Faculty of Computing Science and Engineering
Department of Software Engineering

Course Contents

PART I: HARMATTAN SEMESTER

Grouping	Course Code	Course Title	Prerequisites Corequisites	L-T-P	Units
a) Core-compulsory courses	CSC 101	Introduction to Computer Science		2-0-3	3
	PHY 101	General Physics I		3-0-0	3
	PHY 107	Experimental Physics IA		0-0-3	1
	CHM 101	Introductory Chemistry I		3-1-0	4
	CHM 103	Practical Chemistry I		0-0-3	1
	MTH 101	Elementary Mathematics I		4-1-0	5
b) General Studies	TPD 101	Engineers in Society		1-0-0	1
	SE	Special Elective		2-0-0	2
	SE	Special Elective		2-0-0	2
				17-2-9	**22**

PART I: RAIN SEMESTER

Grouping	Course Code	Course Title	Prerequisites Corequisites	L-T-P	Units

a) Core-compulsory courses	MTH 102	Elementary Mathematics II		4-1-0	5
	MTH 104	Vectors		2-0-0	2
	PHY 102	General Physics II`		3-0-0	3
	PHY 108	Experimental Physics IB		0-0-3	1
	CHM 102	Introductory Chemistry II		3-1-0	4
	CHM 104	Practical Chemistry II		0-0-3	1
	CSE 102	Introduction to Software Engineering		2-0-3	3
b) General Studies	SE	Special Elective		2-0-0	2
				16-2-6	**21**

PART II: HARMATTAN SEMESTER

Grouping	**Course Code**	**Course Title**	**Prerequisites Corequisites**	**L-T-P**	**Units**
a) Core-compulsory courses	CSC 201	Computer Programming, I		1-1-3	3
	CPE 203	Intro. to Digital System I		2-0-3	3
	MTH 201	Mathematical Methods I	MTH 102	3-1-0	4
	MEE 205	Engineering Mechanics I		2-1-0	3
	CSE 201	Elements of Software Construction I		1-0-3	2
	CHE 201	Introduction to Thermodynamics		2-1-0	3
	EEE 201	Applied Electricity I	PHY 102	2-0-0	2
	EEE 291	Applied Electricity Lab. I		0-0-3	1
	SE	Special Elective		2-0-0	2

b) General Studies				**15-4-12**	**23**

PART II: RAIN SEMESTER

Grouping	Course Code	Course Title	Prerequisites Corequisites	L-T-P	Units
a) Core-compulsory courses	CSC 202	Computer Programming II	CSC 201	0-0-6	2
	CPE 204	Intro. to Digital System II		1-0-3	2
	CPE206	Introd. Computer Network		2-0-0	2
	MTH 202	Mathematical Methods II	MTH 201	3-1-0	4
	CSE 202	Elements of Software Construction II	CSE 201	1-0-3	2
	MEE 206	Engineering Mechanics II	MEE 205	2-0-0	2
	AGE 202	Workshop Practice		1-0-3	2
	EEE 202	Applied Electricity II	EEE 201	2-0-0	2
	EEE 292	Applied Electricity Lab. II	EEE 201	0-0-3	1
b) General Studies	SE	Special Elective		2-0-0	2
				14-1-18	**21**

LONG VACATION

Grouping	Course Code	Course Title	Prerequisites Corequisites	L-T-P	Units
Core Course	CSE 200	SWEP		0-0-9	3

PART III: HARMATTAN SEMESTER

Grouping	Course Code	Course Title	Prerequisites Corequisites	L-T-P	Units

a) Core-compulsory courses	CPE 301	Low Level Languages	CSC 201	2-0-3	3
	CSC 305	Introduction Database Systems		2-0-3	3
	CSC311	Information Systems		2-0-0	2
	CSE 321	Fundamentals of Requirements Engineering		2-0-0	2
	CSE 315	Object-Oriented Programming	CSC 201	2-0-3	3
	CSC317	Automata Theory and Computability		2-0-0	2
	CSC 307	Numerical Computation I	MTH 201	2-0-3	3
	CHE 305	Engineering Analysis I	MTH 202	2-1-0	3
b) General Studies	SE	Special Elective		2-0-0	2
				16-1-9	**23**

PART III: RAIN SEMESTER

Grouping	**Course Code**	**Course Title**	**Prerequisites Corequisites**	**L-T-P**	**Units**
a) Core-compulsory courses	CSC 302	Data Structure and Analysis of Algorithms	CSC 202	2-0-3	3
	CSC 308	Numerical Computation II	MTH 201	2-0-3	3
	CSE302	Object Oriented Programming		2-0-3	3
	CSE 312	Systems Analysis and Design		1-0-3	2
	CSE 322	Software Design	CSC 315	2-1-0	3
	CSE 324	Software Testing		1-0-3	2
	CPE 316	Artificial Intelligence		2-0-0	2

	AGE 302	Statistics for Engineers		2-0-0	2
	CHE 306	Engineering Analysis II	CHE 305	2-1-0	3
				14-2-12	**23**

LONG VACATION

Grouping	**Course Code**	**Course Title**	**Prerequisites Corequisites**	**L-T-P**	**Units**
Core Course	CSE 300	Student Industrial Work Experience Scheme I		0-0-9	3

PART IV: HARMATTAN SEMESTER

Grouping	**Course Code**	**Course Title**	**Prerequisites Corequisites**	**L-T-P**	**Units**
a) Core-compulsory courses	CSE 421	Software Systems Architecture		2-0-0	2
	CPE 401	Computer Architecture and Organisation	CPE 301	3-0-0	3
	CSC 403	Compiling Techniques	CSC 302, CSC 317	2-1-0	3
	CPE 405	Microprocessor Technology and Microprogramming	CPE 204	3-0-0	3
	CSC 415	Operating Systems Principles		2-1-0	3
	CSE407	Techniques in Software Engineering		1-0-3	2
	CSE 425	Software Engineering for Web Applications		2-0-3	3

	CPE 413	Data Communication and Networking		1-0-3	2
	CVE 401	Technical Report Writing		2-0-3	3
				18-2-12	**24**

PART IV: RAIN SEMSTER AND LONG VACATION

Grouping	**Course Code**	**Course Title**	**Prerequisites Corequisites**	**L-T-P**	**Units**
Core Course	CSE 400	Student Industrial Work Experience Scheme II	Not more than 12 Units outstanding	0-0-27	9

PART V: HARMATTAN SEMESTER

Grouping	**Course Code**	**Course Title**	**Prerequisites Corequisites**	**L-T-P**	**Units**
a) Core-compulsory courses Software Engineering &	CSC 505	Operations Research		2-0-0	2
	CSC 501	Ethics and Rights Issues in Computing		2-0-0	2
	CSE 503	Individual Project I		0-0-9	3
	CSE 521	Aspect Oriented Development		1-0-3	2
	CSE 523	Principles of Human Computer Interaction		2-0-3	3
	CSE 519	Formal Methods and Program Verification	CSE 425	1-0-3	2
	CSE 525	Software Maintenance		2-0-3	3

Information systems /Database	TPD 501	Industrial and Engineering Economics		2-0-0	2
	Restricted Electives - plus 2 courses from any of these electives				
	CSC507	Intelligent System Design		1-0-3	2
	CSC523 CSC515 CSC 527	Techniques in Data Analysis Data warehousing Software Economics		1-0-3 1-0-3 1-1-0	2 2 2
				14-1-27	**23**

PART V: RAIN SEMESTER

Grouping	**Course Code**	**Course Title**	**Prerequisites Corequisites**	**L-T-P**	**Units**
a) Core-compulsory courses	CSE 504	Individual Project II	CSE 503	0-0-9	3
	CPE 506	Industrial Application Studies & Innovations		2-1-0	3
	CPE 502	Hardware System Studies	CPE 301	1-1-0	2
	CSE 526	Software Engineering Process and Configuration Management		2-1-0	3
	CSC508	Computer System Project Management		2-1-0	3
	CSE 514	Software Quality Management		1-0-3	2
	TPD 502	Technology Policy		2-0-0	2
	Restricted Electives - plus 2 courses from any of these electives				
	CPE520	Mobile Computing		1-0-3	2
	CSC524	Cloud Computing		2-0-0	2

	CSC510 CSC 528	Human Language Processing & Application Agile Software Development		2-0-3 1-0-3	3 2
				16-4-21	**23**

Note that CSE 504 is a continuation of CSE 503

Course Contents

CSC101: INTRODUCTION TO COMPUTING I (2 UNIT [2-0-0])

a. Definition of computer and computer related concepts such as programme, computer software:

Systems and application programmes; minicomputers, mainframes and supercomputer.

b. Discussion of selected application of personal computers: word processing, database

management, spreadsheet, graphics, data analysis.

c. Comprehensive history of modern computer technology. Evolution of microcomputer systems.

History of computer programme.

d. Number system: Binary, Decimal, Hexadecimal. Binary arithmetic; Addition, subtraction,

multiplication, division.

e. Social impact of computers: positive impacts, negative impacts.

CSC 102: INTRODUCTION TO SOFTWARE ENGINEERING (2 UNIT [1-0-3])

a. An introduction to software engineering with emphasis on the practical usage of the personal

computers; concepts of computer hardware, software firmware.

b. Definition of the following terms: bits, bytes, word, word length, data, information, records, fields,

files, database. Data types and organization. Data coding; ASCII Problem-solving process.

Algorithms; flowcharting.

c. Basic logic gates and their operation. Examples with elementary logic circuits.

d. Introduction to a scripting programming language.

CSC 201: COMPUTER PROGRAMMING I (3 UNITS [2-0-3])

a. Brief survey of programming paradigms: Procedural programming , Object-oriented programming,

Functional programming , Declarative programming, non-algorithmic programming, Scripting

languages. The effects of scale on programming methodology.

b. Programming the computer in current version of FORTRAN: Declarative statements; Input and

Output statements; Pro- gram compilation and execution; Control and conditional statements;

Loops and iteration; Functions, Routines and Subprogrammes.

c. Input/Output; File processing; Port addressing.

d. Program testing and debugging techniques.

CSE 201:Elements of Software Construction I 1-0-3 2

The goal of this course will be to provide an introduction to the fundamental principles and techniques of software development that have greatest impact on practice. Possible topics will include capturing the essence of a problem by recognizing and inventing suitable abstractions; key paradigms, including state machines, functional programming, and object-oriented programming; use of design patterns to bridge gap between models and code; the role of interfaces and specification in achieving modularity and decoupling; reasoning about code using invariants; testing, test-case

generation and coverage; and essentials of programming with objects, functions, and abstract types

CPE 203: INTRODUCTION TO DIGITAL COMPUTER SYSTEMS I (2 UNITS [2-0-0])

a. Binary arithmetic operations addition, subtraction, multiplication, division. Binary point. Floating

point and fixed point. Basic logic gates, symbols, truth tables. Boolean Alge bra; Theorems,

Minimization methods, Karnaugh maps (up to six variables) etc.

b. Logic expressions, minimization, sum of product forms.

c. Physical considerations (gate delays, fan-in, and fan-out).

d. Error correlation, parity.

e. Introduction to type of circuits- combinatorial and sequential.

f. Introduction to instruction format: OPCODE OPERAND. Binary representation: bits, nibbles, bytes:

character representation: numeric, non-numeric, alphanumeric, EBCDIC, BCD, ASCII, ANSI.

CSC 202: COMPUTER PROGRAMMING II (2 UNITS [0-0-6]) (Pre; CSC 201)

a. This is a programming laboratory course consisting of applications of programming, through case study problems. Students are expected to carry out four laboratory assignments and make two oral presentations after the completion of the second and the fourth assignments. Programmes will be developed using FORTRAN 90 or latest version of FORTRAN.
Laboratory Assignment I: Programming basics- Data type, basic programme structure; Compiling

and executing programmes in text and graphics environment.

b. Laboratory Assignment II: Loop, arrays, searching and sort ing.

c. Laboratory Assignment III: Function, Routine, Subroutine subprogramme: multiple procedure

calls from a main programme

d. Laboratory Assignment IV: Extensive programming problem with Application to student's

field of study and interest.

CPE 204: INTRODUCTION TO DIGITAL COMPUTER SYSTEMS II (2 UNITS [2-0-0]) (Pre; CSE 203)

a. Fundamental building blocks (logic gates, flip-flops, counters, registers, PLA/PAL)

b. Design of combinatorial circuits with emphasis on application.

c. Design of sequential circuits, synchronous and asynchronous.

d. Design of simple CPU, peripheral devices (I/O), memory/storage devices

e. Stored-program concept.

.

CPE 206: DIGITAL LABORATORY (2 UNITS [0-0-6]) (Pre; CSE 203)

a. Experience with digital systems design tools: Truth tables, Karnaugh maps (up to six

variables)

b. Experience with the use of Programmable Array Logic and Programmable Logic Array.

c. Design, implementation, and testing of the following: Circuits using basic logic operations;

devices based on Combinational logic: Devices using simple latches and flip-flop (Using, for

example, TTL devices)

d. Experiments with Binary adders, synchronous counters, up- down counters.

CSE 202:Elements of Software Construction II 1-0-3 2

In the course, students learn the fundamentals of structuring a web application and writing modular code, with an emphasis on conceptual design to achieve clarity, simplicity, and modularity.

CPE 301: LOW LEVEL LANGUAGES (3 UNITS [2-0-3]) (Pre: CPE204)

a. Bits, bytes, and words, Numeric data representation and number bases, Fixed- and floating-point systems, Signed and twos-complement representations, Representation of nonnumeric data (character codes, graphical data), Representation of records and arrays

b. Basic organization of the von Neumann machine. Detailed architecture of a microprocessor (Intel, Zilog, or Motorola) Control unit; instruction fetch, decode, and execution, Instruction sets and types (data manipulation, control, I/O)

c. Assembly/machine language programming, Instruction for-Controls-Loops and Switches. Macro Instructions. Interrupts and modular programming, Input/Output. File Structures and Storage

e. Control Relocatable Code, Linking/Loading. Device driver writing.

f. Introduction to C programming language as modern tools for low level programming. Port addressing in C. Bit slicing, Bit concatenation. Advanced memory management in C.

CSC 305: INTRODUCTION TO DATABASE SYSTEMS (3 UNITS [2-1-0])

a. Database and databases users

b. Database system concepts and architecture; data modeling using Entity-Relationship (ER)

and Enhanced Entity-Relationship (EER) model, Record storage and primary file organization,

index structures for files

c. Overview of database languages; SQL (data definition, query formulation, update

sublanguage, constraints, integrity); Query optimization

d. QBE and 4th-generation environments, Embedding non-procedural queries in a procedural

language

e. Database security and authorization

f. Data warehousing and data mining

g. Introduction to Object Query Language

CSC: 311: INTRODUCTION TO INFORMATION SYS- TEM (2 UNITS [2-0-0])

a. History of information communication and information system

b. Definitions and basic concepts of IS.

Component, Structure, Configuration and management of In- formation System

d. The Internet and Internet based services; Internet base ap plication development

e. The World Wide Web(WWW) and Website development: page design and development (MS Outlook)

f. Fundamentals of HTTP based programming.

CSE 321: Fundamentals of Requirements Engineering (2 UNITS [2-0-0])

Current techniques, methods, tools and processes used in requirements analysis, definition and specification, including system modeling.

CSC 312: Systems Analysis and Design (2 UNITS [1-0-3])

Vital steps in systems analysis: Techniques of systems analysis. General Systems. Considerations: Data capture; Data management; Data security; Communications systems Maintenance, User involvement; Project handling and control.

CSC 317: AUTOMATA THEORY AND COMPUTABIL- ITY (2 UNITS [1-0-3])

Finite-state machines, Context-free grammars, Tractable and in- tractable problems Uncomputable functions, the halting problem, Implications of uncomputability Deterministic Finite automata (DFAs), Non-deterministic finite automata (NFAs) Equivalence of DFAs and NFAs, Regular expressions.

The pumping lemma for regular expressions, Push-down au- tomata (PDAs), Relationship of PDAs and context-free grammars, Properties of context-free grammars, Turing machines, Non-deterministic Turing machines, Sets and languages, Chomsky hierarchy, The Church-Turing thesis.

CSC 302 Data Structure and Analysis of Algorithms **(3 UNITS [2-0-3])**

a. Basic structures for data representation -Data definition languages.

b. Sequential and linked storage allocation, for linear lists, for multi-linked structures, for string processing techniques, Trees –implementation, traversal, mathematical properties, balanced, trees, heaps, hash tables 82

c. Efficient algorithms for sorting, searching, and selection. Algorithm analysis: worst and average, case analysis. Recurrences and asymptotic

d. Algorithm design techniques: divide-and-conquer, dynamic programming, greedy algorithms, amortized analysis

e. Primitive types, Arrays, Records, Strings and string processing, Data representation in memory, Static, stack, and heap allocation, Runtime storage management, Pointers and references, Linked structures, Implementation strategies for stacks, queues, and hash tables

f. Algorithms for fundamental graph problems such as depth first search, connected components, topological sort, and shortest paths. Possible additional topics: network flow, string searching, parallel computation g. C/C++ Programme to implement basic data structure and their manipulation.

CSC 308: NUMERICAL COMPUTATIONS II (3 UNITS [2-0-3]) (Pre: CSC 307)

a. Curve fitting; function approximation

b. Numerical differentiation and integration (Simpson's Rule, etc.)

c. Explicit and implicit methods

d. Differential equations (Euler's Method, etc.)

e. Linear algebra

f. Finite differences

CSE 302: Object Oriented Programming

This course introduces the object-oriented paradigm in software development to the students. The basic concepts that are focused upon include: objects, classes, messaging, inheritance, methodologies; Analysis: defining objects, structures, attributes, services; Design: transforming the analytic model into the design model using TCL or Java. Implementation: comparison of the support features provided by languages such as Java.
a. Definition of alphabet, syntax and semantics as used in selected higher-level language. Explanation of the terms; Variables, data types, expressions, and assignment statements, Simple I/O, Conditional and iterative control structures.

b. Multidimensional arrays; pointers, pointer to pointer, dynamic data structures and memory allocation

c. Functions and parameter passing, Structured decomposition.

d. Development of programmes for solving basic economic, engineering and scientific problems.

e. Programming with relevant and current OOP languages.

f. Implementation of object-oriented concepts; encapsulation and information-hiding, Separation of

behavior and imple- mentation, classes and subclasses, Inheritance (overriding, dynamic dispatch),

Polymorphism (subtype polymorphism vs. inheritance), Class hierarchies.

G Control structures, Arrays, Methods, Applets.

h. Graphics and event handling

i. Programme testing and exception handling

CSE 322: Software Design (3 UNITS [2-1-0])

The goal here will be to teach modular software design. Different architectural design styles for software. UML diagrams, Client/server paradigm. Choice of appropriate software and hardware system capabilities. Dealing with timing constraints. Formal software design specification techniques. Configuration management. Software design for distributed systems. Reusability and commercial off-the-shelf software modules

CSE 324: Software Testing (2 UNITS [1-0-3])

Testing, Validation and Verification Role of software testing, verification and validation (VV) in the system life cycle. Current techniques, tools and methods are addressed as well as current testing and VV standards. Laboratory instruction.

CPE 316: INTRODUCTION TO ARTIFICIAL INTEL- LIGENCE (2 UNITS [1-0-3])

a. History of artificial intelligence, Philosophical questions: Definition of AI, The Turing test Searle's

"Chinese Room" thought experiment, Ethical issues in AI, Fundamental definitions

Optimal vs. human-like reasoning, Optimal vs. human-like behavior.

b. Modeling the world; the role of heuristics.

c. Knowledge representation and reasoning: Review of propositional and predicate logic Resolution

and theorem proving, Nonmonotonic inference, Structured representation, Frames and objects,

Description logics, Inheritance systems, Non- monotonic reasoning, Non classical logics, Default

reasoning, Belief revision, Preference logics, Integration of knowledge sources, Aggregation of

conflicting belief, Reasoning on ac- tion and change, Situation calculus, Event calculus, Temporal

and spatial reasoning(breadth-first, depth-first, depth- first with iterative deepening).

d. Softcompuring: Fuzzy sets and possibility theory, neural net- works, genetic algorithm.

Two-player games (minimax search, alpha-beta pruning; illustration with Ayo)

f. Introduction to robotics and AI planning

g. Developments in artificial intelligence, natural language understanding, knowledge

representation, expert systems, pat- tern recognition, the language LISP or PROLOG.

CSE 421:Software Systems Architecture (2 UNITS [2-0-0])

Software Architecture Knowledge of complex programs recommended. Domain models, generic architectures and software design as well the context, scope, current and future state of software architecture.

CPE 401: COMPUTER ARCHITECTURE AND ORGA- NIZATION (3 UNITS [3-0-0])

a. Definition. Structural and functional characteristics of computer system components.

b. Organization and design of digital computing systems; description of current typical computing structure, processor micro-architecture and pipelining; cache and virtual memory organizations CPU configuration and possible architecture software/ hardware tradeoffs.

c. Introduction to SIMD, MIMD, VLIW, EPIC, Systolic architecture, Interconnection networks (hypercube, shuffle-exchange, mesh, crossbar), Shared memory systems, Cache coherence, Memory models and memory consistency.

d. Superscalar architecture, Branch prediction, Pre-fetching, Speculative execution, multithreaded architecture, Scalability, sym- metric multiprocessors; and parallel computers.

CSC 403: PRINCIPLES OF COMPILERS (3 UNITS [3- 0-0])

a. Definition; Basic concepts of compiler.

b. Application of regular expressions in lexical scanners, Parsing

(concrete and abstract syntax, abstract syntax trees).

c. Application of context-free grammars in table-driven and recursive- descent parsing

d. Symbol table management, code generation by tree walking

e. Compilation approaches - Multipass, Single Pass, Load and Go; Compiler

implementation- Scanning, syntax directed table driven. Architecture-specific operations:

instruction selection and reg ister allocation

g. Optimization techniques

h. The use of tools in support of the translation process and the advantages thereof Program

libraries and separate compila tion Building syntax-directed tools

CPE 405: INTRODUCTION TO MICROPROCESSOR TECHNOLOGY AND MICRO-PROGRAMMING (3 UNITS [3-0-0])

a. I/O fundamentals: handshaking, buffering, programmed I/O, interrupt-driven I/O

b. Interrupt structures: vectored and prioritized, interrupt ac knowledgment

c. External storage, physical organization, and drives

d. Buses: bus protocols, arbitration, direct-memory access (DMA)

e. Introduction to multiprocessor configuration and parallel processing

f. Multimedia support

g. RAID architectures

h. Microprocessor architecture and microprogramming model. Design requirements for microinstructions and instruction reper toire.

CSC 415: OPERATING SYSTEMS AND SYSTEM PRO- GRAMMING (3 UNITS [3-0-0])

a. Role and purpose of the operating system

b. History of operating system development

c. Functionality of a typical operating system

d. Mechanisms to support client-server models, hand-held de- vices

e. Design issues (efficiency, robustness, flexibility, portability, security, compatibility)

f. Influences of security, networking, multimedia, windows: Struc- turing methods (monolithic, layered, modular, micro-kernel models): Abstractions, processes, and resources: Concepts of application program interfaces (APIs): Application needs and the evolution of hardware/software techniques: Device organization: Interrupts: methods and implementations: Con- cept of user/system state and protection, transition to kernel mode

g.File systems: Management of secondary stores, Examples of operating systems VM, OS, DOS/VS, WINDOWS, UNIX, LINUX, etc. A detailed study of two systems most commonly used in the country.

CSE 425:Software Engineering for Web Applications (3 UNITS [2-0-3])

With this course, it is expected that students already have some programming and software engineering experience. The goal is to give students some experience in dealing with those challenges that are unique to Internet applications, such as: concurrency; unpredictable load; security risks; opportunity for wide-area distributed computing;

creating a reliable and stateful user experience on top of unreliable connections and stateless protocols; extreme requirements and absurd development schedules; requirements that change mid-way through a project, sometimes because of experience gained from testing with users; user demands for a multi-modal interface. This can serve as a drive for those students interested in Software Engineering for Cyber-Physical Systems, Smart Cities etc

CSC 505: OBJECT-ORIENTED SYSTEMS DESIGN (2 UNITS [2-0-0])

Software design and construction in context of large OOP libraries. Taught in Java. Topics: review of OOP, the structure of Graphical User Interface (GUI) OOP libraries, GUI application design and construction, OOP Software Development strategies, approaches to programming in teams

CSC 501: INTRODUCTION TO OPERATIONS RESEARCH (2 UNITS [2-0-0])

a. Linear programming

b. Integer programming

c. The Simplex method

d. Probabilistic modeling

e. Queuing theory, Petri nets Markov models and chains

f. Optimization, Network analysis and routing algorithms.

g. Prediction and estimation, Decision analysis, Forecasting, Risk management, Econometrics,

microeconomics, Sensitivity analysis, Dynamic programming, Sample applications, Software tools.

CSE 521:Aspect Oriented Development (2 UNITS [1-0-3])

Aspect-Oriented software development (AOD) is a new programming paradigm that increases modularity with a separation of cross-cutting concerns. This course provides a broad perspective of AOD. The topics include: Aspect-Oriented design in C# and visual programming languages, separation of concern in .Net web development, visual simulations, patterns and frameworks, aspects design in video game and robotics software development.

CSE 519: Formal Methods and Program Verification (2 UNITS [1-0-3])

Why verification? Advantages and disadvantages, Costs and gains. Basics of deductive program Verification Hoare Logic and Dynamic Logic

CSE 512: Software Maintenance (2 UNITS [2-0-3])

Topics include a maintenance lifecycle model, corrective maintenance, adaptive maintenance, and perfective maintenance, preventative maintenance, reverse engineering techniques, configuration management, process engineering and IT Governance

CSC 507: COMPUTERS IN SOCIETY (2 UNITS [2-0-0])

a. Introduction to the effects of the application of computers on social life: in education, medicine, business, science and research, etc. Effects on skills evaluation and employment.

b. Social implications of networked communication (WWW). Growth of, control of, and access to the Internet and Inter- national issues

d. Computer usage appropriate for education; Promises and achievements of artificial intelligence; the computerized society; limitations of computers; human failures; Professional responsibility.

e. Computer crime; history and examples of computer crime "Cracking" (hacking) and its effects. Viruses, worms, and Trojan horses Crime prevention strategies

CSC 523: MATHEMATICAL PROGRAMMING (2 UNITS [2-0-0])

a. Counting arguments – Sum and product rule, – Inclusion- exclusion principle, Linear programming: Convex sets and linear forms.

b. Graphical solution of linear programming problems. The basic simplex algorithm. The dual problems. Integer linear programming.

Arithmetic and geometric progressions,– Fibonacci numbers, The pigeon hole principle,

d. Permutations and combinations, Basic definitions, Pascal's identity, The binomial theorem, Solving recurrence relations, Common examples, The Master theorem.

CSC 515: DATABASE DESIGN AND MANAGEMENT (2 UNITS [2-0-0])

a. Functional dependency; Normal forms (1NF, 2NF, 3NF, BCNF); Multivalued dependency (4NF);

Join dependency (PJNF, 5NF)

b. Data modeling; Conceptual models (including entity-relationship and UML) Object-oriented

model.

c. Relational data model. Mapping conceptual schema to a relational schema Entity and

referential integrity

d Relational algebra and relational calculus.

e. Distributed data storage, Distributed query processing, Distributed transaction model,

Concurrency control, Homogeneous and heterogeneous solutions, Client-server

CSE 527: SOFTWARE ECONOMICS (2 UNITS [2-0-0])

Possible topics to cover in this course include: cost estimation of software projects; business cases for software projects; software valuation; value-driven software design; economic drivers of buy-versus-build and software outsourcing decisions; business models of software companies; international software product distribution channels; electronic software distribution; business partnerships involving software companies,

CPE 502: HARDWARE SYSTEM STUDIES (3 UNITS [1-1-3])

a.. Design and implementation of simple systems that utilize the following devices:

Multiplexer/Demultiplexer; MODEMS, ADC, DAC.

b. Hardware circuit reliability: Graceful degradation; MTTR, MTBF, etc.

c. Computer selection for a given application, economic versus technical considerations.

d. A survey of fundamentals with emphasis on hardware and systems concepts.

e. Interfaces and configurations.

CSC 508: COMPUTER SYSTEM PROJECT MANAGEMENT (2 UNITS [2-0-0])

a. Definition of computer project and project management.

b. Components and features of a good computer based project management technique.

Computer network administration.

c. Function of members and team management, team processes, team organization and decision-making, roles and responsi bilities in a software team, role identification and assignment, team problem resolution.

d. Project trackingSoftware Project scheduling, Budgeting and Planning; Project organization.

f. Software measurement and estimation techniques, Risk analysis.

g. Software quality assurance

h. Software configuration management

i. Project management tools

CSE 526: Software Engineering Process and Configuration Management (3 UNITS [2-1-0])

Software Processes Detailed coverage of the theory, application, assessment and evaluation of the Unified Process Model. Course will cover the process modelling, process assessment, quality assessment of process models and process improvement techniques

CSE 514: Software Quality Management (2 UNITS [1-0-3])

Software metrics, Quality planning and quality control. Inspections and formal technical reviews. problem analysis and reporting techniques. Verification and validation techniques. Process and product quality assessment. Process measurement. Software quality assurance standards. In addition to what is in the initial content. Evolution, principles, Quality improvement, quality standards, Quality management software, Quality terms, etc.

CSE 503/504: INDIVIDUAL PROJECT I/II (3 UNITS [0-0-9]) each)

This offers an opportunity for graduating year students to apply theoretical, conceptual and practical knowledge gained to solve selected problem, either within the Technology itself or any other area of application subject to approval of the Department of Computer Science and Engineering and viability of adequate supervision. At the end of the project, the student submits a report. This course is a compulsory requirement for the three programmes offered in the Department.

CSC 524: TECHNIQUES IN DATA ANALYSIS (2 UNITS [2-0-0])

a. Data collection and coding. Data cleaning - completeness, range, consistency.

b. Design of algorithms for statistical analysis; count, sum, mean, standard deviation, analysis of

variance, Frequency, cross- tabulation, multiple regression.

c. Design of algorithm for implementing Non parametric statis tical test. Case studies using SPSS or

PSTAT or SAS pack- ages.

d. Software implementation of statistical algorithms.

e. Interpretation of results.

CPE 510: HUMAN LANGUAGE PROCESSING & APPLICATION (2 UNITS [2-0-0])

a. Deterministic and stochastic grammars, Parsing algorithms, Corpus-based methods, Information retrieval, Language translation.

b. Fundamentals of natural language modeling (using a local language; Yoruba, Hausa, Igbo, or any other African Lan guage).

c. Speech recognition; isolate, connected and continuous speech; definition, methods, problems, examples; speaker recognition.

d. Speech synthesis; articulatory, format, concatinative, defini tion, methods, problems, examples.

e. Speech understanding; definition, methods, problems, exam- ples.

f. Natural language based interface to intelligent systems

g. Web based speech interface design.

CSC 506: INDUSTRIAL APPLICATION STUDIES (3 UNITS 2-1-0)

These are special application courses arranged with industry for the students in-depth experience of industrial type problems. A mini- system is defined and solution proposed. Students are required to apply knowledge acquired over the course of their training in proposing solution to industrial type problems. At the end of each course the student submits a detailed system report. The student gives one seminar on each problem solved. Different seminar is organized with speakers from the Industry to intimate the students with Industry requirements and expectations. Also, postgraduate research works are presented to expose students to possible research areas for future endeavours.

CSE 529: AGILE SOFTWARE DEVELOPMENT (2 UNITS [2-0-0])

Topics here can include: Introduction to Agile methods, behaviour-driven software development, continuous integration, deployment of (a choice programming language) on (any given application) to the cloud, development and deployment of a sample software product

CSE 523: PRINCIPLES OF COMPUTER HUMAN INTERACTION (3 UNITS 2-0-3)

An introduction to the concepts underlying the design of human computer interaction: usability and affordances, systemic design methods, user conceptual models and interface metaphors, human

cognitive and physical ergonomics, information and inter activity structures, design tools and environments. Structured around a set of case studies in which notable interface designs, and/or projects are analyzed as illustrative of underlying principles. Discussion of cases and interface analysis and design exercises.