freely distributed among possible assessments				☐	30
8 to 12	☐	20 MARKS				☐	20	
13 to 15	☐	10 MARKS				☐	10	
16 or 17	40 Final						40	
Total	Exams	Assign	Quizzes	Reports	Present	Lab.	100	

REFERENCES

Textbook	Ogata , "Modern control Engineering", Prentice – Hall , 1985
Other	Nagrath & Gopal, "Control System Engineering", John Wiley & Son, NY 1982

4.4. Specific Specialization Requirements:

4.4.1. Concentration 1: Embedded Systems

ECE3612 Tiny AI Models

COURSE INFORMATION

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2104	3	6	1	0	2	2

COURSE AIM

This course introduces students to the principles, techniques, and applications of Tiny AI, focusing on designing, training, and deploying machine learning models that are optimized for resource-constrained environments. Students will gain practical experience with model compression, quantization, and efficient neural network architectures, preparing them to implement AI in devices with limited computational power and storage.

COURSE WEEKLY CONTENTS

1	Introduction to Tiny AI
2	Basics of Machine Learning for Tiny AI
3	Data Collection and Processing for Tiny Devices
4	Model Optimization and Compression
5	Quantization and Pruning
6	Efficient Neural Network Architectures
7	7th Week Exam
8	Hardware for Tiny AI (practical examples like Raspberry Pi, Arduino, etc)
9	Deployment Strategies for Tiny AI
10	Power Management for Tiny AI
11	Real-Time Inference on Tiny Devices.
12	12th Week Exam
13	Security Considerations in Tiny AI
14	Ethics and Sustainability in Tiny AI.
15	Project Submission
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS		→	30
8 to 12	←		20	MARKS		→	20
13 to 15	←		10	MARKS		→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Warden, P., & Situnayake, D. (2020). TinyML: Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers. O'Reilly Media.
Other	Ng, A. (2018). Machine Learning Yearning. Self-published.

ECE4205 Data Acquisition Systems**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
EEE2305 EEC2320	3	5	2	2		3

COURSE AIM

To introduce student to the basic techniques of automatic sensing and measurements in the non-digital world. Train students on the principles and how to acquire skills in dealing with sensing different physical phenomena; mechanical, electric, electronic, and optical. Getting the students familiar with industry related data acquisition tools like PLC and embedded systems. Introducing the students to tools that would allow them to emulate the working of PLC

COURSE WEEKLY CONTENTS

1	Introduction
2	Introduction to Control systems & data acquisition channel.
3	Sensors, transducers and actuators
4	Signal conditioning: operational amplifiers & applications
5	Signal conditioning: operational amplifiers circuit analysis and design
6	Introduction to Data conversion, Sampling and Hold, ADC
7	7th week exam
8	Data Conversion Circuits using ADC
9	Data Conversion Circuits using DAC
10	Introduction to PLC
11	PLCs – Ladder Logic (Part 1)
12	12th week Assessment
13	PLCs – Ladder Logic (Part 2)
14	SCADA systems
15	Recent Advances in PLCs
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm1	✓	✓			✓	30
8 to 12	15 Midterm2		✓				20
13 to 15		✓		✓	✓	✓	10
16 or 17	40 Final						40
Total	Exams	Assign	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	•Floyd, Thomas L, "Fundamentals of Analog Circuits ", Pearson Education 2ED
Other	•Kamel, Khaled, and Eman Kamel. Programmable logic controllers: Industrial control. New York: McGraw-Hill Education, 2013 •Frank, D. Petruzella. Programmable logic controllers. Tata McGraw Hill Education Private Limited, 2010. •Kilian, Christopher T. Modern control technology: components and systems. Delmar/Thomson Learning, 2006.

ECE4209 Embedded Software Testing**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3204 ECE3503	4	8	1	0	2	2

COURSE AIM

This course is designed to introduce students to the techniques, tools, and challenges of testing embedded software. It covers the fundamentals of testing methodologies, practical applications, and strategies specific to the embedded systems domain. Students will learn to implement rigorous testing processes to ensure the reliability and performance of embedded software across various platforms.

COURSE WEEKLY CONTENTS

1	Introduction to Embedded Systems and Software Testing
2	Basics of Embedded Systems Architecture
3	Introduction to Software Testing
4	Unit Testing in Embedded Systems
5	Integration Testing
6	System Testing
7	7th Week Exam
8	Real-Time Operating System (RTOS) Testing
9	Performance Testing
10	Stress and Load Testing
11	Usability Testing for Embedded Systems.
12	12th Week Exam
13	Security Testing in Embedded Systems
14	Automation in Embedded Software Testing
15	Advanced Topics and Emerging Technologies
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS			30
8 to 12	←		20	MARKS			20
13 to 15	←		10	MARKS			10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Grenning, J. (2011). Test Driven Development for Embedded C. Pragmatic Bookshelf.
Other	Noergaard, T. (2012). Embedded Systems Architecture. Elsevier.

ECE4307 Cloud Computing**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE4317	4	8	1	0	2	2

COURSE AIM

To get the students familiar with fundamental concepts related to cloud computing technologies. Also, to get familiar with different cloud service and deployment models, different architectural styles and design patterns for cloud computing, the classification of virtualization techniques. The concept of HPC and how it could be used in cloud computing. Understand the use of container service to deploy your application in the cloud and the use of DevOps in configuration management

COURSE WEEKLY CONTENTS

1	Introduction to the course – The enabling technologies of cloud computing, especially virtualization and its different types.
2	Introduction to Cloud Computing (Part 1): history, definition, attributes, advantages, limitation, practical examples.
3	Introduction to Cloud Computing (Part 2): reference model, service models, deployment models, Applications/Data, and lifecycle of a cloud computing application using Amazon Web Services.
4	How to design your Cloud application (Part 1): differences between monolithic and microservices architectures), different architecture styles, and design principles.
5	How to design your Cloud application (Part 2): different pillars of software quality and how to measure them, and diverse types of cloud design patterns (using Microsoft and Amazon Web services as examples).
6	Cloud Infrastructure (Part 1): In-depth of different compute, datastore, and network technologies.
7	7 th week Assessment
8	Cloud Infrastructure (Part 2): In-depth of different compute, datastore, and network technologies.
9	Introduction to Containerization (Part 1): the usage and implementation of containers using dockers.
10	Introduction to Containerization (Part 2): Benefits of Containerization, container orchestration using Kubernetes, and the use of containers in Amazon.
11	Introduction to HPC and cluster computing: the concept, the challenges, design constraints, uses and benefits
12	12 th week Assessment.
13	Configuration Migration and Request for Proposals, Automation, and DevOps
14	Different eras in Cloud Computing: IoT, Fog Computing and Blockchain.
15	Seminars
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	15 Midterm	←	15 MARKS			→	30
			To be freely distributed among possible assessments				
8 to 12	15 Midterm	←	5 MARKS			→	20
			To be freely distributed among possible assessments				
13 to 15	←		10 MARKS			→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Arshdeep Bahga and Vijay Madisetti, “Cloud Computing Solutions Architect: A Hands-On Approach: A Competency-based Textbook for Universities and a Guide for AWS Cloud Certification and Beyond”.
Other	• Rajkumar Buyya, Christian Vecchiola, S. Thaman Selvi, “Mastering Cloud Computing: Foundations and Application Programming”, Morgan Kaufmann, Latest • “Cloud application Architecture Guide”, Microsoft Azure, Microsoft 2021 • Sean Keery, Clive Harber, and Marcus Young, “Implementing Cloud Design Patterns for AWS: Solutions and design ideas for solving system design problems”, 2nd Edition.

ECE4513 Real-time Systems**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3503	4	7	2	2	1	3

COURSE AIM

This course introduces the fundamental concepts, design principles, and applications of real-time systems. Students will learn about the development of systems that respond to events within strict time constraints, focusing on both soft and hard real-time systems. The curriculum covers theoretical concepts and practical implementation, preparing students to design, analyse, and implement real-time systems across various industries.

COURSE WEEKLY CONTENTS

1	Introduction to Real-Time Systems
2	Real-Time System Characteristics
3	Real-Time Operating Systems (RTOS)
4	Task Scheduling in RTOS
5	Static Priority Realtime Scheduling Algorithms
6	Dynamic Priority Realtime Scheduling Algorithms
7	7th Week Exam
8	Soft vs Hard Realtime Systems
9	Algorithms for soft Realtime Systems
10	Concurrency in Real-Time Systems
11	Real-Time Communication.
12	12th Week Exam
13	Dependability and Fault Tolerance
14	Real-Time System Testing and Validation
15	Real-Time Applications in Industry and Future Trends
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total	
1 to 7	20 Midterm	←	10	MARKS			→	30
To be freely distributed among possible assessments								
8 to 12	←		20	MARKS			→	20
13 to 15	←		10	MARKS			→	10
16 or 17	40 Final							40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.		100

REFERENCES

Textbook	Liu, J. W. S. (2000). Real-Time Systems. Prentice Hall.
Other	Laplante, P. A., & Ovaska, S. J. (2012). Real-Time Systems Design and Analysis: Tools for the Practitioner. Wiley-IEEE Press..

ECE4514 Embedded and Realtime Systems (RTOS)**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3503, ECE4513	4	8	2	2	1	3

COURSE AIM

This course explores the fundamentals, design principles, and practical applications of embedded and Realtime operating systems (RTOS). Students will gain hands-on experience with RTOS architecture, system programming, and resource management in constrained environments typical of embedded systems.

COURSE WEEKLY CONTENTS

1	Introduction to Embedded Systems Review
2	Basics of Operating Systems Review
3	Real-Time Operating Systems Review
4	Design and Architecture of RTOS
5	Inter-task Communication and Synchronization
6	Memory Management in Embedded Systems
7	7th Week Exam
8	Embedded File Systems and Storage
9	Device Drivers and Hardware Interfaces
10	Interrupts and Interrupt Handling
11	Power Management and Energy Efficiency.
12	12th Week Exam
13	Safety, Reliability, and Testing of Embedded Systems
14	Real-Time Operating Systems in IoT
15	Advanced Topics and Emerging Trends
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10 MARKS			→	30
			To be freely distributed among possible assessments				
8 to 12	←		20 MARKS			→	20
13 to 15	←		10 MARKS			→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Noergaard, T. (2012). Embedded Systems Architecture. Elsevier
Other	Valvano, J. (2012). Real-Time Operating Systems for ARM Cortex-M Microcontrollers. CreateSpace Independent Publishing Platform

EEC4509 Applied Digital Signal Processing**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3204 & EBA2202	4	7	2	2	1	3

COURSE AIM

Introducing the different techniques for designing and implementing digital signal processing systems, Discrete – time transforms. The student should be familiar with Discrete time signal and systems, the DFT, the DCT and the FFT algorithms. The discrete time transfer function, Design procedures of digital filters, realization topologies, Design of FIR and IIR filters, MATLAB usage in designing digital filters and some practical applications of DSP; speech and image processing

COURSE WEEKLY CONTENTS

- 1 Introduction to DSP, Real time DSP system, Sampling and S/H Circuits
- 2 Discrete Time signals and systems, system properties
- 3 The Z-transform and its application in signal processing
- 4 Z-Transform Cont'd
- 5 Fourier representation of signals, Transform analysis of LTI systems
- 6 The DFT, the DCT and image compression
- 7 FFT algorithm and its inverse (IFFT).
- 8 FFT and audio analysis + 7th week exam
- 9 Discrete-time systems Realizations
- 10 Introduction to Digital Filters; Types of digital filters, FIR vs IIR, Design o FIR digital filters- Linear Phase FIR
- 11 Design o FIR digital filters using windowing Method
- 12 FIR filters and noise reduction + 12th week exam
- 13 Design of IIR filters using the Bilinear z-transform, IIR filter for biomedical signals.
- 14 Discrete Transforms Applications; Speech Processing
- 15 Discrete Transforms Applications; Image ProcessingFinal Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Project	Total
1 to 7	10		10			0	20
8 to 15	15		5			0	20
10 to 15	0		0			20	20
16 or 17	40		0			0	40
Total	65		15			20	100

REFERENCES

Textbook • Hamdy, Nader, “Applied Signal Processing”, CRC Press, 2008

4.4.2. Concentration 2: Autonomous Vehicles**ECE3308 Cyber-Physical Systems Design and Analysis****COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2202 ECE2104	4	7 or 8	2	2	0	3

COURSE AIM

This course introduces the tools and models that will allow students to develop high confidence in the resulting system's proper operation prior to any operational test. Included are tools for model-based systems engineering, and cyber-physical system verification and validation currently in use by the CPS industry. Numerous examples will be considered, from aerospace, automotive, medical devices, etc. The frequent presence of human operators is also acknowledged and discussed in-depth. Various verification and validation formal methods are described and applied to simple examples.

COURSE WEEKLY CONTENTS

- 1 Introduction to CPS: definitions, trends, Areas of interest, Fundamental approach, examples, Genesis, Modeling, Design, Verification and Validation, Assembly and Deployment
- 2 Review: Calculus, Differential equations, Markov models, Linear systems
- 3 Models: Nature and Computation Myths: Airborne Collision avoidance examples, From Continuous to discrete dynamics, Examples: Water tank, spring oscillator, Modeling trade-offs: Faithfulness, manageability, level-of-detail, accuracy
- 4 Low-level CPS design: Canonical Example: Stopping a car, Feedback, Reduced-gravity Drone, Linear Control, Controllability: Train example
- 5 Mid-level CPS design requirements: Trajectory Planning and examples, Aviation example, Typical requirements
- 6 Mid-level CPS design: Guidance techniques, Classical optimization and examples, Dynamic Programs, Automotive example, Rapidly-exploring Random Trees
- 7 7th week Assessment
- 8 High-level CPS design: High-level decision-making: Linear Temporal Logic (LTL), Self-driving car example, LTL and trajectory planning, Reduced-gravity drone example
- 9 Processors and Sensors: Sensors and CPS – trends, Sensors, CPS, and IoT, Actuators and servos, Embedded CPS architectures, Communications, Processors
- 10 Systems Engineering, general approach: Vee Design cycle and activities, Industry standards and advisory documents, Tools and frameworks: Software-centric viewpoint, Model-based engineering pitfalls, AADL virtual integration – cost savings
- 11 Architecture Analysis and Design Language (AADL): General Principles and positioning, Why modeling in design?, Models, Processes, and tools, AADL introduction, AADL Components: software, hardware, AADL properties
- 12 12th week Assessment
- 13 AADL Crazyflie Case Study: Modeling, Implementing a UAV control logic, AADL functional chain, Flow analysis, Latency / real-time scheduling, AADL, middleware, code generation, Error modeling, and analysis
- 14 Formal methods: Concerns of formal methods, Concerns about formal methods, Abstractions, Abstract interpretation, Model Checking, Hunting for invariants
- 15 Project Submission
- 16 Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	☐	10 MARKS			☐	30
8 to 12	☐	20 MARKS			☐		20
13 to 15	☐	10 MARKS			☐		10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook Pedro H. J. Nardelli, Cyber-physical Systems: Theory, Methodology, and Applications, Wiley-IEEE Press; 1st edition 2022

Other

ECE3408 Sensor Data Fusion & Processing Techniques**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
EBA3206 ECE2104	3	6	2		2	3

COURSE AIM

The course introduces students to the techniques of sensor data fusion and processing for autonomous systems.

COURSE WEEKLY CONTENTS

- 1 Introduction to Sensor Data Fusion
- 2 Types of Sensors and their Characteristics
- 3 Data Preprocessing Techniques
- 4 Kalman Filter for Data Fusion
- 5 Extended Kalman Filter
- 5 Unscented Kalman Filter
- 7 7th-week exam
- 8 Particle Filter for Data Fusion
- 9 Multi-sensor Fusion Architectures
- 10 Data Association Techniques
- 11 Outlier Detection and Robust Data Fusion
- 12 12th-week exam
- 13 Real-time Data Fusion Systems
- 14 Case Studies in Sensor Data Fusion
- 15 Future Trends in Sensor Data Fusion
- 16 Final exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams		Assign.	Quizzes	Reports	Present.	Lab.	Total	
1 to 7	10	Midterm	←	15	MARKS			→	25
To be freely distributed among possible assessments									
8 to 12	10	Midterm		20	MARKS			→	30
13 to 15	←			10	MARKS			→	10
16 or 17	35	Final							35
Total	Exams		Assign.	Quizzes	Reports	Present.	Lab.	100	

REFERENCES

Textbook	• Daniel Simon, "Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches," Wiley.
Other	• Rudolph van der Merwe and Eric A. Wan, "The Unscented Kalman Filter for Nonlinear Estimation," IEEE Proceedings.

ECE4110 Model-Based Software Engineering**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2104	4	8	1	2		2

COURSE AIM

This course introduces students to the principles of model-based software engineering and its applications in autonomous systems.

COURSE WEEKLY CONTENTS

- 1 Introduction to Model-Based Software Engineering
- 2 UML and SysML for System Modeling
- 3 Software Development Processes
- 4 Requirements Engineering
- 5 System Design and Architecture
- 6 Model Transformation and Code Generation
- 7 7th-week exam
- 8 Verification and Validation
- 9 Model-Based Testing
- 10 Software Product Lines
- 11 Tools and Frameworks for Model-Based Engineering
- 12 12th-week exam
- 13 Case Studies in Model-Based Engineering
- 14 Safety-Critical Systems
- 15 Future Trends in Model-Based Engineering
- 16 Final exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	10 Midterm	←	15 MARKS	To be freely distributed among possible assessments			25
8 to 12	10 Midterm		20 MARKS			→	30
13 to 15	←		10 MARKS			→	10
16 or 17	35 Final						35
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

- Textbook** • Bruce Powel Douglass, "Real-Time UML Workshop for Embedded Systems," Elsevier.
- Other** • Jon Holt, "SysML for Systems Engineering," IET.

ECE4312 Internet of Vehicles (IoV)**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3301	4	7	1	2	0	2

COURSE AIM

This course will introduce students to emerging technologies, standards, and applications in vehicular ad-hoc networks (VANETs) and facilitate a discussion of issues in building VANETs. These issues include VANET architectures, communication protocols, routing and MAC protocols, broadcast protocols, applications, and performance mobility and traffic modeling. . Examples of emerging applications of vehicular communications in Intelligent Transportation Systems will also be studied and discussed.

COURSE WEEKLY CONTENTS

- 1 Introduction to the course.
- Vehicular ad hoc Networks (VANET): Definition, Evolution, and Standardization.
- 2 VANET Architectures.
- 3 VANET Applications, Challenges, and Requirements.
- 4 V2X Communication Protocols.
- 5 Medium Access Control Protocols for Vehicular Ad Hoc Networks.
- 6 Data Dissemination and Broadcasting in Vehicular Networks.
- 7 7th week Assessment.
- 8 Routing in VANET (Part 1): Ad-hoc or Topology Driven Routing, Location Based Routing, Cluster Based Routing
- 9 Routing in VANET (Part 2): Broadcast Routing, Geocast Routing.
- 10 VANET Mobility Models.
- 11 VANET Traffic Models.
- 12 12th week Assessment.
- 13 Emerging VANET Applications
- 14 VANET Security and Privacy Issues.
- 15 Simulation Project Submission.
- 16 Final Exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	☐	10 MARKS			☐	30
8 to 12	☐	20 MARKS			☐		20
13 to 15	☐	10 MARKS			☐		10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

- | | |
|-----------------|--|
| Textbook | Claudia Campolo, Antonella Molinaro, Riccardo Scopigno. Vehicular ad hoc Networks: Standards, Solutions, and Research. Springer Publications, latest edition.
Chen, Wai, ed. Vehicular communications and networks: Architectures, protocols, operation and deployment. Elsevier Publications, latest edition.
Anand Paul, Naveen Chilamkurti, Alfred Daniel, and Seungmin Rho. Intelligent-Vehicular-Networks-and-Communications-Fundamentals-Architectures-and-Solutions. Elsevier Science Publications, latest edition. |
| Other | <ul style="list-style-type: none"> Hassnaa Moustafa and Yan Zhang. Vehicular Networks: Techniques, Standards, and Applications, latest edition. Auerbach Publications, USA. |

ECE4406 Introduction to Data Mining**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3601	4	7 or 8	2	2	0	3

COURSE AIM

To get the students familiar techniques for preprocessing data before mining. Present methods for mining frequent patterns, associations, and correlations. Present methods for data classification and prediction and data-clustering approaches. Apply the concepts on real life data sets. Learn and use Python software to perform data mining tasks.

COURSE WEEKLY CONTENTS

1	Introduction to Data Mining and tools
1	Data Mining and Knowledge Representation
2	Data Similarity and Data preprocessing
3	Classification – Decision Trees
4	Overfitting and Evaluation Metrics
5	Rule- Based Classifier
6	7th week exam
7	Naïve Bayes Classifier
8	Artificial Neural Networks and Ensemble Classification
9	Association Analysis
10	Association Analysis
11	12th week exam
12	Clustering
13	Clustering
14	Revision

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm1	✓			✓		30
8 to 12	20 Midterm2						20
13 to 15		✓		✓	✓		10
16 or 17	40 Final						40
Total	Exams	Assign	Quizzes	Reports	Present.	Lab.	100

REFERENCES**REFERENCES**

Textbook	• Witten, Ian H., et al. Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann, 2016
Other	• Tan, P., Steinbach, M. & Kumar, V: Introduction to Data Mining. Pearson, 2014 • Han, J., Kamber, M., & Pei, J. Data mining: Concepts and techniques (3rd ed.). Waltham: Morgan Kaufmann, 2011

ECE4409 Localization Techniques for Autonomous Driving**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
EBA3206 ECE2104	4	7	2		2	3

COURSE AIM

This course provides students with in-depth knowledge of localization techniques and technologies used in autonomous driving, including GPS, LIDAR, and SLAM.

COURSE WEEKLY CONTENTS

1	Introduction to Localization
2	GPS and GNSS Systems
3	Dead Reckoning and Inertial Navigation Systems
4	LIDAR and its Applications in Localization
5	Visual Odometry
6	Sensor Fusion for Localization
7	7th-week exam
8	Kalman Filters
9	Particle Filters
10	Simultaneous Localization and Mapping (SLAM)
11	Advanced SLAM Techniques
12	12th-week exam
13	Localization in Dynamic Environments
14	Localization Accuracy and Reliability
15	Case Studies in Autonomous Vehicle Localization
16	Final exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	10 Midterm	←	15	MARKS		→	25
8 to 12	10 Midterm		20	MARKS		→	30
13 to 15	←		10	MARKS		→	10
16 or 17	35 Final						35
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	- Hugh Durrant-Whyte and Tim Bailey, "Simultaneous Localization and Mapping: Part I," IEEE Robotics & Automation Magazine. - Thrun, S., Burgard, W., & Fox, D. (2005). Probabilistic Robotics. MIT Press
Other	Steven M. LaValle, "Planning Algorithms," Cambridge University Press.

ECE4512 Fundamentals of Accelerated Computing**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2104	4	8	1	2		2

COURSE AIM

This course introduces students to the fundamental tools and techniques for accelerating C/C++ applications to run on massively parallel GPUs with CUDA. Students will learn how to write code, configure code parallelization with CUDA, optimize memory migration between the CPU and GPU accelerator, and implement the workflow that they've learned on a new task—accelerating a fully functional, but CPU-only, particle simulator for observable massive performance gains. The course provides students with the foundational knowledge of accelerated computing techniques and their applications in autonomous systems.

COURSE WEEKLY CONTENTS

1	Introduction to Accelerated Computing
2	Overview of Parallel Computing Architectures
3	CUDA Programming Basics
4	OpenCL Programming Basics
5	Performance Optimization Techniques
6	GPU Architectures and Design
7	7th-week exam
8	Multi-GPU Programming
9	FPGA Programming for Accelerated Computing
10	Real-Time Data Processing with Accelerated Computing
11	Applications of Accelerated Computing in Autonomous Systems
12	12th-week exam
13	Case Studies in Accelerated Computing
14	Future Trends in Accelerated Computing
15	Ethics and Safety in Accelerated Computing
16	Final exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	10 Midterm	←	15	MARKS			25
8 to 12	10 Midterm		20	MARKS			30
13 to 15	←		10	MARKS			10
16 or 17	35 Final						35
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• David B. Kirk and Wen-mei W. Hwu, "Programming Massively Parallel Processors: A Hands-on Approach," Morgan Kaufmann.
Other	• John Cheng, Max Grossman, and Ty McKercher, "Professional CUDA C Programming," Wrox.

ECE4516 Autonomous Vehicle Middleware**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3503	4	7	1	2		2

COURSE AIM

The course provides students with the necessary knowledge of middleware platforms used in autonomous vehicle systems, focusing on communication and integration.

COURSE WEEKLY CONTENTS

- 1 Introduction to Middleware in Autonomous Vehicles
- 2 Communication Protocols and Standards
- 3 Middleware Architecture and Design
- 4 Data Management and Storage Solutions
- 5 Real-Time Data Processing
- 6 Middleware for Sensor Integration
- 7 7th-week exam
- 8 Distributed Systems and Middleware
- 9 Security and Reliability in Middleware
- 10 Middleware for Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) Communication
- 11 Cloud Computing and Edge Computing in Autonomous Vehicles
- 12 12th-week exam
- 13 Case Studies in Middleware Applications
- 14 Middleware Performance Optimization
- 15 Future Trends in Autonomous Vehicle Middleware
- 16 Final exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	10 Midterm	←	15 MARKS	To be freely distributed among possible assessments			25
8 to 12	10 Midterm		20 MARKS			→	30
13 to 15	←		10 MARKS			→	10
16 or 17	35 Final						35
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Judith M. Myerson, "The Complete Book of Middleware," CRC Press.
Other	• Kai Hwang, Geoffrey C. Fox, and Jack J. Dongarra, "Distributed and Cloud Computing: From Parallel Processing to the Internet of Things," Morgan Kaufmann.

ECE4607 Introduction to Intelligent Human-Computer Interaction

COURSE INFORMATION

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3601, ECE2104	4	7 or 8	1	2	0	2

COURSE AIM

The course Introduces the importance and applications of Human Computer Interaction (HCI). It Discusses the basics of human psychology and their implications in interface design. Introduces the different disciplines, including human psychology and different computer tools in the sector of Graphics, Artificial Intelligence and Software Engineering. The course includes projects in real life case-studies, introducing concepts of collecting usability requirements through user surveys, design improvement and interface personalization for users with a wide range of abilities. The course will handle different HCI technologies including Computer Supported Cooperative Work, Pervasive Computing, Interaction Design, Affective Computing, Accessible Computing and Information Visualization. Novel modalities of interaction including gesture, eye-gaze and head tracking interfaces will be discussed.

COURSE WEEKLY CONTENTS

1	Introduction to Human Computer Interaction (HCI), applications, and the basics of human psychology and their implications in interface design.
2	Perception Technology and different disciplines.
3	Cognition and Motor-action
4	User Modeling in HCI.
5	Application Case-studies.
6	Usability analysis techniques.
7	7th Week Exam.
8	Alternative Input Modalities
9	Multimodal Interaction.
10	Computer Supported Cooperative Work and Pervasive Computing.
11	Accessible Computing and Information Visualization.
12	Novel modalities of interaction including gestures, eye-gaze tracking interfaces.
13	12th Week Exam.
14	Novel modalities of interaction using head tracking interfaces and Spatial Audio Interface.
15	Projects & Application Case-Studies.
16	Final Exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10 MARKS	To be freely distributed among possible assessments			30
8 to 12	←		20 MARKS			→	20
13 to 15	←		10 MARKS			→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Alan Dix, Janet E. Finlay, Gregory D. Abowd, and Russell Beale, "Human-Computer Interaction", Pearson Education, 6th Edition, 2016
Other	- Human-Computer Interaction - HCI International 2019 (Lecture Notes in Computer Science, 11566-11597) 1st ed. 2019 Edition. - David Benyon, "Designing Interactive Systems Paperback", Pearson Education (US), 2010.

ECE4613 Autonomous Mobile Robotics**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
EEE3108	4	7	1		2	2

COURSE AIM

The course introduces students to the principles and technologies of autonomous mobile robotics, including perception, planning, and control. Students will learn techniques and principles for designing and developing mobile robots that interact autonomously with their environment.

COURSE WEEKLY CONTENTS

1	Introduction to Mobile Robotics
2	Kinematics of Mobile Robots
3	Robot Perception Systems
4	Localization Techniques in Mobile Robots
5	Path Planning Algorithms
6	Motion Control and Navigation
7	7th-week exam
8	Multi-Robot Systems
9	Robot Operating System (ROS) Basics
10	Advanced Path Planning and Obstacle Avoidance
11	SLAM in Mobile Robots
12	12th-week exam
13	Autonomous Navigation in Dynamic Environments
14	Human-Robot Interaction
15	Future Trends in Mobile Robotics
16	Final exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	10 Midterm	←	15	MARKS			25
8 to 12	10 Midterm		20	MARKS			30
13 to 15	←		10	MARKS			10
16 or 17	35 Final						35
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Roland Siegwart, Illah R. Nourbakhsh, and Davide Scaramuzza, "Introduction to Autonomous Mobile Robots," MIT Press.
Other	• Sebastian Thrun, Wolfram Burgard, and Dieter Fox, "Probabilistic Robotics," MIT Press.

ECE4614 Reinforcement Learning**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3601	4	7	2		2	3

COURSE AIM

The course provides a comprehensive understanding of reinforcement learning and its applications in autonomous driving.

COURSE WEEKLY CONTENTS

1	Introduction to Reinforcement Learning
2	Markov Decision Processes
3	Dynamic Programming
4	Monte Carlo Methods
5	Temporal-Difference Learning
6	Q-Learning and SARSA
7	7th-week exam
8	Policy Gradient Methods
9	Deep Reinforcement Learning
10	Applications of RL in Autonomous Driving
11	Multi-agent Reinforcement Learning
12	12th-week exam
13	Safety and Robustness in RL
14	Case Studies in RL for Autonomous Vehicles
15	Future Directions in RL
16	Final exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	10 Midterm	←	15 MARKS			→	25
8 to 12	10 Midterm		20 MARKS			→	30
13 to 15	←		10 MARKS			→	10
16 or 17	35 Final						35
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook • Richard S. Sutton and Andrew G. Barto, "Reinforcement Learning: An Introduction," MIT Press.

Other • Csaba Szepesvári, "Algorithms for Reinforcement Learning," Morgan & Claypool Publishers

ECE4615 Artificial Intelligence for Autonomous Vehicles**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3601	4	7	2		2	3

COURSE AIM

This course provides students with AI techniques specific to autonomous vehicles, covering machine learning, neural networks, and decision-making processes. The course consists of both theoretical and experimental elements and is related to current research and development.

COURSE WEEKLY CONTENTS

1	Introduction to AI in Autonomous Vehicles
2	Machine Learning Basics: Supervised and Unsupervised Learning
3	Neural Networks and Deep Learning
4	Convolutional Neural Networks (CNNs) for Perception
5	Recurrent Neural Networks (RNNs) for Prediction
6	Reinforcement Learning: Basics and Applications
7	7th-week exam
8	Path Planning using AI
9	Behaviour Prediction and Decision Making
10	Sensor Fusion Techniques using AI
11	AI for Object Detection and Classification
12	12th-week exam
13	AI in Adverse Weather and Lighting Conditions
14	Ethics and Safety in AI for Autonomous Vehicles
15	Future Trends in AI for Autonomous Vehicles
16	Final exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	10 Midterm	←	15	MARKS		→	25
8 to 12	10 Midterm		20	MARKS		→	30
13 to 15	←		10	MARKS		→	10
16 or 17	35 Final						35
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Stuart Russell and Peter Norvig, "Artificial Intelligence: A Modern Approach," Pearson.
Other	• Ian Goodfellow, Yoshua Bengio, and Aaron Courville, "Deep Learning," MIT Press.

ECE4704 Advances in Computer Vision**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2104	3	5	2	2		3

COURSE AIM

In this course, students will develop an understanding of the capabilities and limitations of state-of-the-art computer vision solutions for autonomous driving.

COURSE WEEKLY CONTENTS

1	Introduction to Advanced Computer Vision
2	Image Enhancement and Restoration
3	Feature Detection and Matching
4	Advanced Object Detection Techniques
5	Deep Learning for Image Classification
6	Advanced Semantic Segmentation
7	7th-week exam
8	3D Reconstruction and Stereo Vision
9	Motion Analysis and Optical Flow
10	Activity Recognition and Scene Understanding
11	Visual SLAM (Simultaneous Localization and Mapping)
12	12th-week exam
13	Vision-based Navigation for Autonomous Vehicles
14	Adversarial Attacks and Robustness in Computer Vision
15	Future Trends in Computer Vision
16	Final exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	10 Midterm	←	15	MARKS		→	25
8 to 12	10 Midterm		20	MARKS		→	30
13 to 15	←		10	MARKS		→	10
16 or 17	35 Final						35
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Richard Szeliski, "Computer Vision: Algorithms and Applications," Springer.
Other	• Dario S. Taubman and Michael W. Marcellin, "JPEG2000 Image Compression Fundamentals, Standards and Practice," Springer. • Andrea Vedaldi, Andrew Zisserman, and K. Mikolajczyk, "Efficient Dense Stereo, Segmentation-aware 3D Object Detection and SLAM," MIT Press.

ECE4705 Perception Software Models for Autonomous Driving**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3601 ECE2104	4	7	1	2		2

COURSE AIM

This course provides students with knowledge of perception models and software used in autonomous driving.

COURSE WEEKLY CONTENTS

1	Introduction to Perception in Autonomous Driving
2	Computer Vision Basics
3	Image Processing Techniques
4	Object Detection Algorithms
5	Object Tracking Techniques
6	Semantic Segmentation
7	7th-week exam
8	3D Perception using LIDAR
9	Sensor Fusion for Perception
10	Deep Learning for Perception
11	Real-time Perception Systems
12	12th-week exam
13	Perception in Adverse Conditions
14	Case Studies in Perception for Autonomous Driving
15	Future Trends in Perception Technology
16	Final exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	10 Midterm	←	15	MARKS		→	25
8 to 12	10 Midterm		20	MARKS		→	30
13 to 15	←		10	MARKS		→	10
16 or 17	35 Final						35
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Richard Szeliski, "Computer Vision: Algorithms and Applications," Springer.
Other	• Alexander H. Levis and Lanny J. Treinish, "Handbook of Perception and Action," Academic Press.

4.4.3. Concentration 3: Multimedia & Intelligent Systems

ECE3106 Introduction to Software Engineering

COURSE INFORMATION

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2104 ECE3511	4	7	1	2	0	2

COURSE AIM

In this course, students learn and gain practical experience with software engineering principles, techniques, and tools. Topics in this course include requirements analysis, specification, design, abstraction, programming style, testing, maintenance, communication, teamwork, and software project management. The practical experience centers on a team project, in which a software development project is carried through all the stages of the software life cycle. Particular emphasis is placed on communication skills and on developing maintainable software. In-class activities further provide a hands-on experience in using state-of-the-art techniques and tools.

COURSE WEEKLY CONTENTS

1	Introduction to Software Engineering: different software models.
2	Software Development LifeCycle
3	Requirements Engineering
4	Use cases and Data Modelling
5	Agile and Scrum (Part 1)
6	Agile and Scrum (Part 2)
7	7 th Week Exam
8	Software Architecture: different architecture styles, uses and benefits
9	UML diagrams (Part 1): Sequence Diagram, Class Diagram, Collaboration and Deployment.
10	UML diagrams (Part 2): Activity and State Diagram
11	Software Testing (Part 1): Build Systems and Code Review
12	12 th week Assessment.
13	Software Testing (Part 2): Unit Testing
14	Continuous Integration and Delivery – DevOps; Interviewing and Wrap-up
15	Project Seminars
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	15 Midterm	←	15 MARKS			→	30
		To be freely distributed among possible assessments					
8 to 12	10 Midterm	←	10 MARKS			→	20
		To be freely distributed among possible assessments					
13 to 15	←		10 MARKS			→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Ian Sommerville, “Software Engineering”, 10 th Edition, 2015.
Other	• John Ousterhout , “A Philosophy of Software Design”, 1 st Edition, 2018

ECE4406 Introduction to Data Mining**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3601	4	7	2	2	0	3

COURSE AIM

To get the students familiar techniques for preprocessing data before mining. Present methods for mining frequent patterns, associations, and correlations. Present methods for data classification and prediction and data-clustering approaches. Apply the concepts on real life data sets. Learn and use Python software to perform data mining tasks.

COURSE WEEKLY CONTENTS

1	Introduction to Data Mining and tools
2	Data Mining and Knowledge Representation
3	Data Similarity and Data preprocessing
4	Classification – Decision Trees
5	Overfitting and Evaluation Metrics
6	Rule- Based Classifier
7	7th week exam
8	Naïve Bayes Classifier
9	Artificial Neural Networks and Ensemble Classification
10	Association Analysis
11	Association Analysis
12	12th week exam
13	Clustering
14	Clustering
15	Revision

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm1	✓			✓		30
8 to 12	20 Midterm2						20
13 to 15		✓		✓	✓		10
16 or 17	40 Final						40
Total	Exams	Assign	Quizzes	Reports	Present.	Lab.	100

REFERENCES**REFERENCES**

Textbook	• Witten, Ian H., et al. Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann, 2016
Other	• Tan, P., Steinbach, M. & Kumar, V: Introduction to Data Mining. Pearson, 2014 • Han, J., Kamber, M., & Pei, J. Data mining: Concepts and techniques (3rd ed.). Waltham: Morgan Kaufmann, 2011

ECE4602 Neural Networks**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2104	3	6	1	2	0	2

COURSE AIM

This course will provide students with detailed skills to use Artificial neural networks for solving many types of engineering problems such as mapping, clustering, and constrained optimization, in such areas as pattern recognition, signal processing, and control systems.

COURSE WEEKLY CONTENTS

1	Introduction to basic concepts of neural networks, applications and different architectures.
2	Perceptron Network and Perceptron Learning Rule.
3	Multilayer Perceptron and Back-propagation Learning Rule
4	Building Neural Networks using MATLAB.
5	Competitive Neural Networks.
6	Kohonen Self-Organizing Learning Rules.
7	7th Week Exam.
8	Hebbian Networks and Hebbian Learning Rule.
9	Adaptive Reasoning Theory (ART) – part 1.
10	Adaptive Reasoning Theory (ART) – part 2.
11	Hopfield Neural Networks.
12	12th Week Exam.
13	Convolutional Neural Networks
14	Implementing Deep Learning Architectures.
15	Neural networks case studies & term work Projects.
16	Final Exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10 MARKS	To be freely distributed among possible assessments			30
8 to 12	←		20 MARKS			→	20
13 to 15	←		10 MARKS			→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Charu C. Aggarwal, "Neural Networks and Deep Learning", 1 st ed., 2018.
Other	• Simon O. Haykin, "Neural Networks And Learning Machines" 3 rd ed, 2008

ECE4603 Intelligent Robotics**COURSE INFORMATION**

Prerequisites		Academic Year & Level		Teaching Methods			Credit Hrs.
		Year	Semester	Lecture	Tutorial	Laboratory	
ECE3204	EEE3108	4	7	2	2	-	3

COURSE AIM

To teach students the principles and techniques of designing intelligent robotic systems.

COURSE WEEKLY CONTENTS

1	Introduction.
2	Object location.
3	General transformation.
4	Kinematics: Homogenous Transformation.
5	Kinematics: Forward / Inverse kinematics.
6	Introduction to AI.
7	7th week Exam.
8	Robot Sensors.
9	Image Processing.
10	Pattern recognition and computer vision.
11	Autonomous Mobile Robots.
12	12th week Exam.
13	Trajectory planning for Robot.
14	Robot Control.
15	Revision.
16	Application.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS		→	30
8 to 12	←		20	MARKS		→	20
13 to 15	←		10	MARKS		→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Mark Song, "Robot Modeling and Control", John Wiley, latest edition.
Other	• J. J. Craig, "Introduction to Robotics Mechanics and Control", Prentice Hall, latest edition. • Phillip j. McKerrow, "Introduction to Robotics", Addison Wesley, latest edition.

ECE4604 Web Engineering**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2104 ECE3511	4	7	2	1	2	3

COURSE AIM

This course aims to introduce the methods and techniques used in Web-based system development. This course draws upon previous programming and computing experience to develop practical web development and maintenance skills. The course teaches the students the Design and Analysis using HTML, CSS, Java Script, Bootstrap, Node JS, PHP, MYSQL-NOSQL, ASP.NET, and Ajax. This course is intended for students with knowledge of both Internet communication concepts and an introductory programming knowledge..

COURSE WEEKLY CONTENTS

1	Introduction to Web hosting, Client-Side, Server-Side and Full stack development. Web System Design Disciplines, Tools & Methods
2	HTML : general introduction – basic tags.
3	CSS selectors, properties of CSS.
4	Responsive Web Applications: Bootstrap Framework Supporting CSS.
5	Basic of Java Script Programming.
6	Understand of Document Object Model (DOM)
7	7 th week Assessment, JQuery, React JS and Angular Frameworks.
8	Introduction to PHP
9	MYSQL Web Development
10	PHP and file I/O
11	AJAX
12	12 th week Assessment
13	Introduction to Node JS
14	Introduction to Express JS
15	ASP.NET
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams		Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20	Midterm	← 5	MARKS			→ 5	30
To be freely distributed among possible assessments								
8 to 12	←		20 MARKS			→		20
13 to 15	←		10 MARKS			→		10
16 or 17	40	Final						40
Total	Exams		Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Danny Goodman ” Dynamic HTML: The Definitive Reference: A Comprehensive Resource for XHTML, CSS, DOM, JavaScript”
Other	• Douglas Crockford, "JavaScript: The Good Parts" • Martin S. Roden, "JavaScript: The Definitive Guide, 6th Edition".

ECE4605 Mobile Application**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2104	4	7	2	2	1	3

COURSE AIM

This course introduces MOBILE APPLICATION DEVELOPMENT to students who are already familiar with java Programming language and database systems. Advanced features of MOBILE APPLICATION DEVELOPMENT will be introduced through applications. Among those advanced features: Building Graphical User Interface (GUI), Access mobile resources (Camera, Sensors, Bluetooth), processing Multimedia resources (Video and Audio processing), Location tracking and map allocation, Android storage options and data management (database connectivity using SQLite), and Animations applications. The course also covers mobile application fundamentals, application components, intents, application anatomy compression between multiple mobile operating system and define android architecture and its platform. The course also introduces the basics of cross platforms (XAMARIN) for mobile developments.

COURSE WEEKLY CONTENTS

1	Introduction to mobile application development
2	Android Architecture and platform
3	Mobile application fundamentals
4	Graphical user interface
5	Access mobile resources + Storage (Contact, Gallery) + Timer + Camera
6	Processing Multimedia resources + Video + Audio + Bluetooth + Sensors
7	7th Week Exam
8	Accessing Google Map + Linking with database + Graphics and Animations
9	Introduction to interfacing with external Micro-controller
10	Introduction to Cross platforms Using Xamarin
11	Layouts & Interfaces
12	12th Week Exam
13	Building Mobile Applications Using Xamarin
14	Projects & Application Case-studies on Cross platforms
15	Projects and Case studies.
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS	→		30
8 to 12	←		20	MARKS	→		20
13 to 15	←		10	MARKS	→		10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

- | | |
|-----------------|---|
| Textbook | • Introduction to Android Application Development: Android Essential, Fourth Edition. Available on Kortext Portal. Authors: Joseph Annuzzi Jr., Lauren Darcey, Shane Conder |
| Other | - The Android Developer's Cookbook, James Steele, Nelson To
- Professional Android Sensor Programming, Greg Milette, Adam Stroud |

ECE4606 Computational Intelligence in Biology**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2104 ECE3601	4	7	1	2	0	2

COURSE AIM

The goal of the course is to enhance the knowledge of students with the help of AI technology to apply Artificial Intelligence in Biology. The course introduces students with advances in biology such as sequencing of the human genome and advances in imaging techniques have led to an explosion of data in the biosciences {Genomics, Microarray Data and Protein Interactions}. Students learn how AI can be used in the Biotech & healthcare industry, making them the most competitive and dynamic knowledge partners in this Artificial intelligence revolution. This course sets out practical solutions to the application of machine learning, artificial intelligence and bio-inspired computing to solving biomedical problems. The strength and limitations of the approaches are discussed, and practical solutions provided in a form accessible to non-computer scientists.

COURSE WEEKLY CONTENTS

1	Biological Intelligence Vs Artificial Intelligence.
2	Overview of AI Basics: concepts, terminologies and workflow, and Machine Learning.
3	Deep learning architectures, platforms and Design Tools
4	Artificial Intelligence for Biomarker Discovery.
5	AI in Precision Medicine
6	AI in health diagnostics / Confluence of AI and Smart Devices for Monitoring Health and Disease.
7	7th Week Exam.
8	AI in medical imaging.
9	Data in the biosciences.
10	AI-driven applications for drug design, lead optimization, and clinical trials.
11	Artificial Intelligence and Synthetic Biology.
12	12th Week Exam.
13	Practical application case-studies of machine learning methods to solve biological/medical data problems.
14	AI in biology: Risks involved and ethical concerns.
15	Future Prospects of AI in healthcare.
16	Final Exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS			30
8 to 12	←		20	MARKS			20
13 to 15	←		10	MARKS			10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Ujjwal Maulik, Sanghamitra Bandyopadhyay, Jason T. Wang, "Computational Intelligence and Pattern Analysis in Biology Informatics", (Wiley Series in Bioinformatics Book 13), 2018, WILEY
----------	--

ECE4607 Introduction to Intelligent Human-Computer Interaction

COURSE INFORMATION

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3601, ECE2104	4	7 or 8	1	2	0	2

COURSE AIM

The course Introduces the importance and applications of Human Computer Interaction (HCI). It Discusses the basics of human psychology and their implications in interface design. Introduces the different disciplines, including human psychology and different computer tools in the sector of Graphics, Artificial Intelligence and Software Engineering. The course includes projects in real life case-studies, introducing concepts of collecting usability requirements through user surveys, design improvement and interface personalization for users with a wide range of abilities. The course will handle different HCI technologies including Computer Supported Cooperative Work, Pervasive Computing, Interaction Design, Affective Computing, Accessible Computing and Information Visualization. Novel modalities of interaction including gesture, eye-gaze and head tracking interfaces will be discussed.

COURSE WEEKLY CONTENTS

1	Introduction to Human Computer Interaction (HCI), applications, and the basics of human psychology and their implications in interface design.
2	Perception Technology and different disciplines.
3	Cognition and Motor-action
4	User Modeling in HCI.
5	Application Case-studies.
6	Usability analysis techniques.
7	7th Week Exam.
8	Alternative Input Modalities
9	Multimodal Interaction.
10	Computer Supported Cooperative Work and Pervasive Computing.
11	Accessible Computing and Information Visualization.
12	Novel modalities of interaction including gestures, eye-gaze tracking interfaces.
13	12th Week Exam.
14	Novel modalities of interaction using head tracking interfaces and Spatial Audio Interface.
15	Projects & Application Case-Studies.
16	Final Exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10 MARKS	To be freely distributed among possible assessments			30
8 to 12	←		20 MARKS			→	20
13 to 15	←		10 MARKS			→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Alan Dix, Janet E. Finlay, Gregory D. Abowd, and Russell Beale, "Human-Computer Interaction", Pearson Education, 6th Edition, 2016
Other	- Human-Computer Interaction - HCI International 2019 (Lecture Notes in Computer Science, 11566-11597) 1st ed. 2019 Edition. - David Benyon, "Designing Interactive Systems Paperback", Pearson Education (US), 2010.

ECE4608 Natural Language Processing**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3601 ECE2104	4	7 or 8	1	2		2

COURSE AIM

To get the students familiar with the current and future applications of NLP; describe a fundamental technique for processing language for several subtasks, such as morphological processing, parsing, summarization etc and understand how these techniques draw on and relate to other areas of computer science.

COURSE WEEKLY CONTENTS

1	Introduction. Brief history of NLP research, some current applications, components of NLP systems.
2	Finite-state techniques. Inflectional and derivational morphology, finite-state automata in NLP, finite-state transducers.
3	Prediction and part-of-speech tagging. Corpora, simple N-grams, word prediction, stochastic tagging, evaluating system performance.
4	Context-free grammars and parsing. Generative grammar, context-free grammars, parsing with context-free grammars, weights and probabilities. Some limitations of context-free grammars.
5	Dependencies. Dependency structures. English as an outlier. Universal dependencies. Introduction to dependency parsing.
6	Compositional semantics. Compositional semantics. Logical representations. Compositional semantics and lambda calculus. Inference and robust entailment. Negation.
7	7 th week exam
8	Lexical semantics. Semantic relations, WordNet, word senses.
9	Distributional semantics. Representing lexical meaning with distributions. Similarity metrics.
10	Distributional semantics and deep learning. Embeddings. Grounding. Multimodal systems and visual question answering.
11	Discourse processing. Anaphora resolution, summarization.
12	12 th week exam
13	Language generation and regeneration. Components of a generation system. Summarization. Generation of referring expressions.
14	Recent NLP research. Some recent NLP research- Part 1.
15	Recent NLP research. Some recent NLP research – Part 2.
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm1	✓			✓		30
8 to 12	20 Midterm2						20
13 to 15		✓		✓	✓		10
16 or 17	40 Final						40
Total	Exams	Assign	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook

Other

Jurafsky, Dan, and James H. Martin. "Speech and Language Processing (3rd draft ed.)." (2019).
Bird, Steven, Ewan Klein, and Edward Loper. Natural language processing with Python: analyzing text with the natural language toolkit. " O'Reilly Media, Inc.", 2009.

ECE4616 Introduction to Generative AI**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3601	4	8	2	0	2	3

COURSE AIM

This course introduces the fundamental concepts and techniques of generative AI, covering various models, applications, and ethical considerations. By the end of the course, students will have a solid understanding of generative AI principles and practical skills to build and evaluate generative models.

COURSE WEEKLY CONTENTS

1	Introduction to Generative AI
2	Fundamentals of Machine Learning and Deep Learning
3	Generative Models Overview
4	Generative Adversarial Networks (GANs)
5	Variational Autoencoders (VAEs)
6	Autoregressive Models
7	7th Week Exam
8	Advanced GAN Techniques (Conditional GANs, StyleGAN, CycleGAN, etc)
9	Text Generation with Generative Models
10	Image Generation and Style Transfer
11	Evaluation Metrics for Generative Models.
12	12th Week Exam
13	Applications of Generative AI
14	Ethical and Societal Implications of Generative AI.
15	Future Directions in Generative AI
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS	→		30
8 to 12	←		20	MARKS	→		20
13 to 15	←		10	MARKS	→		10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Foster, D. (2019). Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play. O'Reilly Media
Other	Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.

ECE4702 Multimedia & Virtual Reality Systems**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2104	4	7	2	2	1	3

COURSE AIM

The objective of this course is to provide students with a basic understanding of multimedia systems. This course focuses on topics in multimedia information representation and relevant signal processing aspects, and multimedia standards especially on the audio, image and video compression. All of these topics are important in multimedia industries. Through this course, students are expected to achieve a basic understanding of multimedia and VR systems. With such background equipment, students would be able to evaluate more advanced or future multimedia systems. This course will also arouse students' interest in the course and further motivate them towards developing their career in the area of multimedia and internet applications. By the end of this course, the learner will have created and deployed a VR application. Student will understand the physical principles of VR and you will use that knowledge to create a comfortable, high-performance VR application using Unity. Students will get experiencing on design 3D Models, Graphic User Interfaces (GUI), and Experiencing development environment and toolset using Unity – a popular industry standard game engine. VR development is an amazing career, with both demand and salaries rising.

COURSE WEEKLY CONTENTS

1	Introduction to Multimedia Systems, and Virtual Reality Systems (VR) + Platforms and Paradigms. Learn the difference between native, game engines, and web platforms + Learn about the different platforms currently available for VR development
2	Introduction to Multimedia applications & Challenges + Multimedia documents, storage and representations+ Images, audio and video media + Importing multimedia documents
3	Multimedia data processing and Filtering + Image registration, Matching Techniques and multimedia retrieval systems + Rotation and Transformation
4	3D Object modeling + 3D objects processing, storage and loading + Design Object Patterns, Manipulating Objects, Creating and processing of Multimedia Databases
5	Multimedia Compression standards (Image, Video and Audio)+ JPEG Compression and DCT + Transform + Introduction to Blending, Auto desk Mishmixer + Experiencing AR, Development environment and toolset
6	Video representation + Video Compression Techniques (MPEG)
7	7th Week Exam
8	Introduction to Virtual Reality Development/Design + Unity Virtual Reality Fundamentals, Setting up the Development environment + Intro. to Unity and Vuforia + Downloading required packages + Integrating Vuforia into Unity + Panoramic imaging, future Directions in Panoramic imaging + Panoramic Image generation using Sift technique.
9	Virtual Reality Development/Design Graphic User Interfaces (GUI), scripting and programming, Texture, Material and effects, Unity Lightings
10	Importing and Exporting Assets + Types of Assets, importing 3D Models, Importing Scripts and Asset Packages + Learn how tracking in VR works
11	Combining Assets, GUIs, animation and application case studies
12	Generating Characters + Gamification concepts and applications
13	Developing VR environments with animation
14	Application case-Studies & term-work projects
15	Application case-Studies & term-work projects
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS			30
8 to 12	←		20	MARKS			20
13 to 15	←		10	MARKS			10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	- Multimedia Signals and Systems: Basic and Advanced Algorithms for Signal Processing 2nd ed. 2016 Edition by Srdjan Stanković (Author), Irena Orović (Author), Ervin Sejdić (Author) - Building Virtual Reality with Unity and Steam VR Jeff W Murray, A K Peters/CRC Press, 1st Edition, 2017
Other	- Virtual Reality Technology and Applications Authors: Mihelj, Matjaž, Novak, Domen, Beguš, Samo, 2014

ECE4703 Image Processing & Pattern Recognition**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE4711	4	8	2	2	0	3

COURSE AIM

- Getting the student familiar with the field of image processing and pattern recognition
- Study of basic image processing algorithms like edge detection, filtering, image enhancement and morphological techniques.
- Introduction to pattern recognition concepts as supervised and un-supervised learning.

COURSE WEEKLY CONTENTS

1	Introduction to digital image processing
2	Digital Image fundamentals
3	Spatial filtering – 1: basic mathematical tools and intensity transformation functions, piecewise linear transformation functions
4	Spatial filtering – 2: Histogram processing and fundamental of spatial filtering
5	Spatial filtering – 3: low-pass; high-pass; band-pass & band-reject filters and combined spatial filters
6	Morphological Image Processing: erosion, dilation, opening & closing and basic morphological algorithms
7	7 th week Assessment
8	Image segmentation: point detection, line detection, edge detection, thresholding, segmentation using region growing
9	Feature Extraction: boundary processing, boundary feature descriptors, region descriptors, and principle components as features descriptors
10	Image pattern classification – 1: pattern classes, performance measures, pattern classification using prototype, byes classifier, K-nearest neighbor
11	Image pattern classification – 2: Support vector machine, neural networks
12	12 th week Assessment
13	Image pattern classification – 3: Deep convolution neutral networks and clustering techniques
14	Application case studies
15	Application case studies
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	5	5				30
8 to 12	10 EXAM			5	5		20
13 to 15		5	5				10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• R. Gonzalez and R. Woods, "Digital Image Processing", Pearson Hall, Fourth Edition.
Other	• E. Gose, R. Johnsonbaugh, "Pattern Recognition and Image Analysis", Prentice Hall PTR

4.4.4. Concentration 4: High-Performance Computing Systems

ECE3106 Introduction to Software Engineering

COURSE INFORMATION

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2104 ECE3511	4	7	1	2	0	2

COURSE AIM

In this course, students learn and gain practical experience with software engineering principles, techniques, and tools. Topics in this course include requirements analysis, specification, design, abstraction, programming style, testing, maintenance, communication, teamwork, and software project management. The practical experience centers on a team project, in which a software development project is carried through all the stages of the software life cycle. Particular emphasis is placed on communication skills and on developing maintainable software. In-class activities further provide a hands-on experience in using state-of-the-art techniques and tools.

COURSE WEEKLY CONTENTS

1	Introduction to Software Engineering: different software models.
2	Software Development LifeCycle
3	Requirements Engineering
4	Use cases and Data Modelling
5	Agile and Scrum (Part 1)
6	Agile and Scrum (Part 2)
7	7 th Week Exam
8	Software Architecture: different architecture styles, uses and benefits
9	UML diagrams (Part 1): Sequence Diagram, Class Diagram, Collaboration and Deployment.
10	UML diagrams (Part 2): Activity and State Diagram
11	Software Testing (Part 1): Build Systems and Code Review
12	12 th week Assessment.
13	Software Testing (Part 2): Unit Testing
14	Continuous Integration and Delivery – DevOps; Interviewing and Wrap-up
15	Project Seminars
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	15 Midterm	←	15 MARKS			→	30
			To be freely distributed among possible assessments				
8 to 12	10 Midterm	←	10 MARKS			→	20
			To be freely distributed among possible assessments				
13 to 15	←		10 MARKS			→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

- | | |
|----------|--|
| Textbook | • Ian Sommerville, “Software Engineering”, 10 th Edition, 2015. |
| Other | • John Ousterhout , “A Philosophy of Software Design”, 1 st Edition, 2018 |

ECE3206 Computer Design and Performance Evaluation**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3203	3	6	1	2	0	2

COURSE AIM

The course introduces the student to the principles of design, build and test of special-purpose processors. Moreover, the students are introduced to the concepts of evaluating the performance of such processors. It is intended for the final year BS. Students or first year graduates specializing in computer engineering.

COURSE WEEKLY CONTENTS

1	Introduction to ISA-based Computer Design, Sequencing and Control.
2	Hardwired and Micro-Programmed Control.
3	Single-Cycle Hardwired Control and Multiple-Cycle Micro-Programmed Control.
4	Pipelined Control and Performance Evaluation.
5	Instruction Set Architecture and Addressing Evaluation.
6	Central Processing Unit Design.
7	7th week exam.
8	High Performance CPU Concepts.
9	Design Parameters; Area, Time, and Cost.
10	Operational Analysis.
11	M/G/I Queuing model.
12	12th Week Exam.
13	Discrete-Time Markov Chains.
14	Benchmark System Evaluation.
15	Revision
16	Final Exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS	→		30
8 to 12	←		20	MARKS	→		20
13 to 15	←		10	MARKS	→		10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• M.Mano, C.R.Kime, "Logic and Computer Design Fundamentals", Prentice Hall, latest edition.
Other	• M. Mano, "Computer System Architecture", Englewood Cliffs, NY: Prentice Hall, latest edition. • Patterson, D.A., and Hennessy J. L., "Computer Organization and Design: The Hardware/Software Interface", Morgan Kaufmann, latest edition.

ECE4109 Parallel Programming for Multicore Machines**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3503 ECE2104	4	8	2	2	0	3

COURSE AIM

The aim of the course is to provide students with a detailed understanding of writing effective programs to harness the unprecedented power provided by parallel computers to attain the highest levels of performance. covers different programming systems and methodologies such as OpenMP and MPI. Develop an understanding of analytical frameworks to understand performance, such as performance models, scalability analysis and efficiency. Analyze sequential programs and determine if they are worthwhile to parallelize.

develop, analyze, and implement algorithms for parallel computers. This applies both to computers with shared memory and with distributed memory.

COURSE WEEKLY CONTENTS

- 1 Introduction: Principles of parallel algorithm design and parallel computation models
- 2 Performance with Complexity: Moore's Law, Parallelism, Latency and Bandwidth, Pipelining, Branch Prediction
- 3 Caches and Memory Optimizations: Caches and Cache performance, Prefetching, Virtual Memory, Data Layout
- 4 Shared Memory Programming: Vectorization, SIMD, Tools, Shared Memory Machines, Parallel loops and OpenMP
- 5 Basic OpenMP: OpenMP parallel loop construct, Sharing of variables, Dependencies and restructuring, Loop schedules, Parallel construct
- 6 Advanced OpenMP: Synchronization, Critical sections, Sequential consistency, Flush construct, Case studies
- 7 7th week exam.
- 8 Performance Issues in OpenMP: Cache related performance issues, False sharing, Nested parallelism, Explicit dependencies, TasksContinue with Embedded Systems Programming
- 9 Distributed Memory Programming: Pthreads,, Distributed Memory Machines, Basic MPI
- 10 Other Distributed Models: Cost Model, One-sided communication, Hybrid programming (MPI + OpenMP)
- 11 Theoretical Models: Charm++
- 12 12th week exam
- 13 Parallel Algorithms Sorting algorithms: Algorithms for Broadcast/Reduction and collective operations, Scalability and IsoefficiencyImplementation and Testing
- 14 Distributed Parallel Applications: Matrix Multiplication, Interconnection Topologies, Fault Tolerance
- 15 Parallel Languages Overview: Parallel Discrete Event Simulations, Combinatorial Search, GPGPUs
- 16 Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	□	10 MARKS			□	30
8 to 12	□	20 MARKS				□	20
13 to 15	□	10 MARKS				□	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

- | | |
|-----------------|--|
| Textbook | <ul style="list-style-type: none"> Peter Pacheco, Matthew Malensek, An Introduction to Parallel Programming, Elsevier Science, 2021 Subodh Kumar, Introduction to Parallel Programming , 2022 |
| Other | Tim Mattson, Helen He, Alice Koniges, OpenMP Common Core: Making OpenMP Simple Again, (2019)
Using MPI: Portable Parallel Programming with the Message-Passing Interface, 3rd Ed - William Gropp, Ewing Lusk, Anthony Skjellum (2014) |

ECE4307 Cloud Computing**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE4317	4	8	1	0	2	2

COURSE AIM

To get the students familiar with fundamental concepts related to cloud computing technologies. Also, to get familiar with different cloud service and deployment models, different architectural styles and design patterns for cloud computing, the classification of virtualization techniques. The concept of HPC and how it could be used in cloud computing. Understand the use of container service to deploy your application in the cloud and the use of DevOps in configuration management

COURSE WEEKLY CONTENTS

1	Introduction to the course – The enabling technologies of cloud computing, especially virtualization and its different types.
2	Introduction to Cloud Computing (Part 1): history, definition, attributes, advantages, limitation, practical examples.
3	Introduction to Cloud Computing (Part 2): reference model, service models, deployment models, Applications/Data, and lifecycle of a cloud computing application using Amazon Web Services.
4	How to design your Cloud application (Part 1): differences between monolithic and microservices architectures), different architecture styles, and design principles.
5	How to design your Cloud application (Part 2): different pillars of software quality and how to measure them, and diverse types of cloud design patterns (using Microsoft and Amazon Web services as examples).
6	Cloud Infrastructure (Part 1): In-depth of different compute, datastore, and network technologies.
7	7 th week Assessment
8	Cloud Infrastructure (Part 2): In-depth of different compute, datastore, and network technologies.
9	Introduction to Containerization (Part 1): the usage and implementation of containers using dockers.
10	Introduction to Containerization (Part 2): Benefits of Containerization, container orchestration using Kubernetes, and the use of containers in Amazon.
11	Introduction to HPC and cluster computing: the concept, the challenges, design constraints, uses and benefits
12	12 th week Assessment.
13	Configuration Migration and Request for Proposals, Automation, and DevOps
14	Different eras in Cloud Computing: IoT, Fog Computing and Blockchain.
15	Seminars
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	15 Midterm	←	15 MARKS			→	30
			To be freely distributed among possible assessments				
8 to 12	15 Midterm	←	5 MARKS			→	20
			To be freely distributed among possible assessments				
13 to 15	←		10 MARKS			→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Arshdeep Bahga and Vijay Madisetti, “Cloud Computing Solutions Architect: A Hands-On Approach: A Competency-based Textbook for Universities and a Guide for AWS Cloud Certification and Beyond”.
Other	• Rajkumar Buyya, Christian Vecchiola, S. Thaman Selvi, “Mastering Cloud Computing: Foundations and Application Programming”, Morgan Kaufmann, Latest • “Cloud application Architecture Guide”, Microsoft Azure, Microsoft 2021 • Sean Keery, Clive Harber, and Marcus Young, “Implementing Cloud Design Patterns for AWS: Solutions and design ideas for solving system design problems”, 2nd Edition.

ECE4506 Distributed and Parallel Systems**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3503 ECE3301	4	7	2	2	-	3

COURSE AIM

This course studies the fundamental aspects of distributed systems and applications. Early foundations and recent developments in distributed systems will be investigated. Both client-server and peer-to-peer application designs will be discussed. Other topics include sockets, reliability, replication, group membership protocols, clock synchronization, and logical timestamps.

COURSE WEEKLY CONTENTS

1	Introduction to distributed architectures.
2	Types of distributed systems.
3	Distributed System Architecture.
4	Process, Thread and Virtual Machine.
5	Communication and Distributed System.
6	Naming, Identification and address.
7	7th week exam.
8	Introduction to Parallel System.
9	Parallel System Architecture.
10	Parallel System Memory Models.
11	Parallel Program Platform
12	12th week Exam.
13	Distributed Vs. Parallel Systems.
14	Distributed systems Applications (Part 1).
15	Distributed systems Applications (Part 2)
16	Final exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS		→	30
8 to 12	←		20	MARKS		→	20
13 to 15	←		10	MARKS		→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Andrew S. Tanenbaum and Maarten van Steen, "Distributed Systems: Principles and Paradigms", Prentice Hall, latest edition.
Other	

ECE4507 Computer Systems Engineering**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3203 ECE3503	4	7	2	2	-	3

COURSE AIM

This is mostly a hands-on course, which evolves around implementing a series of hardware and software modules to integrate key topics from algorithms, computer architecture, operating systems, compilers, and software engineering, in one unified framework.

COURSE WEEKLY CONTENTS

17	Course Overview and demonstration of some games
18	Boolean arithmetic
19	Sequential Logic
20	Machine Language
21	Computer Architecture
22	Assembler
23	7th week exam
24	Virtual machine I
25	Virtual machine II
26	High Level Language
27	Compiler I
28	12th week
29	Compiler II
30	Operating System I
31	Operating System II
32	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams		Assign.	Quizzes	Reports	Present.	Lab.	Total	
1 to 7	20	Midterm	←	10	MARKS		→	30	
To be freely distributed among possible assessments									
8 to 12	←			20	MARKS		→	20	
13 to 15	←			10	MARKS		→	10	
16 or 17	40	Final							40
Total	Exams		Assign.	Quizzes	Reports	Present.	Lab.	100	

REFERENCES

Textbook	• Nisan and Schocken, "The Elements of Computing Systems", MIT Press, latest edition.
Other	• Selected Papers from Journals and Transactions.

ECE4508 Selected Topics in Computing**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2202 ECE2104	4	8	1	2	-	3

COURSE AIM

To present fundamental algorithms and impossibility results from the concurrent programming literature, and to cover techniques for formally specifying and verifying concurrent systems. Some advanced computing systems will be covered such as embedded systems, reconfigurable computing and real-time systems.

COURSE WEEKLY CONTENTS

1	Concurrent Programming concepts: an overview.
2	Techniques for parallelizing programs.
3	Synchronization algorithms for shared-memory systems.
4	Distributed Programming: an overview.
5	Fault-Tolerance in Distributed Systems.
6	Parallel Programming: Languages, Libraries, and Tools.
7	7th week exam.
8	System Design for maintainability and power efficiency
9	Embedded Systems.
10	Cloud Computing.
11	Polymorphic processors.
12	12th week exam.
13	Real-time Systems
14	Reconfigurable computing.
15	High performance Computing.
16	Final Exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total	
1 to 7	20 Midterm	←	10	MARKS			→	30
To be freely distributed among possible assessments								
8 to 12	←		20	MARKS			→	20
13 to 15	←		10	MARKS			→	10
16 or 17	40 Final							40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.		100

REFERENCES

Textbook	• M. Ben-Ari, "Principles of Concurrent and Distributed Programming: Algorithms And Models", prentice-hall, latest edition. • Scott hauck, André Dehon, "Reconfigurable Computing: The Theory And Practice of FPGA-Based Computation", Morgan Kaufmann, latest edition.
Other	• Christophe Bobda, "Introduction to Reconfigurable Computing Architectures, Algorithms, and Applications", Springer Verlag, latest edition. • Gregory Andrews, "Foundations of Multithreaded, Parallel, and Distributed Programming", Addison-Wesley, latest edition.

ECE4510 High-Performance Computing**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3503	4	7	2	2	0	3

COURSE AIM

The course aims to provide students with a detailed understanding of how computation can seamlessly scale from a single processor to virtually limitless computing power. Develop an understanding of both the computational architecture and the principles of program optimization. Study state-of-the-art processor architectures such as the IBM CELL BE, and Nvidia Tesla GPU. Study parallel algorithm design and programming issues for such systems. Evaluate power, memory, and ILP challenges from the perspectives of the Programming Model, Computational Model, Processor Architecture Model, Threading Model, Memory Model, and Power Model.

COURSE WEEKLY CONTENTS

Introduction: Fundamental concepts in HPC

Parallel Processing Concepts: Levels of parallelism (instruction, transaction, task, thread, memory, function). Models (SIMD, MIMD, SMT, SPMD, Dataflow Models, Demand-driven Computation etc). Architectures: N-wide superscalar architectures, multi-core, multi-threaded

Shared memory programming (openMP)

Distributed memory

Parallel programming with CUDA: Processor Architecture, Interconnect, Communication, Memory Organization, and Programming Models in high performance computing architectures: (Examples: IBM CELL BE, Nvidia Tesla GPU, Intel Larrabee Microarchitecture and Intel Nehalem microarchitecture)

Parallel programming with CUDA: Memory hierarchy and transaction specific memory design and thread Organization

7th week exam.

Fundamental Design Issues in Parallel Computing: Synchronization, Scheduling, Job Allocation and Job Partitioning

Fundamental Design Issues in Parallel Computing: Dependency Analysis, Mapping Parallel Algorithms onto Parallel Architectures and Performance Analysis of Parallel Algorithms

Fundamental Limitations Facing Parallel Computing, Bandwidth Limitations, Latency Limitations, Latency Hiding/Tolerating Techniques and their limitations

Power-Aware Computing and Communication: Power-aware Processing Techniques and Power-aware Memory Design

12th week exam

Power-Aware Computing and Communication: Power-aware Interconnect Design and Software Power Management

Advanced Topics; Petascale Computing and Optics in Parallel Computing Parallel Advanced Topics; Quantum Computers and Recent developments in Nanotechnology and its impact on HPC

Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	□	10 MARKS			□	30
8 to 12	□	20 MARKS				□	20
13 to 15	□	10 MARKS				□	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

- Textbook**
- Sergei Kurgalin and Sergei Borzunov, "A Practical Approach to High-Performance Computing," Springer, 2019
 - Thomas Sterling, Maciej Brodowicz, Matthew Anderson, "High Performance Computing Modern Systems and Practices", Elsevier Science, 2017
- Other**
- Tim Mattson, Helen He, Alice Koniges, OpenMP Common Core: Making OpenMP Simple Again, (2019)
 - Peter Pacheco, Introduction to Parallel Programming, Morgan Kaufmann Publishers, 2011

ECE4609 Selected Topics in Quantum Computing**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
2 EBA3206	4	8	1	2	0	2

COURSE AIM

The aim of this course is to give students a grounding in the fundamentals of quantum computing. As one of the technologies driving the innovation of the future, it is important that students have at least a working background in quantum computing. The course introduces the basic concepts, and then works its way up to designing quantum circuits and algorithms, while taking into account quantum stability and error correction.

COURSE WEEKLY CONTENTS

1	An introduction to the concepts of quantum computing
2	Wave-Particle Duality and Schrodinger's Equation
3	Quantum Super-position and entanglement
4	Qubits
5	Methods of creating Qubits
6	Decoherence and Qubit Stability
7	7th Week Exam
8	Quantum Error Correction
9	Quantum Gates
10	Quantum Circuits
11	Quantum Circuits Continued
12	12th Week Exam
13	Quantum Algorithms
14	Shor's Algorithm and Grover's Algorithm
15	Implications of quantum computing on encryption
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS			30
8 to 12	←		20	MARKS			20
13 to 15	←		10	MARKS			10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Quantum Computation and Quantum Information: 10 th Anniversary Edition, 1 st Edition, Michael A. Nielsen and Isaac L. Chuang
Other	Programming Quantum Computers: Essential Algorithms and Code Samples, Eric R. Johnston, Nic Harrigan, Mercedes Gimeno-Segovia

4.4.5. Concentration 5: Internet of Things (IoT) Engineering

ECE3106 Introduction to Software Engineering

COURSE INFORMATION

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2104 ECE3511	4	7	1	2	0	2

COURSE AIM

In this course, students learn and gain practical experience with software engineering principles, techniques, and tools. Topics in this course include requirements analysis, specification, design, abstraction, programming style, testing, maintenance, communication, teamwork, and software project management. The practical experience centers on a team project, in which a software development project is carried through all the stages of the software life cycle. Particular emphasis is placed on communication skills and on developing maintainable software. In-class activities further provide a hands-on experience in using state-of-the-art techniques and tools.

COURSE WEEKLY CONTENTS

1	Introduction to Software Engineering: different software models.
2	Software Development LifeCycle
3	Requirements Engineering
4	Use cases and Data Modelling
5	Agile and Scrum (Part 1)
6	Agile and Scrum (Part 2)
7	7 th Week Exam
8	Software Architecture: different architecture styles, uses and benefits
9	UML diagrams (Part 1): Sequence Diagram, Class Diagram, Collaboration and Deployment.
10	UML diagrams (Part 2): Activity and State Diagram
11	Software Testing (Part 1): Build Systems and Code Review
12	12 th week Assessment.
13	Software Testing (Part 2): Unit Testing
14	Continuous Integration and Delivery – DevOps; Interviewing and Wrap-up
15	Project Seminars
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	15 Midterm	←	15 MARKS			→	30
			To be freely distributed among possible assessments				
8 to 12	10 Midterm	←	10 MARKS			→	20
			To be freely distributed among possible assessments				
13 to 15	←		10 MARKS			→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

- | | |
|----------|--|
| Textbook | • Ian Sommerville, “Software Engineering”, 10 th Edition, 2015. |
| Other | • John Ousterhout , “A Philosophy of Software Design”, 1 st Edition, 2018 |

ECE3308 Cyber-Physical Systems Design and Analysis**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2202 ECE2104	4	7 or 8	2	2	0	3

COURSE AIM

This course introduces the tools and models that will allow students to develop high confidence in the resulting system's proper operation prior to any operational test. Included are tools for model-based systems engineering, and cyber-physical system verification and validation currently in use by the CPS industry. Numerous examples will be considered, from aerospace, automotive, medical devices, etc. The frequent presence of human operators is also acknowledged and discussed in-depth. Various verification and validation formal methods are described and applied to simple examples.

COURSE WEEKLY CONTENTS

- 1 Introduction to CPS: definitions, trends, Areas of interest, Fundamental approach, examples, Genesis, Modeling, Design, Verification and Validation, Assembly and Deployment
- 2 Review: Calculus, Differential equations, Markov models, Linear systems
- 3 Models: Nature and Computation Myths: Airborne Collision avoidance examples, From Continuous to discrete dynamics, Examples: Water tank, spring oscillator, Modeling trade-offs: Faithfulness, manageability, level-of-detail, accuracy
- 4 Low-level CPS design: Canonical Example: Stopping a car, Feedback, Reduced-gravity Drone, Linear Control, Controllability: Train example
- 5 Mid-level CPS design requirements: Trajectory Planning and examples, Aviation example, Typical requirements
- 6 Mid-level CPS design: Guidance techniques, Classical optimization and examples, Dynamic Programs, Automotive example, Rapidly-exploring Random Trees
- 7 7th week Assessment
- 8 High-level CPS design: High-level decision-making: Linear Temporal Logic (LTL), Self-driving car example, LTL and trajectory planning, Reduced-gravity drone example
- 9 Processors and Sensors: Sensors and CPS – trends, Sensors, CPS, and IoT, Actuators and servos, Embedded CPS architectures, Communications, Processors
- 10 Systems Engineering, general approach: Vee Design cycle and activities, Industry standards and advisory documents, Tools and frameworks: Software-centric viewpoint, Model-based engineering pitfalls, AADL virtual integration – cost savings
- 11 Architecture Analysis and Design Language (AADL): General Principles and positioning, Why modeling in design?, Models, Processes, and tools, AADL introduction, AADL Components: software, hardware, AADL properties
- 12 12th week Assessment
- 13 AADL Crazyflie Case Study: Modeling, Implementing a UAV control logic, AADL functional chain, Flow analysis, Latency / real-time scheduling, AADL, middleware, code generation, Error modeling, and analysis
- 14 Formal methods: Concerns of formal methods, Concerns about formal methods, Abstractions, Abstract interpretation, Model Checking, Hunting for invariants
- 15 Project Submission
- 16 Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	☐	10 MARKS			☐	30
8 to 12	☐	20 MARKS			☐		20
13 to 15	☐	10 MARKS			☐		10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook Pedro H. J. Nardelli, Cyber-physical Systems: Theory, Methodology, and Applications, Wiley-IEEE Press; 1st edition 2022

Other

ECE4307 Cloud Computing**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE4317	4	8	1	0	2	2

COURSE AIM

To get the students familiar with fundamental concepts related to cloud computing technologies. Also, to get familiar with different cloud service and deployment models, different architectural styles and design patterns for cloud computing, the classification of virtualization techniques. The concept of HPC and how it could be used in cloud computing. Understand the use of container service to deploy your application in the cloud and the use of DevOps in configuration management

COURSE WEEKLY CONTENTS

1	Introduction to the course – The enabling technologies of cloud computing, especially virtualization and its different types.
2	Introduction to Cloud Computing (Part 1): history, definition, attributes, advantages, limitation, practical examples.
3	Introduction to Cloud Computing (Part 2): reference model, service models, deployment models, Applications/Data, and lifecycle of a cloud computing application using Amazon Web Services.
4	How to design your Cloud application (Part 1): differences between monolithic and microservices architectures), different architecture styles, and design principles.
5	How to design your Cloud application (Part 2): different pillars of software quality and how to measure them, and diverse types of cloud design patterns (using Microsoft and Amazon Web services as examples).
6	Cloud Infrastructure (Part 1): In-depth of different compute, datastore, and network technologies.
7	7 th week Assessment
8	Cloud Infrastructure (Part 2): In-depth of different compute, datastore, and network technologies.
9	Introduction to Containerization (Part 1): the usage and implementation of containers using dockers.
10	Introduction to Containerization (Part 2): Benefits of Containerization, container orchestration using Kubernetes, and the use of containers in Amazon.
11	Introduction to HPC and cluster computing: the concept, the challenges, design constraints, uses and benefits
12	12 th week Assessment.
13	Configuration Migration and Request for Proposals, Automation, and DevOps
14	Different eras in Cloud Computing: IoT, Fog Computing and Blockchain.
15	Seminars
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	15 Midterm	←	15 MARKS			→	30
			To be freely distributed among possible assessments				
8 to 12	15 Midterm	←	5 MARKS			→	20
			To be freely distributed among possible assessments				
13 to 15	←		10 MARKS			→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Arshdeep Bahga and Vijay Madisetti, “Cloud Computing Solutions Architect: A Hands-On Approach: A Competency-based Textbook for Universities and a Guide for AWS Cloud Certification and Beyond”.
Other	• Rajkumar Buyya, Christian Vecchiola, S. Thamar Selvi, “Mastering Cloud Computing: Foundations and Application Programming”, Morgan Kaufmann, Latest • “Cloud application Architecture Guide”, Microsoft Azure, Microsoft 2021 • Sean Keery, Clive Harber, and Marcus Young, “Implementing Cloud Design Patterns for AWS: Solutions and design ideas for solving system design problems”, 2nd Edition.

ECE4309 IoT Protocols and Technologies for Smart Cities**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3301	4	9	2	0	2	3

COURSE AIM

This course presents the main concepts of the Internet of Things (IoT) ranging from the physical layer and medium access to the applications and standards. The students will be able to understand the fundamental concepts and challenges of IoT systems. They will be able to develop an in-depth knowledge of IoT protocol stack and reference architecture. Moreover, they will understand the main requirements of the physical layer, networking, computing, analytics, and security in IoT systems. Finally, they will be exposed to the main IoT standards and applications.

COURSE WEEKLY CONTENTS

1	Introduction to the course
2	Introduction to Internet of Things – IoT Definitions, Characteristics and Applications
3	IoT Elements.
4	IoT Enabling Technologies
5	IoT Common Standards and Protocols (Part 1): Application Protocols (CoAP, XMPP, MQTT)
6	7 th week Assessment
7	IoT Common Standards and Protocols (Part 2) - Service Discovery Protocols - Infrastructure Protocols (Connectivity technologies)
8	QoS Criteria, IoT Challenges, and Future Directions
9	Big Data Analytics, cloud computing, and fog computing in support of the IoT
10	A Comparative Study of current IoT simulation packages.
11	12 th week Assessment
12	IoT security: Authentication, privacy, encryption
13	IoT Case Studies: Smart home, health care, and smart city
14	Project Submission
15	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	☐	10 MARKS			☐	30
8 to 12	☐	20 MARKS				☐	20
13 to 15	☐	10 MARKS				☐	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	David Hanes et. Al. IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things. Cisco Press, latest edition
Other	Kai Hwang Jack Dongarra Geoffrey Fox, Distributed and Cloud Computing: From Parallel Processing to the Internet of Things, 1st Edition, Morgan Kaufmann, 2012 Ala Al-Fuqaha et al., "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications, IEEE Communication Surveys and tutorials, vol.17, No.4, 2015 Recent Research paper

ECE4310 Mobile and Sensor Computing**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
EEEC3220	4	6	1	2	0	2

COURSE AIM

The course focuses on "Internet of Things" (IoT) systems and technologies, sensing, computing, and communication. Explores fundamental design and implementation issues in the engineering of mobile and sensor computing systems. Topics include battery-free sensors, sensing through walls, robotic sensors, vital sign sensors (breathing, heartbeats, emotions), sensing in cars and autonomous vehicles, sensor security, positioning technologies (including GPS and indoor WiFi), inertial sensing (accelerometers, gyroscopes, inertial measurement units, dead-reckoning), embedded and distributed system architectures, sensing with radio signals, sensing with microphones and cameras, wireless sensor networks, embedded and distributed system architectures, mobile libraries and APIs to sensors, and application case studies. The course includes laboratory assignments and a significant term project.

COURSE WEEKLY CONTENTS

1	Introduction to the course.
	IoT Systems Overview: Sensing Tasks & Modalities, Computation, Power/Energy, Connectivity
2	Fundamentals of IoT Localization.
3	Seeing through Walls & Device-Free Localization.
4	Principles of Wireless Sensing.
5	Network Technologies: fundamentals, applications, and implications.
6	Mesh Networks & Multi-Hop Routing
7	7th week Assessment.
8	Battery-less Connectivity & Smart Cities
9	Introduction to Inertial Sensing & Sensor Fusion
10	The Pothole Patrol: application of inertial sensing.
11	Split Computing / Continuous Object Recognition (Part 1)
12	12th week Assessment.
13	Split Computing / Continuous Object Recognition (Part 2)
14	Mobile health
15	Simulation Project Submission.
16	Final Exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	☐	10 MARKS To be freely distributed among possible assessments			☐	30
8 to 12	☐			20 MARKS		☐	20
13 to 15	☐			10 MARKS		☐	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook

Other

- Manish J. Gajjar, Mobile Sensors and Context-Aware Computing. Morgan Kaufmann, 2017
- Latest Research paper reading list

ECE4404 Big Data Management**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial/Lab	Laboratory	
ECE3511	4	7, 8	1	2	-	2

COURSE AIM

- Helps the students to understand the different research methodologies.
- Introduces the students to the era of Big Data and its main characteristics.
- Describing the reasons behind evolving new Big Data platforms from the perspective of Big Data management systems and analytical tools.
- Getting the students familiar with the recent related technologies and tools used in storage and management of Big Data.
- Identifying the key roles of the new Big Data Ecosystem.
- Study of the MapReduce, Hadoop and Hadoop Ecosystem.
- Provide the students a practitioner's approach to some of the key techniques and tools used in Big Data Analytics.
- Acquiring advanced theoretical knowledge and technical competences about the topics covered in the course as well as some research methods that are relevant for these topics.

COURSE WEEKLY CONTENTS

1	Introduction to Big Data & Data Structures.
2	Big Data Sources, Characteristics.
3	Analytic Life Cycle.
4	Advanced Analytical Theory & Methods: Clustering, Classification.
5	Advanced Analytical Theory & Methods: Association Rules.
6	Advanced Analytical Theory & Methods: Time Series Analysis & Text analysis.
7	7 th Week Exam
8	Advanced Analytics Technology & Tools: MapReduce & Hadoop.
9	Advanced Analytics Technology & Tools: Hadoop Ecosystem.
10	Case Study: Discovery & Data Collection.
11	Case Study: Data Preparation & Model Planning.
12	12 th Week Exam
13	Case Study: Model Building, Communicating Results & Operationalise.
14	Projects' Presentations.
15	Projects' Presentations.
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS		→	30
8 to 12	10 Midterm		10	MARKS		→	20
13 to 15	←		10	MARKS		→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Data Science and Big Data Analytics: Discovering, Analyzing, Visualization and Presenting Data, by EMC Education Services, Wiley publishers, Latest edition.
Other	Hadoop and Big Data: Introduction to Basics of Big data analytics, by Rajesh Pasupuleti, August 2016, Kindle Edition. The Impact of Big Data on The Retail Sector: Examples and Use-Cases, by Mona Lebed, Apr 5th 2018. Big Data: Using SMART Big Data, Analytics and Metrics To Make Better Decisions, by Beranrd Marr, Wiley Publishers, 2015.

ECE4407 Data Analytics for the Internet of Things**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE4309	4	8	2	2	0	3

COURSE AIM

This course aims to teach students the task of extracting value from huge amounts of barely intelligible data. The data takes a convoluted route just to be on the servers for analysis, but insights can emerge through visualization and statistical modeling techniques. Students will learn to develop data science solutions for big data generated from IoT systems. Moreover, they will get to know strategies to collect and store the data to optimize the potential for analytics, and strategies to handle data quality concerns. Cloud resources are a great match for IoT analytics, so Amazon Web Services and Microsoft Azure will be reviewed in detail next. Geospatial analytics is then introduced to leverage location information. Combining IoT data with environmental data is also discussed to enhance predictive capability.

COURSE WEEKLY CONTENTS

16	Introduction to the course
17	IoT Analytics: Definition and Challenges
18	IoT Devices and Networking Protocols
19	IoT Analytics for The Cloud
20	Creating an AWS Cloud Analytics Environment
21	Data Collection: Strategies and Techniques
22	7 th week Assessment
23	Exploring IoT Data
24	Designing visual analysis for IoT data
25	Strategies to Organize Data for Analytics
26	Data Science for IoT Analytics
27	12 th week Assessment
28	Applying Geospatial Analytics to IoT Data
29	Project Submission
30	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	☐	10 MARKS To be freely distributed among possible assessments			☐	30
8 to 12	☐		20 MARKS			☐	20
13 to 15	☐		10 MARKS			☐	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Andrew Minter, Analytics for the Internet of Things (IoT): Intelligent analytics for your intelligent devices, Packt Publishing, latest edition
Other	

ECE4506 Distributed and Parallel Systems**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3503 ECE3301	4	7	2	2	-	3

COURSE AIM

This course studies the fundamental aspects of distributed systems and applications. Early foundations and recent developments in distributed systems will be investigated. Both client-server and peer-to-peer application designs will be discussed. Other topics include sockets, reliability, replication, group membership protocols, clock synchronization, and logical timestamps.

COURSE WEEKLY CONTENTS

33	Introduction to distributed architectures.
34	Types of distributed systems.
35	Distributed System Architecture.
36	Process, Thread and Virtual Machine.
37	Communication and Distributed System.
38	Naming, Identification and address.
39	7th week exam.
40	Introduction to Parallel System.
41	Parallel System Architecture.
42	Parallel System Memory Models.
43	Parallel Program Platform
44	12th week Exam.
45	Distributed Vs. Parallel Systems.
46	Distributed systems Applications (Part 1).
47	Distributed systems Applications (Part 2)
48	Final exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS		→	30
8 to 12	←		20	MARKS		→	20
13 to 15	←		10	MARKS		→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Andrew S. Tanenbaum and Maarten van Steen, "Distributed Systems: Principles and Paradigms", Prentice Hall, latest edition.
Other	

ECE4610 Tiny Machine Learning Principles**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2104 EBA3206	4	7	2	0	2	3

COURSE AIM

This course will introduce efficient AI computing techniques that enable powerful deep learning applications on resource-constrained devices. Topics include model compression, pruning, quantization, neural architecture search, distributed training, data/model parallelism, gradient compression, and on-device fine-tuning. It also introduces application-specific acceleration techniques for large language models, diffusion models, video recognition, and point cloud. This course will also cover topics about quantum machine learning. Students will get hands-on experience deploying large language models (e.g., LLaMA 2) on a laptop. Tiny Machine Learning (TinyML) is an introductory course at the intersection of Machine Learning and Embedded IoT Devices.

COURSE WEEKLY CONTENTS

Basics of Deep Learning
 Pruning and Sparsity
 Quantization
 Neural Architecture Search
 Knowledge Distillation
 MCUNet: Tiny Deep Learning on IOT Devices
 7th Week Exam
 TinyEngine
 Transformer and LLM
 Vision Transformer
 Point Cloud
 12th Week
 Distributed Training
 On-Device Training
 Quantum Machine Learning
 Project Presentations

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	□	10 MARKS To be freely distributed among possible assessments			□	30
8 to 12	□			20 MARKS		□	20
13 to 15	□			10 MARKS		□	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	TinyML: Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers
Other	<i>TinyML Cookbook: Combine Artificial Intelligence and Ultra-low-power Embedded Devices to Make the World Smarter</i> , Gian Marco Iodice, Ronan Naughton <i>Hands-on TinyML: Harness the power of Machine Learning on the edge devices (English Edition)</i> , Rohan Banerjee

ECE4611 Embedded Artificial Intelligence Applications**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3601	4	7	1		2	2

COURSE AIM

In this course, students will learn more about embedded machine learning. Embedded systems have enabled microcontrollers to run complicated machine-learning models. Embedded devices for machine learning applications can fulfil many tasks in the industry. This course the knowledge and skills required to take advantage of this next major shift in technologies and the related growth in job demand. The course explores the specialized tools, frameworks, technologies, platforms, and methods to create exciting new embedded AI devices. Study TinyML – the field of applying ML technologies to embed AI in resource-constrained devices. Discover how complex embedded AI applications work on smartphones, drones, and other devices with constrained processing, memory, power, and other resources.

COURSE WEEKLY CONTENTS

49	Embedded AI Definition, Advantages, Challenges, and Applications
50	Fundamentals of Embedded Systems and AI/ML
51	Hardware AI/ML Accelerators
52	ML Sensors
53	Federated Learning
54	Mobile AI
55	7 th Week assessment
56	Flower: A Friendly Federated Learning Framework
57	Acoustic AI
58	Intro. to TinyML
59	Robotic AI: From Narrow Robots to General Robots
60	Physical Knowledge-Informed Learning Adaptation for Embedded AI
61	12 th Week assessment
62	Embedded Data
63	Concepts of Tensor Virtual Machine (TVM)
64	Project Presentations

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS		→	30
8 to 12	←		20	MARKS		→	20
13 to 15	←		10	MARKS		→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	<i>TinyML: Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers, (1st Edition), Pete Warden, Daniel Situmayake</i> <i>Embedded Artificial Intelligence: Devices, Embedded Systems, and Industrial Applications (1st Edition), Ovidiu Vermesan, Mario Diaz Nava, Björn Debaillie</i>
Other	

4.4.6. Advanced Undergraduate Elective Courses

ECE3302 Threat Modelling

COURSE INFORMATION

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3301	3	5	2	2	1	3

COURSE AIM

This course aims to equip students with the skills to identify, assess, and mitigate cybersecurity threats using comprehensive threat modelling techniques. By exploring methodologies like STRIDE, DREAD, PASTA, and VAST, students will learn to analyse potential threats and vulnerabilities systematically. Combining theory with hands-on practice, the course prepares students to apply threat modelling to real-world scenarios and emerging challenges, enhancing their ability to strengthen security measures in various technological environments.

COURSE WEEKLY CONTENTS

1	Introduction to threat modelling and cybersecurity
2	Identifying threats and vulnerabilities
3	Risk assessment and management
4	Mitigation strategies
5	Attack trees and attack graphs
6	Data flow diagrams and trust boundaries
7	7th Week Exam
8	STRIDE and DREAD
9	PASTA and VAST
10	Trike and IriusRisk
11	Case studies.
12	12th Week Exam
13	Emerging trends and challenges
14	Hands-on practice.
15	Project Discussion
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10 MARKS	To be freely distributed among possible assessments			30
8 to 12	←		20 MARKS			→	20
13 to 15	←		10 MARKS			→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Shostack, A. (2014). Threat Modeling: Designing for Security. Wiley.
Other	Tarandach, I., & Coles, M. J. (2020). Threat Modeling: A Practical Guide for Development Teams. O'Reilly Media.

ECE3403 Data Analytics and Optimization using Python**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2401 ECE2104	3	5	2	2	0	3

COURSE AIM

This course presents fundamentals of optimization and data analytics to students with limited knowledge of the topic. The students shall learn how to formulate an optimization problem, apply different optimization methods to solve real life problems. Implementations of different optimization techniques will be developed using Python. In addition, students will be able to describe, analyze data, and use advanced statistical tools to make decisions. Python data analytics toolkits will be used to provide hands on experience to the students. Topics include deterministic and stochastic optimization methods and data analysis techniques. Examples of optimization approaches include probabilistic and evolutionary approaches. The course offers a range of theoretical and practical topics to help the students handle emergent practical problems.

COURSE WEEKLY CONTENTS

1	Introduction to Optimization and Data Analytics Concepts and Python Tools
2	Optimization Problem Formulation and Overview on related Python Toolkits
3	Deterministic vs Stochastic Optimization.
4	Constrained vs Unconstrained Optimization
5	Linear and Non-Linear Programming.
6	Mixed Integer Linear and Non-Linear Programming)
7	7th Week Exam
8	Nature Inspired Optimization
9	Graph-based Optimization
10	Other Stochastic Optimization methods :Harmony Search
11	Data Preprocessing and Cleaning
12	12 th week Assessment
13	Descriptive and Inferential Statistics
14	Data Visualization
15	Case studies using Python
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS		→	30
8 to 12	←		20	MARKS		→	20
13 to 15	←		10	MARKS		→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Singiresu S. Rao , "Engineering Optimization : Theory and Practice", JOHN WILEY & SONS, INC, Fourth Edition.
Other	• <i>Python for Data Analysis</i>, by Wes McKinney, O'Reilly. • <i>Python in a Nutshell</i>, by Alex Martelli, O'Reilly. • <i>Data Mining and Business Analytics with R</i>, by Johannes Ledolter; Publisher: Wiley (2013), ISBN-13: 978-1118447147; • <i>Elements of Statistical Learning: Data Mining, Inference, and Prediction</i>, by Trevor Hastie, Robert Tibshirani and Jerome Friedman G R Sinha, "Modern Optimization Methods for Science, Engineering and Technology", IOP Science, latest edition.

ECE3405 Modelling and Simulation**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2104 EBA3206	3	6	1	2	-	2

COURSE AIM

To understand the models as tools to make predictions about the behavior of systems. Acquiring the necessary conceptual and practical background for representing discrete and continuous random processes.

COURSE WEEKLY CONTENTS

1	Introduction.
2	Review of probability, random variables and distributions
3	The exponential and Poisson distributions.
4	Generation of random numbers and distributions.
5	Operational laws. Little law and its applications.
6	Random processes, state transition diagrams.
7	7th week exam.
8	Discrete and Continuous Markov processes, steady state solutions.
9	Continuous Markov processes (continue).
10	Queueing models – performance measures.
11	Applications.
12	12th week exam.
13	Modeling with stochastic Petri nets.
14	Modeling with stochastic Petri nets (continue).
15	Features of Simulation languages.
16	Final Exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10 MARKS	To be freely distributed among possible assessments			30
8 to 12	←		20 MARKS			→	20
13 to 15	←		10 MARKS			→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Averill M. Low & W. David Kelton, “Simulation Modeling and Analysis”, Mc Graw -Hill, latest edition.
Other	• Barry L. Nelson, “Stochastic Modeling: Analysis and Simulation”, Mc Graw - Hill, latest edition.

ECE4106 Assembly Language**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3204	4	7 or 8	1	2	0	2

COURSE AIM

This course introduces the students with the organization of computer systems and assembly language, including what's needed to write, link and execute a program written in assembly language. It focuses on the major components of a microprocessor and its function. Assembly language using Turbo Assembler and interfacing assembly language program will also be included. Students are expected to design a system applying assembly language. The course includes a project that teaches students how to deal with various topics in the course.

COURSE WEEKLY CONTENTS

1	Introduction to assembly language and machine language.
2	Addressing Modes; x86 Architecture: Registers.

3	Memory Model; Instruction Set.
4	Arithmetic skills & project assignment.
5	Processor-I/O Interface; Interrupts and DMA
6	Stack operations and parameter passing.
7	7th week exam.
8	Arrays and string operations.
9	Dynamic memory allocation and pointers.
10	High-Level Language Interface.
11	Subroutines and function calls.
12	12th Week Exam.
13	Real and Protected mode Interrupt structure.
14	Disk and file handling.
15	Peripheral device drivers.
16	Final Exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10 MARKS			→	30
8 to 12	10 Midterm		10 MARKS			→	20
13 to 15	←		10 MARKS			→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Introduction to Assembly Language Programming: For Pentium and RISC Processors, by Sivarama P. Dandamudi, 2nd Edition, Springer.
Other	• An Assembly Language Introduction to Computer Architecture: Using the Intel Pentium, by Karen Miller and Jim Goodman, Oxford University Press, latest edition. • Barry B. Brey, "Intel Microprocessors", Prentice Hall, latest edition.

ECE4108 Advanced Data Structures in Applications

COURSE INFORMATION

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2103	4	7 or 8	1	2	0	2

COURSE AIM

- Understand a variety of popular data structures and recognize how to implement and use them to solve useful problems
- Explore different ideas that would be beneficial for revealing application contexts in which they can be deployed.
- Acquire practical programming experience for applications relevant to data mining, networks and big data.

COURSE WEEKLY CONTENTS

1	Introduction to the course
2	User defined data structures and its applications
3	Priority queues
4	Quad- and Oct-trees
5	Splay trees and B trees

B.Sc. in Computer Engineering

6	Suffix Trees and Suffix Arrays
7	7 th week Assessment
8	KD Trees and R-Trees
9	Image data structures
10	Data Structures in Data Mining
11	Data structures in Network Applications
12	12 th week Assessment
13	Data Structures in Big Data Stores
14	Project submission
15	Revision
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	5	5				30
8 to 12	15 EXAM		5				20
13 to 15		5			5		10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Dinesh P. Mehta and Sartaj Sahni "Handbook of Data Structures and Applications", 2nd ed. Chapman & Hall/CRC.
Other	• Suman Saha, Shailendra Shukla " Advanced Data Structures Theory and Applications ", Chapman and Hall/CRC, 2020

ECE4207 Micro-Computer Based Design**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3204	4	7 or 8	1	2	0	2

COURSE AIM

Understand the complete design cycle and different platforms for micro based systems. Study interfacing standards and hardware/ software portioning.

COURSE WEEKLY CONTENTS

1	Introduction to Micro based systems.
2	Embedded systems design cycle, Memory and Operating System for embedded systems. Lab: Programming micro controllers using assembly language.
3	Embedded system design considerations and Challenges. Lab: Programming microcontrollers using C – part 1.
4	Programmable devices and FPGAs. Lab: Interfacing with 7-segment display.
5	Microprocessors Vs. Microcontrollers. Lab: Interfacing with keypad.
6	I/O interface ISA bus. Lab: Serial port communication
7	7th week exam.
8	I/O interface cont. Lab: Interfacing with ADC and DAC.
9	Interface standards for external devices. Lab: Motor control
10	Ethernet Network interface. Lab: Sensor interfacing
11	12th Week Exam.
12	Real Time operating System. Lab: Design of final project.
13	Real Time operating systems cont. Lab: CAD Design tools.
14	Presentation of project.
15	Final Exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS		→	30
8 to 12	←		20	MARKS		→	20
13 to 15	←		10	MARKS		→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Mohamed Rafiquzzaman, “The AVR microcontrollers and embedded systems”, Pearson
Other	• Mohamed Rafiquzzaman, “Microprocessors and Microcomputer Based System design”, CRC Pub., latest edition.

ECE4211 Computer Aided Design**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3203	4	7 or 8	2	2	1	3

COURSE AIM

To learn new design technologies, large-scale designs using VLSI technology and modern computer techniques used in Digital circuit designs & implementation.

COURSE WEEKLY CONTENTS

1	Introduction to CMOS Circuits.
2	Circuit & system Representation.
3	Circuit Characterization.
4	Circuit performance estimation.
5	Interconnect and Wiring.
6	Combinational Circuit Design.
7	7th week Exam.
8	Sequential Circuit Design.
9	Design methodology and Tools.
10	Datapath subsystems – part 1.
11	Datapath subsystems – part 2.
12	12th week Exam.
13	Design tools I.
14	Design tools II.
15	Revision.
16	Final exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS			30
8 to 12	←		20	MARKS			20
13 to 15	←		10	MARKS			10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	• Neil H. E. Weste and Kamran Eshraghian, “Principles of CMOS VLSI Design, a Systems Perspective”, Addison – Wesley, latest edition.
Other	• Steven M. Rubin, “Computer Aids for VLSI Design”, Addison –Wesley, latest edition. • D. Pucknell and K. Eshraghian, “Basic VLSI Design, Systems and Circuits”, Prentice Hall, latest edition.

ECE4304 Data Security**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE2104 EBA3206	3	6	1	2	-	2

COURSE AIM

The aim of the course is to teach the principles and practice of communication security. The course emphasis on the underlying principles and techniques of network security including policies, cryptanalysis, symmetric and asymmetric cryptography, and authentication protocols. Students successfully completed this class will be able to explain the basic number theory required for cryptographic applications as well as various cryptographic systems. At the end of the course, a student will have an understanding of the challenges in network security and to what extent these challenges can be addressed with today's network technologies.

COURSE WEEKLY CONTENTS

1	Goals of data & information security and Threats -Types of Attacks.
2	Classical Encryption Techniques.
3	Block Ciphers and the Data Encryption Standard (DES).
4	Advanced Encryption Standard (AES).
5	Pseudo Random Number Generation and Stream Ciphers
6	Hashing Algorithms.
7	7th week exam.
8	Message Digests and Message Authentication Code.
9	Basic Concepts in Number Theory and Finite Fields.
10	Public key Cryptography and RSA.
11	Digital Signatures & Authentication Protocols.
12	12th week exam.
13	Key Management and Distribution.
14	Securing Traffic: Email and Web Security.
15	Firewalls and Intrusion Detection Systems.
16	Revision and start of final exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total	
1 to 7	20 Midterm	←	10	MARKS			→	30
To be freely distributed among possible assessments								
8 to 12	10 Midterm		10	MARKS			→	20
13 to 15	←		10	MARKS			→	10
16 or 17	40 Final							40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.		100

REFERENCES

Textbook	• Cryptography and Network Security Principles and Practices, by William Stallings 7th Edition.
Other	1- Applied Cryptography, by Bruce Schneier, John Wiley & Sons, 2nd Edition. 2- Cryptography and Network Security, by Atul Kahate, Tata McGraw-Hill Education, 2003.

ECE4305 Internetwork Security**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3301	4	7	1	2	-	2

COURSE AIM

Develops a basic understanding of the techniques used to protect users online and addresses some of the design choices behind these techniques. The course emphasizes to give a basic understanding of previous attacks on cryptosystems with the aim of preventing future attacks. The course also provide the foundation and theoretical underpinning which aims to give an understanding of the way in which IT systems can be attacked and penetrated by circumventing security or exploiting vulnerabilities in the system.

COURSE WEEKLY CONTENTS

1	Security definition, Attacks, Risk, Threats and Vulnerabilities.
2	Footprinting and Reconnaissance.
3	Scanning Networks.
4	Vulnerability Analysis.
5	System Hacking
6	Evading IDS, Firewalls, and Honeypots
7	7th week exam.
8	Malware Threats.
9	Sniffing.
10	Session Hijacking.
11	Hacking Web Servers & Web Application.
12	12th week exam.
13	SQL Injection.
14	Hacking Wireless Networks and Mobile Platforms.
15	IoT Hacking and Cloud Computing.
16	Final Exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS		→	30
To be freely distributed among possible assessments							
8 to 12	10 Midterm		10	MARKS		→	20
13 to 15	←		10	MARKS		→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Linda volonino, Reynaldo anzalduwa Computer Forensics: principles and practices, Prentice Hall, Latest edition.
Other	- Aaron Philipp, David Cowen, and Chris Davis, "Hacking Exposed Computer Forensics: Computer Forensics Secrets and Solutions", McGraw-Hill Osborne Media, latest edition. - Michael Solomon, Neil Broom, Diane Barrett, "Computer Forensics JumpStart", Wiley, latest edition.

ECE4306 Selected Topics in Networks**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE4317	4	8	2	2	0	3

COURSE AIM

The Internet of Things is transforming our physical world into a complex and dynamic system of connected devices on an unprecedented scale. Advances in technology are making possible a more widespread adoption of IoT. This course introduces the students to the Internet of Things broad field. Topics covered includes: IoT definitions, Characteristics and Applications, IoT Elements, IoT Enabling Technologies, IoT Common Standards and Protocols including Application protocols (CoAP, XMPP, MQTT), Service Discovery Protocols, Infrastructure Protocols. In addition, the course explores the role of Big Data Analytics, cloud computing and fog computing in support of the IoT. Moreover, IoT simulators are reviewed. Finally, several IoT case studies are discussed. The students are required to build IoT systems as a course project.

COURSE WEEKLY CONTENTS

1	Introduction to the course
2	Introduction to Internet of Things – IoT Definitions, Characteristics and Applications
3	IoT Elements.
4	IoT Enabling Technologies
5	IoT Common Standards and Protocols (Part 1): Application Protocols (CoAP, XMPP, MQTT)
6	7 th week Assessment
7	IoT Common Standards and Protocols (Part 2) - Service Discovery Protocols - Infrastructure Protocols (Connectivity technologies)
8	QoS Criteria, IoT Challenges and Future Directions
9	Big Data Analytics, cloud computing and fog computing in support of the IoT
10	A Comparative Study of current IoT simulation packages.
11	12 th week Assessment
12	Cloud Computing: Definition and foundation, Enabling Technologies, Service Models, Deployment models
13	IoT Case Studies
14	Project Submission
15	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS			30
8 to 12	←		20	MARKS			20
13 to 15	←		10	MARKS			10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Kai Hwang Jack Dongarra Geoffrey Fox, Distributed and Cloud Computing: From Parallel Processing to the Internet of Things, 1st Edition, Morgan Kaufmann, 2012
Other	- Ala Al-Fuqaha et al., "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications, IEEE Communication Surveys and tutorials, vol.17, No.4, 2015 - Recent Research paper

ECE4311 Digital Forensics**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE4304	4	7	2	2	1	3

COURSE AIM

This course provides an in-depth introduction to digital forensics, covering the principles, techniques, and tools used to investigate and analyse digital devices. Students will gain practical skills in data recovery, evidence preservation, and legal considerations, preparing them for careers in digital forensics and cybersecurity.

COURSE WEEKLY CONTENTS

1	Introduction to Digital Forensics
2	Legal and Ethical Considerations
3	Fundamentals of Data Storage
4	Data Acquisition and Preservation
5	Data Recovery Techniques
6	Analysis of Digital Evidence
7	7th Week Exam
8	Network Forensics
9	Mobile Device Forensics
10	Cloud and IoT Forensics
11	Incident Response and Handling.
12	12th Week Exam
13	Emerging Trends and Challenges
14	Hands-on practice.
15	Project Discussion
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10 MARKS			→	30
			To be freely distributed among possible assessments				
8 to 12	←		20 MARKS			→	20
13 to 15	←		10 MARKS			→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Nelson, B., Phillips, A., & Steuart, C. (2025). Guide to Computer Forensics and Investigations. Cengage Learning.
Other	Johansen, G. (2020). Digital Forensics and Incident Response: Incident response techniques and procedures to respond to modern cyber threats. Packt Publishing

ECE4313 Introduction to Blockchain Technologies and Applications

COURSE INFORMATION

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE4304	4	7	2	2	1	3

COURSE AIM

This course provides an in-depth introduction to blockchain technologies, exploring the principles, architectures, and applications of blockchain systems. Students will gain practical skills in blockchain development and understand the potential impact of blockchain on various industries.

COURSE WEEKLY CONTENTS

1	Introduction to Blockchain Technology
2	Cryptographic Foundations
3	Decentralized Networks and Consensus Mechanisms
4	Blockchain Architecture and Components
5	Smart Contracts
6	Blockchain Platforms and Ecosystems
7	7th Week Exam
8	Blockchain Applications in Finance
9	Blockchain in Supply Chain Management
10	Blockchain and IoT
11	Legal and Regulatory Considerations.
12	12th Week Exam
13	Emerging Trends and Challenges
14	Hands-on practice.
15	Project Discussion
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20	Midterm	← 10 MARKS → To be freely distributed among possible assessments				30
8 to 12	←	20 MARKS				→	20
13 to 15	←	10 MARKS				→	10
16 or 17	40	Final					40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Bashir, I. (2020). Mastering Blockchain: Unlocking the Power of Cryptocurrencies, Smart Contracts, and Decentralized Applications (3rd ed.). Packt Publishing.
Other	Drescher, D. (2017). Blockchain Basics: A Non-Technical Introduction in 25 Steps. Apress.

ECE4318 White Hat Hacking**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE4304 ECE4317	4	8	1		2	2

COURSE AIM

This course provides a comprehensive introduction to the principles and practices of white hat hacking, also known as ethical hacking. Students will learn to assess and improve security systems by legally exploiting vulnerabilities and testing these systems against potential security breaches.

COURSE WEEKLY CONTENTS

1	Introduction to White Hat Hacking
2	Setting Up the Hacking Lab (Tools and environment)
3	Network and System Scanning Techniques
4	Enumeration (using practical tools like Wireshark)
5	System Hacking
6	Malware Threats
7	7th Week Exam
8	Sniffing and Spoofing
9	Social Engineering
10	Denial of Service Attacks
11	Web Application Hacking.
12	12th Week Exam
13	Wireless Network Hacking
14	Cryptography.
15	Incident Response and Forensics
16	Final Exam

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS		→	30
8 to 12	←		20	MARKS		→	20
13 to 15	←		10	MARKS		→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

Textbook	Erickson, J. (2008). Hacking: The Art of Exploitation (2nd ed.). No Starch Press.
Other	Weidman, G. (2014). Penetration Testing: A Hands-On Introduction to Hacking. No Starch Press.

ECE4509 Advanced Database Systems**COURSE INFORMATION**

Prerequisites	Academic Year & Level		Teaching Methods			Credit Hrs.
	Year	Semester	Lecture	Tutorial	Laboratory	
ECE3511	4	7 or 8	1	2		2

COURSE AIM

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

Understand physical database design concepts.

Tune the performance of existing Database systems by changing the physical design

Understand different Database security protocols.
 Understand query processing and optimization techniques for relational DBMS.
 Study Transaction Management and Concurrency control and Recovery techniques
 Study basics of Distributed Database System Design and Management
 Get familiar with current DBMS applications and current models to support them like NOSQL Data bases

COURSE WEEKLY CONTENTS

1. Review on Database concepts.
2. Review of conceptual and Logical Database system design
3. Record Storage and Primary File Organizations
4. Index Structures for Files
5. Physical Database Design and Tuning
6. Database System Architectures and the System Catalog
7. 7th week exam
8. Query Processing and Optimization
9. Transaction Processing Concepts
10. Concurrency Control Techniques
11. Database Recovery Techniques
12. 12th week exam
13. Database Security and Authorization
14. Distributed Database Concepts
15. Models for current applications
16. Final Exam.

STUDENT GRADING & ASSESSMENT

Weeks	Exams	Assign.	Quizzes	Reports	Present.	Lab.	Total
1 to 7	20 Midterm	←	10	MARKS		→	30
8 to 12	←		20	MARKS		→	20
13 to 15	←		10	MARKS		→	10
16 or 17	40 Final						40
Total	Exams	Assign.	Quizzes	Reports	Present.	Lab.	100

REFERENCES

- Textbook** • Ramez Elmasry & S. Navathe, "Database Management Systems", Pearson Publishers, latest edition.
- Other** Ramakrishnan. Gehrke, "Database Management systems", Mc-Graw Hill, latest edition.