COURSE STRUCTURE & DESCRIPTION OF COURSE CONTENT

8.1 Course Structure

The course structure contains course code, course title and credit units (CU) spread according to the semester they will be taught from level 100 to level 400.

Table 8.1: Course Structure - Level 100

LEVEL 100

First Semester

Course Code	Course Title	Credit Units	Course Status	LH	PH
GST 111	Communication in English	2	**C**	30	0
COS 101	Introduction to Computer Science	3	**C**	30	45
MTH 101	Elementary Mathematics I	2	**C**	30	0
STA 111	Descriptive Statistics	3	**C**	45	0
PHY 101	General Physics I	2	**C**	30	0
PHY 107	General Practical Physics I	1	**C**	0	45
SLU-COS 121	Introduction to Application Packages	2	**C**	15	45
Total		**15 Units**			

Second Semester

Course Code	Course Title	Credit Units	Course Status	LH	PH
GST 112	Nigeria People and Culture	2	**C**	30	0
COS 102	Introduction to Problem Solving	2	**C**	15	45
MTH 102	Elementary Mathematics II	2	**C**	30	0
PHY 102	General Physics II	2	**C**	30	30
PHY 108	General Practical Physics II	1	**C**	0	45

SLU-IFT 102	Introduction to Information System	2	C	30	0
SLU-STA 112	Probability I	2	C	30	0
SLU-COS 104	Introduction to File Processing	2	C	15	45
Total		**15**			

Summary: Level 100 Student should register a total of **30 credit unit** for this session; **15 credits** for first semester and **15 credit** for second semester.

Table 8.2: Course Structure - Level 200

LEVEL 200

First Semester

Course Code	Course Title	Credit Units	Course Status	LH	PH
COS 201	Computer Programming - I	3	**C**	30	45
CSC 203	Discrete Structures	2	**C**	30	00
IFT 211	Digital Logic Design	2	**C**	15	45
SEN 201	Introduction to Software Engineering	2	**C**	30	00
MTH 201	Mathematical Methods	2	**C**	30	00
ENT 211	Entrepreneurship and Innovation	2	**C**	30	00
SLU-IFT 207	Information Technology in Business	2	**C**	30	00
Total		**15 Units**			

Second Semester

Course Code	Course Title	Credit Unit	Status	LH	PH
COS 202	Computer Programming - II	3	**C**	30	45
IFT 212	Computer Architecture and Organization	2	**C**	15	45

MTH 202	Elementary Differential Equations	2	**C**	30	00
GST 112	Philosophy, Logic and Human Existence	2	**C**	30	00
CSC 299	SIWES I	3	**C**	00	135
SLU-COS 204	System Analysis and Design	3	**C**	30	45
Total		**15 Units**			

Summary: Level 200 Student should register a total of **30 credit unit** for this session; **15 credit** for first semester and **15 credit** for second semester.

Note: SIWES 1: Students are to observe mandatory three (3) months SIWES at the end of the session.

Table 8.3: Course Structure - Level 300

LEVEL 300
First Semester

Course Code	Course Title	Credit Units	Status	LH	PH
CSC 301	Data Structures	3	**C**	30	45
CSC 309	Artificial Intelligence	2	**C**	15	45
ICT 305	Data Communication System & Networking	3	**C**	30	45
CYB 303	Introduction to Cyber Security and Strategy	2	**C**	30	00
SLU-COS 303	Java Programming	3	**C**	30	45
SLU-COS 305	Computer Graphics & Animation	2	**C**	30	45
Total		**15 Units**			

Second Semester

Course Code	Course Title	Credit Units	Course Status	LH	PH

CSC 308	Operating Systems	3	**C**	30	45
CSC 322	Computer Science Innovation & New Technologies	2	**C**	15	45
GST 312	Peace and Conflict Resolution	2	**C**	30	00
ENT 312	Venture Creation	2	**C**	15	45
DTS 304	Data Management - I	3	**C**	30	45
CSC 399	SIWES II	3	**C**	00	135
Total		**15 Units**			

Summary: Level 300 Student should register a total of **30 credit unit** for this session; **15 credit** for first semester and **15 credit** for second semester.

Note: SIWES II: Students are to observe mandatory three (3) months SIWES at the end of the session.

Table 8.4: Course Structure - Level 400

LEVEL 400
First Semester

Course Code	Course Title	Credit Units	Course Status	LH	PH
COS 409	Research Methodology and Technical Report Writing	2	**C**	30	00
CSC 401	Algorithms and Complexity Analysis	2	**C**	30	00
INS 401	Project Management	2	**C**	30	00
CSC 497	Project I	3	**C**	00	135
SLU-COS 403	Human Computer Interface	3	**C**	30	45
SLU-COS 405	Database Design and Management II	3	**C**	30	45
Total		**16 Units**			

Second Semester

Course Code	Course Title	Credit Units	Course Status	LH	PH
CSC 402	Ethics & Legal Issues in Computer Science	2	**C**	30	00
CSC 498	Project II	3	**C**	00	135
SLU-CSC 404	Organization of Programming Languages	3	**C**	45	00
SLU-CSC 406	Compiler Construction	3	**C**	45	00
SLU-CSC 408	Object Oriented Programming	3	**C**	30	45
SLU-CSC 410	Internet of Things	2	**E**	15	45
Total		**16 Units**			

Summary: Level 400 Student should register a total of **32 credit unit** for this session; **16 Core credit** from first semester and **14 Core credit** & **2 Elective credit** from second semester.

Summary of TCU

S/ No.	Level	UTME	DE
1.	100	30	-
2.	200	30	30
3.	300	30	30
4.	400	32	32
Total		**122**	**92**

8.2 Description of Course Content

8.2.1 Level 100

GST 111: Communication in English **(2 Units C: LH 15; PH 45)**

Learning Outcomes:

At the end of this course, students should be able to:

1. identify possible sound patterns in English Language;

2. list notable Language skills;

3. classify word formation processes;

4. construct simple and fairly complex sentences in English;

5. apply logical and critical reasoning skills for meaningful presentations;

6. demonstrate an appreciable level of the art of public speaking and listening; and

7. write simple and technical reports.

Course Contents

Sound patterns in English Language (vowels and consonants. Phonetics and phonology). English word classes (lexical and grammatical words, definitions, forms, functions, usages, collocations). Sentence in English (types: structural and functional, simple and complex). Grammar and Usage (tense, mood, modality and concord, aspects of language use in everyday life). Logical and Critical Thinking and Reasoning Methods (Logic and Syllogism, Inductive and Deductive Argument and Reasoning Methods, Analogy, Generalisation and Explanations). Ethical considerations, Copyright Rules and Infringements. Writing Activities: (Pre-writing, Writing, Post-writing, Editing and Proofreading; Brainstorming, outlining, Paragraphing, Types of writing, Summary, Essays, Letter, Curriculum Vitae, Report writing, Note making, etc. Mechanics of writing). Comprehension Strategies: (Reading and types of Reading, Comprehension Skills, 3RsQ). Information and Communication Technology in Modern Language Learning. Language skills for effective communication. Major word formation processes. Writing and reading comprehension strategies. Logical and critical reasoning for meaningful presentations. Art of public speaking and listening. Report writing.

GST 112: Nigerian Peoples and Culture (2 Units C: LH 30)

Learning Outcomes:

At the end of the course, students should be able to:

1. analyse the historical foundation of the Nigerian culture and arts in pre-colonial times;

2. list and identify the major linguistic groups in Nigeria;

3. explain the gradual evolution of Nigeria as a political unit;

4. analyse the concepts of Trade, Economic and Self-reliance status of the Nigerian peoples towards national development;

5. enumerate the challenges of the Nigerian State towards Nation building;

6. analyse the role of the Judiciary in upholding people's fundamental rights;

7. identify acceptable norms and values of the major ethnic groups in Nigeria; and

8. list and suggest possible solutions to identifiable Nigerian environmental, moral and value problems.

Course Contents

Nigerian history, culture and art up to 1800 (Yoruba, Hausa and Igbo peoples and culture; peoples and culture of the ethnic minority groups). Nigeria under colonial rule (advent of colonial rule in Nigeria; Colonial administration of Nigeria). Evolution of Nigeria as a political unit (amalgamation of Nigeria in 1914; formation of political parties in Nigeria; Nationalist movement and struggle for independence). Nigeria and challenges of nation-building (military intervention in Nigerian politics; Nigerian Civil War). Concept of trade and economics of selfreliance (indigenous trade and market system; indigenous apprenticeship system among Nigeria people; trade, skill acquisition and self-reliance). Social justice and national development (law definition and classification). Judiciary and fundamental rights. Individual, norms and values (basic Nigeria norms and values, patterns of citizenship acquisition; citizenship and civic responsibilities; indigenous languages, usage and development; negative attitudes and conducts. Cultism, kidnapping and other related social vices). Re-orientation, moral and national values (The 3R's – Reconstruction, Rehabilitation and Re-orientation) Reorientation Strategies: Operation Feed the Nation (OFN), Green Revolution, Austerity Measures, War Against Indiscipline (WAI), War Against Indiscipline and Corruption (WAIC), Mass Mobilisation for Self-Reliance, Social Justice and Economic Recovery (MAMSER), National Orientation Agency (NOA). Current socio-political and cultural developments in Nigeria.

COC 101: Introduction to Computer Science (3 Units: LH 30, PH 45)

Learning Outcomes:

At the end of the course, students should be able to:

1. explain basic components of computers and other computing devices;

2. describe the various applications of computers;

3. explain information processing and its roles in the society;

4. describe the Internet, its various applications and its impact;

5. explain the different areas of the computing discipline and its specializations; and

6. demonstrate practical skills on using computers and the internet.

Course Contents

Brief history of computing. Description of the basic components of a computer/computing device. Input/Output devices and peripherals. Hardware, software and human ware. Diverse and growing computer/digital applications. Information processing and its roles in society. The Internet, its applications and its impact on the world today. The different areas/programs of the computing discipline. The job specializations for computing professionals. The future of computing.

Lab Work: Practical demonstration of the basic parts of a computer. Illustration of different operating systems of different computing devices including desktops, laptops, tablets, smart boards and smart phones. Demonstration of commonly used applications such as word processors, spreadsheets, presentation software and graphics. Illustration of input and output devices including printers, scanners, projectors and smartboards. Practical demonstration of the Internet and its various applications. Illustration of browsers and search engines. How to access online resources.

COC 102: Introduction to Problem Solving (2 Units C: LH 30; PH 0)

Learning Outcomes:

At the end of this course, students should be able to:

1. explain problem solving processes;

2. demonstrate problem solving skills;

3. describe the concept of algorithms development and properties of algorithms;

4. discuss the solution techniques of solving problem;

5. solve computer problems using algorithms, flowcharts, pseudocode, etc.; and

6. solve problems using programming language using C, PYTHON etc.

Course Contents

Core concepts of computing. Identification of problems. Types of problems (routine problems and non-routine problems). Problem-solving. Methods of solving computing problems. Algorithms and heuristics. Solvable and unsolvable problems. Solution techniques of solving problems; abstraction; analogy; brainstorming; trial and error; hypothesis testing; reduction; literal thinking; means-end analysis. Method of the focal object; morphological analysis; research; root cause analysis; proof; divide and conquer. General Problem-solving process. Solution formulation and design; flowchart; pseudocode; decision table; decision tree. Programming in any language. 210 Lab Work: Use of simple tools for algorithms and flowcharts; writing pseudocode; writing assignment statements, input-output statements and condition statements; demonstrating simple programs using any programming language (Visual Basic, Python, C)

MTH 101: Elementary Mathematics I (Algebra and Trigonometry) (2 Units C: LH 30)

Learning Outcomes:

At the end of the course students should be able to:

1. understand the basic definition of Set, Subset, Union, Intersection, Complements and use of Venn diagrams;

2. solve quadratic equations;

3. solve trigonometric functions;

4. understand various types of numbers; and

5. solve some problems using the Binomial theorem.

Course Contents

Elementary set theory, subsets, union, intersection, complements, venn diagrams. Real numbers; integers, rational and irrational numbers, mathematical induction, real sequences and series, theory of quadratic equations, binomial theorem. Complex numbers; algebra of complex numbers; the Argand diagram. De-Moivre's theorem, nth roots of unity. Circular measure, trigonometric functions of angles of any magnitude, addition and factor formulae.

MTH 102: Elementary Mathematics II (Calculus) (2 Units C: LH 30)

Learning Outcomes:

At the end of the course, students should be able to:

1. distinguish types of rules in Differentiation and Integration;

2. describe the meaning of Function of a real variable, graphs, limits and continuity; and

3. solve some applications of definite integrals in areas and volumes.

Course Contents

Function of a real variable, graphs, limits and idea of continuity. The derivative, as limit of rate of change. Techniques of differentiation. Extreme curve sketching; Integration as an inverse of differentiation. Methods of integration, Definite integrals. Application to areas, volumes.

STA 111: Descriptive Statistics: (3 Units C: LH 45)

Learning Outcomes:

At the end of the course, students should be able to:

1. explain the basic concepts of descriptive statistics.

2. present data in graphs and charts.

3. differentiate between measures of location, dispersion and partition.

4. describe the basic concepts of Skewness and Kurtosis as well as their utility function in a given data set.

5. differentiate rates from ratio and how they are use.

6. compute the different types of index number from a given data set and interpret the output.

Course content

Statistical data. Types, sources and methods of collection. Presentation of data. Tables chart and graph. Errors and approximations. Frequency and cumulative distributions. Measures of location, partition, dispersion, skewness and Kurtosis. Rates, ratios and index numbers.

SLU-PHY 101 General Physics I (Mechanics) (2 Units C: LH 30)

Learning Outcomes:

At the end of the course students should be able to:

1. identify and deduce the physical quantities and their units;

2. differentiate between vectors and scalars;

3. describe and evaluate motion of systems on the basis of the fundamental laws of mechanics;

4. apply Newton's laws to describe and solve simple problems of motion;

5. evaluate work, energy, velocity, momentum, acceleration, and torque of moving or rotating objects;

6. explain and apply the principles of conservation of energy, linear and angular momentum.

7. describe the laws governing motion under gravity; and

8. explain motion under gravity and quantitatively determine behaviour of objects moving under gravity.

Course Contents

Space and time; units and dimension, Vectors and Scalars, Differentiation of vectors: displacement, velocity and acceleration; kinematics; Newton laws of motion (Inertial frames, Impulse, force and action at a distance, momentum conservation); Relative motion; Application of Newtonian mechanics; Equations of motion; Conservation principles in physics, Conservative forces, conservation of linear momentum, Kinetic energy and work, Potential energy, System of particles,

Centre of mass; Rotational motion; Torque, vector product, moment, rotation of coordinate axes and angular momentum. Polar coordinates; conservation of angular momentum; Circular motion; Moments of inertia, gyroscopes and precession; Gravitation: Newton's Law of Gravitation, Kepler's Laws of Planetary Motion, Gravitational Potential Energy, Escape velocity, Satellites motion and orbits.

SLU-PHY 102: General physics II (Electricity & magnetism) (2 Units C: LH 30)

Learning Outcomes At the end of the course, students should be able to:

1. describe the electric field and potential, and related concepts, for stationary charges;

2. calculate electrostatic properties of simple charge distributions using Coulomb's law, Gauss's law, and electric potential;

3. describe and determine the magnetic field for steady and moving charges;

4. determine the magnetic properties of simple current distributions using Biot-Savart and Ampere's law;

5. describe electromagnetic induction and related concepts and make calculations using Faraday and Lenz's laws;

6. explain the basic physical of Maxwell's equations in integral form;

7. evaluate DC circuits to determine the electrical parameters;

8. determine the characteristics of ac voltages and currents in resistors, capacitors, and Inductors.

Course Contents

Forces in nature. Electrostatics (electric charge and its properties, methods of charging). Coulomb's law and superposition. Electric field and potential. Gauss's law. Capacitance. Electric dipoles. Energy in electric fields. Conductors and insulators. DC circuits (current, voltage and resistance. Ohm's law. Resistor combinations. Analysis of DC circuits. Magnetic fields. Lorentz force. Biot-Savart and Ampère's laws. Magnetic dipoles. Dielectrics. Energy in magnetic fields. Electromotive force. Electromagnetic induction. Self and mutual inductances. Faraday and Lenz's laws. Step up and step down transformers. Maxwell's equations. Electromagnetic oscillations and waves. AC voltages and currents applied to inductors, capacitors, and resistance.

SLU-PHY 107: General Practical Physics I (1 Unit C: PH 45)

Learning Outcomes At the end of the course, students should be able to:

1. conduct measurements of some physical quantities;

2. make observations of events, collect and tabulate data;

3. identify and evaluate some common experimental errors.

4. plot and analyse graphs; and

5. draw conclusions from numerical and graphical analysis of data.

Course Contents

This introductory course emphasises quantitative measurements, the treatment of measurement errors, and graphical analysis. A variety of experimental techniques should be employed. The experiments include studies of meters, the oscilloscope, mechanical systems, 100 electrical and mechanical resonant systems, light, heat, viscosity, etc., covered in PHY 101 and PHY 102. However, emphasis should be placed on the basic physical techniques for observation, measurements, data collection, analysis and deduction.

PHY 108- General Practical Physics II (1 Unit C: PH 45)

Learning Outcomes:

On completion, the student should be able to:

1. conduct measurements of some physical quantities;

2. make observations of events, collect and tabulate data;

3. identify and evaluate some common experimental errors;

4. plot and analyse graphs;

5. draw conclusions from numerical and graphical analysis of data; and

6. prepare and present practical reports.

Course Contents

This practical course is a continuation of PHY 107 and is intended to be taught during the second semester of the 100 level to cover the practical aspect of the theoretical courses that have been covered with emphasis on quantitative measurements, the treatment of measurement errors, and graphical analysis. However, emphasis should be placed on the basic physical techniques for observation, measurements, data collection, analysis and deduction.

SLU-COS 103 Introduction to Application Packages (2 Units; C: LH 15, PH 90)

Learning Outcomes: At the end of this course, students should be able to;

1) Understand the concept of application packages and their role in modern computing.

2) Explore popular application packages across different domains.

3) Learn essential features and functionalities of application packages.

4) Develop practical skills in using application packages for various tasks.

5) Analyze case studies to understand the real-world applications of different application packages.

Course content

Overview of application packages, Importance and benefits, Evolution and trends in application packages. Productivity Suites, Introduction to productivity suites, Microsoft Office Suite (Word, Excel, PowerPoint, Outlook), Google Workspace (Docs, Sheets, Slides, Gmail), Features and functionalities comparison. Graphic Design and Multimedia Packages, Adobe Creative Cloud Suite (Photoshop, Illustrator, InDesign, Premiere Pro), CorelDRAW Graphics Suite, Basics of graphic design and multimedia editing. Introduction to project management tools. Trello, Asana, Basecamp, Collaboration tools: Slack, Microsoft Teams, Zoom. Accounting and Finance Software, QuickBooks, Sage Intacct, Introduction to accounting principles. Introduction to Enterprise Resource Planning (ERP), Oracle ERP Cloud, Overview of ERP systems, Importance and benefits of ERP in businesses. Specialized Application Packages, Industry-specific software (e.g., AutoCAD, MATLAB, SPSS), Healthcare management software, Legal practice management software, Overview and applications in specialized domains

Lab work: Practical exercises in graphic design software, Case studies on effective project management using tools, Hands-on exercises in accounting software

SLU-IFT 132 Introduction to Information System (3 Units: LH 45, P 45)

Learning Outcomes

At the end of this course, Students should be able to:

1. Explain system concepts and organisational processes;

2. Explain information systems principles and application in modern organisation;

3. Describe information technology security and related ethical issues; and

4. Explain database management and system development life cycle.

Course Contents

Roles and relevance of information systems in organisations to conduct business and solve problems. Information systems principles in modern organisations. Systems concepts; organisational processes; technological aspects of information systems. The internet. Information technology security. Ethical issues. Database management. Systems development life cycle.

SLU-STA 112 Probability I (3 Units; C: LH 45)

Learning Outcomes

At the end of the course, students should be able to:

1. Explain the differences between permutation and combination;

2. Explain the concept of random variables and relate it to probability and distribution functions;

3. Describe the basic distribution functions; and

4. Explain the concept of exploratory data analysis.

Course Contents

Permutation and combination. Concepts and principles of probability. Random variables. Probability and distribution functions. Basic distributions: Binomial, geometric, Poisson, normal and sampling distributions; exploratory data analysis.

SLU-COS 104: INTRODUCTION TO FILE PROCESSING: (2 Units; C: LH; 15 PH: 45)

Learning Outcomes

At the end of this course, students should be able to:

1. differentiate between file management and file processing concepts

2. explain file naming, extensions and attributes

3. explain the relationship between computer or electronic files and operating system

4. explain various file organisation and access methods

5. know the relationship between files and directories

6. explain techniques of record blocking

7. explain how operating systems manage used space in computer memory

8. explain various data validation techniques

9. describe the various techniques in improving system performance, reliability, and security

10. explain various data validation techniques

11. discuss various file sorting, searching, and merging algorithms and applications

12. have idea of how programming languages handle or process files for input and output.

Course content:

Introduction to data management files; and job control language application; an over-view of I/O (input/output) system architecture; logical file organizations; mapping logical organization on to physical storage; back-up procedure; file recovery; higher level language data management facilities, file attribute, STREAM files, record files, GET and PUT, OPEN and CLOSE.

8.2.2 LEVEL 200 Courses

ENT 211: Entrepreneurship and Innovation (2 Units C: LH 15; PH 45)

Learning Outcomes

At the end of this course, students should be able to:

1. explain the concepts and theories of entrepreneurship, intrapreneurship, opportunity seeking, new value creation, and risk taking;
2. state the characteristics of an entrepreneur;
3. analyse the importance of micro and small businesses in wealth creation, employment, and financial independence;
4. engage in entrepreneurial thinking;
5. identify key elements in innovation;
6. describe stages in enterprise formation, partnership and networking including business planning;
7. describe contemporary entrepreneurial issues in Nigeria, Africa and the rest of the world; and
8. state the basic principles of e-commerce.

Course Contents

Concept of Entrepreneurship (Entrepreneurship, Intrapreneurship/Corporate Entrepreneurship,). Theories, Rationale and relevance of Entrepreneurship (Schumpeterian and other perspectives, Risk-Taking, Necessity and opportunity-based entrepreneurship and Creative destruction). Characteristics of Entrepreneurs (Opportunity seeker, Risk taker, Natural and Nurtured, Problem solver and change agent, Innovator and creative thinker). Entrepreneurial thinking (Critical thinking, Reflective Thinking, and Creative thinking). Innovation (Concept of innovation, Dimensions of innovation, Change and innovation, Knowledge and innovation). Enterprise formation, partnership and networking (Basics of Business Plan, Forms of business ownership, Business registration and forming alliances and joint ventures). Contemporary Entrepreneurship Issues (Knowledge, Skills and Technology, Intellectual property, Virtual office, Networking). Entrepreneurship in Nigeria (Biography of inspirational Entrepreneurs, Youth and women entrepreneurship, Entrepreneurship support institutions, Youth enterprise networks and Environmental and cultural barriers to entrepreneurship). Basic principles of e-commerce.

GST 212: Philosophy, Logic and Human Existence (2 Units C: LH 30)

Learning Outcomes

At the end of this course, students should be able to

1. know the basic features of philosophy as an academic discipline;
2. identify the main branches of philosophy & the centrality of logic in philosophical discourse;
3. know the elementary rules of reasoning;
4. distinguish between valid and invalid arguments;
5. think critically and assess arguments in texts, conversations and day-to-day discussions;
6. critically asses the rationality or otherwise of human conduct under different existential conditions;
7. develop the capacity to extrapolate and deploy expertise in logic to other areas of knowledge, and
8. guide his or her actions, using the knowledge and expertise acquired in philosophy and logic.

Course Contents

Scope of philosophy; notions, meanings, branches and problems of philosophy. Logic as an indispensable tool of philosophy. Elements of syllogism, symbolic logic— the first nine rules of inference. Informal fallacies, laws of thought, nature of arguments. Valid and invalid arguments, logic of form and logic of content — deduction, induction and inferences. Creative and critical thinking. Impact of philosophy on human existence. Philosophy and politics, philosophy and human conduct, philosophy and religion, philosophy and human values, philosophy and character molding, etc.

SEN 201: Introduction to Software Engineering (2 units C: LH 30)

Learning Outcomes At the end of this course, students should be able to:

1. describe the concept of the software life cycle;

2. explain the phases of requirements analysis, design, development, testing and maintenance in a typical software life cycle;

3. differentiate amongst the various software development models;

4. utilise UML for object oriented analysis and design;

5. describe different design architectures;

6. explain the various tasks involved in software project management; and

7. describe the basic legal issues related to Software Engineering.

Course Contents

Software Engineering concepts and principles. Design, development and testing of software systems. Software processes: software lifecycle and process models. Process assessment models. Software process metrics. Life cycle of software system. Software requirements and specifications.

Software design. Software architecture. Software metrics. Software quality and testing. Software architecture. Software validation. Software evolution: software maintenance; characteristics of maintainable software; re-engineering; legacy systems; software reuse. Software Engineering and its place as a computing discipline. Software project management: team management; project scheduling; software measurement and estimation techniques; risk analysis; software quality assurance; software configuration management. Software Engineering and law.

COS 201: Computer Programming I (3 Units C1: LH 30; PH 45)

Learning Outcomes

At the end of this course, students should be able to:

1. identify different programming paradigms and their approaches to programming;

2. write programmes using basic data types and strings;

3. design and implement programming problems using selection;

4. design and implement programming problems using loops;

5. use and implement classes as data abstractions in an object-oriented approach;

6. implement simple exception handling in programmes;

7. develop programmes with input/output from text files; and

8. design and implement programming problems involving arrays.

Course Contents

Introduction to computer programming. Functional programming; Declarative programming; Logic programming; Scripting languages. Introduction to object-orientation as a technique for modelling computation. Introduction of a typical object-oriented language, such as Java. Basic data types, variables, expressions, assignment statements and operators. Basic object oriented concepts: abstraction; objects; classes; methods; parameter passing; encapsulation. Introduction to Strings and string processing; Simple I/O; control structures; Arrays; Simple recursive algorithms; inheritance; polymorphism. Lab work: Programming assignments involving hands-on practice in the design and implementation of simple algorithms such as finding the average, standard deviation, searching and sorting. Practice in developing and tracing simple recursive algorithms. Developing programmes involving inheritance and polymorphism.

COS 202: Computer Programming II (3 Units C1: LH 30; PH 45)

Learning Outcome

At the end of this course, students should be able to:

1. Demonstrate the principles of good programming and structured programming concepts;

2. Demonstrate string processing, internal searching, sorting, and recursion;

3. Demonstrate the basic use of OOP concepts: classes, objects, inheritance, polymorphism, data abstraction;

4. Apply the tools for developing, compiling, interpreting and debugging programmes; and

5. Demonstrate the use of syntax and data objects, operators. Central flow constructs, objects and classes programming, Arrays, methods, Exceptions, Applets and the Abstract, OLE, Persistence, Window Toolkit.

Course Contents

Review and coverage of advanced object-oriented programming - polymorphism, abstract classes and interfaces. Class hierarchies and programme organisation using packages/namespaces. Use of API – use of iterators/enumerators, List, Stack, Queue from API. Searching, and sorting. Recursive algorithms. Event-driven programming: event-handling methods, event propagation, exception handling. Applications in Graphical User Interface (GUI) programming.

Lab work:

Programming assignments leading to extensive practice in problem-solving and programme development with emphasis on object-orientation. Solving basic problems using 131 static and dynamic data structures. Solving various searching and sorting algorithms using iterative and recursive approaches. GUI programming.

CSC 203: Discrete Structures (3 Units C: LH 45)

Learning Outcomes

At the end of the course, students should be able to:

1. Convert logical statements from informal language to propositional and predicate logic expressions;

2. Describe the strengths and limitations of propositional and predicate logic;

3. Outline the basic structure of each proof technique (direct proof, proof by contradiction, and induction) described in this unit;

4. Apply each of the proof techniques (direct proof, proof by contradiction, and induction) correctly in the construction of a sound argument;

5. Apply the pigeonhole principle in the context of formal proof;

6. Compute permutations and combinations of a set and interpret the meaning in the context of the particular application;

7. map real-world applications to appropriate counting formalisms, such as determining the number of ways to arrange people around a table, subject to constraints on the seating arrangement, or the number of ways to determine certain hands in cards (e.g., a full house); and

8. Solve a variety of basic recurrence relations.

Course Contents

Structured approach to analysis and design of information systems for businesses. Software development life cycle. Structured top-down and bottom-up design. Dataflow diagramming. Entity relationship modelling. Computer aided software engineering. Input and output, prototyping design and validation. File and database design. Design of user interfaces. Comparison of structured and object-oriented design

Lab Work: system requirements gathering techniques;data modelling techniques (entity relationship modelling);process modelling techniques (data flow diagram); use of UML diagrams; system architectural design; user interface design.

MTH 201: Mathematical Methods 1 (2 Units C: LH 30)

Learning Outcomes At the end of the course, students should be able to:

1. understand Real-valued functions of a real variable;

2. solve some problems using Mean value Theorem and Taylor Series expansion; and

3. evaluate Line Integral, Surface Integral and Volume Integrals.

Course Contents

Real-valued functions of a real variable. Review of differentiation and integration and their applications. Mean value theorem. Taylor series. Real-valued functions of two and three variables. Partial derivatives chain rule, extrema, lagrangian multipliers. Increments, differentials, and linear approximations. Evaluation of line, integrals. Multiple integrals.

MTH 202: Mathematical Methods II (2 Units C: LH 30)

Learning Outcomes:

At the end of the course, students should be able to:

1. define the following: order and degree of a differential equation;

2. describe some techniques for solving first and second order linear and non-linear equations; and

3. solve some problems related to geometry and physics.

Course Contents

Derivation of differential equations from primitive, geometry, physics etc. order and degree of differential equation. Techniques for solving first and second order linear and non-linear equations. Solutions of systems of first order linear equations. Finite linear difference equations. Application to geometry and physics.

IFT 211: Digital Logic Design (2 Units C: LH 15; PH 45)

Learning Outcomes:

At the end of this course, students will be able to:

1. explain why everything is data, including instructions, in computers;

2. describe how negative integers, fixed-length numbers, and non-numeric data are represented;

3. convert numerical data from one format to another;

4. describe computations as a system characterised by a known set of configurations with transitions from one unique configuration (state) to another (state);

5. describe the distinction between systems whose output is only a function of their input (combinational) and those with memory/history (sequential);

6. describe a computer as a state machine that interprets machine instructions;

7. articulate that there are many equivalent representations of computer functionality, including logical expressions and gates, and be able to use mathematical expressions to describe the functions of simple combinational and sequential circuits; and

8. design the basic building blocks of a computer: arithmetic-logic unit (gate-level), registers (gate-level), central processing unit (register transfer-level), and memory (register transfer-level).

Course Contents

Introduction to information representation and number systems. Boolean algebra and switching theory. Manipulation and minimisation of completely and incompletely specified Boolean functions. Physical properties of gates: fan-in, fan-out, propagation delay, timing diagrams and tri-state drivers. Combinational circuits design using multiplexers, decoders, comparators and adders. Sequential circuit analysis and design, basic flip-flops, clocking and timing diagrams. Registers, counters, RAMs, ROMs, PLAs, PLDs, and FPGAs. Lab Work: Simple combinational gates (AND, OR, NOT, NAND, NOR); Combinational circuits design using multiplexers, decoders, comparators and

adders. Sequential circuit analysis and design using basic flip-flops (S-R, J-K, D, T flip-flops); Demonstration of registers, counters, RAMs, ROMs, PLAs, PLDs, and FPGAs

IFT 212: Computer Architecture and Organisation (2 Units C: LH 15; PH 45)

Learning Outcomes

At the end of this course, students will be able to:

1. explain the organisation of the classical von Neumann machine and its major functional units;

2. construct simple assembly language programme segments;

3. describe how fundamental high-level programming constructs are implemented at the machine-language level;

4. discuss the concept of control points and the generation of control signals using hardwired or microprogrammed implementations;

5. describe how the use of memory hierarchy (cache, virtual memory) is used to reduce the effective memory latency; and

6. explain the concept of interrupts and describe how they are used to implement I/O control and data transfers.

Course Contents

Principles of computer hardware and instruction set architecture. Internal CPU organisation and implementation. Instruction format and types, memory, and I/O instructions. Dataflow, arithmetic, and flow control instructions, addressing modes, stack operations, and interrupts. Data path and control unit design. RTL, microprogramming and hardwired control. The practice of assembly language programming. Memory hierarchy. Cache memory, Virtual memory. Cache performance. Compiler support for cache performance. I/O organisations.

Lab work: Practical demonstration of the architecture of a typical computer. Illustration of different types of instructions and how they are executed. Simple Assembly Language programming. Demonstration of interrupts. Programming assignments to practice MS-DOS batch programming, Assembly Process, Debugging, Procedures, Keyboard input, Video Output, File and Disk I/O, and Data Structure. Demonstration of Reduced Instruction Set Computers. Illustration of parallel architectures and interconnection networks.

CSC 299: SIWES I (3 Units C: PH 135)

Learning Outcomes

At the end of this training, students should be able to:

1. assess their knowledge of cybersecurity, its application, preventive measures and defence of the cyber environment;
2. explain how a typical cybersecurity unit/department of an organisation operates;
3. describe the various assignments carried out and the skills acquired during the SIWES period; and
4. submit a comprehensive report on the knowledge acquired and the experience gained during the exercise.

Course Contents

Students are attached to private and public organisations for a period of three months during the second year session long break with a view to making them acquire practical experience and to the extent possible, develop skills in all areas of Computer science. Students are supervised during the training period and shall be expected to keep records designed for the purpose of monitoring their performance. They are also expected to submit a report on the experience gained and defend their reports.

SLU-CSC 204: Systems Analysis and Design (3 Units C: LH 30; PH 45)

Learning Outcomes At the end of this course, students should be able to:

1. describe system requirements gathering techniques;

2. explain data modelling technique (entity relationship modelling);

3. explain process modelling technique (data flow diagram);

4. describe system architectural design;

5. describe process and database design; and

6. explain user interface design.

Course Contents

Structured approach to analysis and design of information systems for businesses. Software development life cycle. Structured top-down and bottom-up design. Dataflow diagramming. Entity relationship modelling. Computer aided software engineering. Input and output, prototyping design and validation. File and database design. Design of user interfaces. Comparison of structured and object-oriented design. Lab work: Practical exercises on software development life cycle (SDLC) activities with different case studies. Use of different information systems case studies to apply the knowledge of structured top-down and bottom –up design, dataflow diagram and entity relationship models.

SLU-IFT 207: Information Technology in Bussiness (2 Units C: LH 30)

Learning objectives

At the end of this course, students should be able to:

1. describe the fundamental concept of innovation and business idea

2. describe the system development process

3. describe business feasibility planning

4. understand business model

Course Contents

Fundamental concepts of innovation and business ideas in general. Product development. Business leadership. Digital marketing. Entrepreneurial opportunities in IT. Legal issues and business ethics. New venture creation process. Business feasibility planning. Market research. Business strategy. Business models and business plans. Technical presentations. Report on a successful entrepreneurial outfit.

8.2.3 LEVEL 300 Course

GST 312: Peace and Conflict Resolution (2 Units C: LH 30)

Learning Outcomes At the end of the course, students should be able to:

1. analyse the concepts of peace, conflict, and security;

2. list major forms, types, and root causes of conflict and violence;

3. differentiate between conflict and terrorism;

4. enumerate security and peace building strategies; and

5. describe roles of international organisations, media, and traditional institutions in peace building.

Course Contents

Concepts of Peace, Conflict, and Security in a multi-ethnic nation. Types and Theories of Conflicts: Ethnic, Religious, Economic, Geo-political Conflicts; Structural Conflict Theory, Realist Theory of Conflict, Frustration-Aggression Conflict Theory. Root causes of Conflict and Violence in Africa: Indigene and settlers Phenomenon; Boundaries/border disputes; Political disputes; Ethnic disputes and rivalries; Economic Inequalities; Social disputes; Nationalist Movements and Agitations; Selected Conflict Case Studies – Tiv-Jukun; Zango Kataf, Chieftaincy, and Land disputes, etc. Peace Building, Management of Conflicts and Security: 72 Peace & Human Development. Approaches to

Peace & Conflict Management --- (Religious, Government, Community Leaders, etc.). Elements of Peace Studies and Conflict Resolution: Conflict dynamics assessment Scales: Constructive & Destructive. Justice and Legal framework: Concepts of Social Justice; The Nigeria Legal System. Insurgency and Terrorism. Peace Mediation and Peace Keeping. Peace & Security Council (International, National, and Local levels) Agents of Conflict resolution – Conventions, Treaties Community Policing: Evolution and Imperatives. Alternative Dispute Resolution, ADR. a) Dialogue b). Arbitration, c). Negotiation d). Collaboration, etc. Roles of International Organisations in Conflict Resolution. (a). The United Nations, UN, and its Conflict Resolution Organs. (b). The African Union & Peace Security Council (c). ECOWAS in Peace Keeping. Media and Traditional Institutions in Peace Building. Managing Post-Conflict Situations/Crisis: Refugees. Internally Displaced Persons, IDPs. The role of NGOs in Post-Conflict Situations/Crisis

ENT 312: Venture Creation **(2 Units C: LH 15; PH 45)**

Learning Outcomes

At the end of this course, students, through case study and practical approaches, should be able to:

1. describe the key steps in venture creation;

2. spot opportunities in problems and in high potential sectors regardless of geographical location;

3. state how original products, ideas, and concepts are developed;

;

7. conduct entrepreneurial marketing and e-commerce;

8. apply a wide variety of emerging technological solutions to entrepreneurship; and

9. appreciate why ventures fail due to lack of planning and poor implementation.

Course Contents

Opportunity Identification (Sources of business opportunities in Nigeria, Environmental scanning, Demand and supply gap/unmet needs/market gaps/market research, Unutilised resources, Social and climate conditions, and technology adoption gap). New business development (business planning, market research). Entrepreneurial finance (venture capital, equity finance, microfinance, personal savings, small business investment organisations, and business plan competition). Entrepreneurial marketing and e-commerce (Principles of marketing, customer acquisition & retention, B2B, C2C and B2C models of e-commerce, first mover advantage, e-commerce business models and successful e-commerce companies,). Small business management/family business: Leadership & Management, basic bookkeeping, nature of family business and family business growth model. Negotiation and business communication (Strategy and tactics of negotiation/bargaining, traditional and modern business communication methods). Opportunity

discovery demonstrations (business idea generation presentations, business idea contest, brainstorming sessions, idea pitching). Technological solutions (the concept of market/customer solution, customer solution, and emerging technologies, business applications of new technologies- Artificial Intelligence (AI), Virtual/Mixed Reality (VR), Internet of Things (IoT), Blockchain, Cloud Computing, renewable energy, etc. digital business and e-commerce strategies).

CSC 301: Data Structures (3 Units C: LH 30; PH 45)

Learning Outcomes

At the end of this course, students should be able to:

1. discuss the appropriate use of built-in data structures;

2. apply object-oriented concepts (inheritance, polymorphism, design patterns, etc.) in software design;

3. implement various data structures and their algorithms, and apply them in implementing simple applications;

4. choose the appropriate data structure for modelling a given problem;

5. analyse simple algorithms and determine their efficiency using big-O notation; and

6. apply the knowledge of data structures to other application domains like data compression and memory management.

Course Contents

Primitive types, Arrays, Records Strings and String processing. Data representation in memory,

Stack and Heap allocation, Queues, Trees. Implementation strategies for stack, queues, trees.

Run time storage management; Pointers and References, linked structures.

Lab work: Writing C+/C++ functions to perform practical exercises and implement using the algorithms on arrays, records, string processing, queues, trees, pointers and linked structures.

CSC 308 Operating System (3 Units C: LH 30; PH 45)

Learning Outcomes

At the end of this course, students should be able to:

1. recognise operating system types and structures;

2. describe OS support for processes and threads;

3. recognise CPU scheduling, synchronisation, and deadlock;

4. resolve OS issues related to synchronisation and failure for distributed systems;

5. explain OS support for virtual memory, disk scheduling, I/O, and file systems;

6. identify security and protection issues in computer systems; and

7. use C and Unix commands, examine behaviour and performance of Linux, and develop various system programmes under Linux to make use of OS concepts related to process synchronisation, shared memory, mailboxes, file systems, etc.

Course Contents

Fundamentals of operating systems design and implementation. History and evolution of operating systems. Types of operating systems. Operating system structures. Process management: processes, threads, CPU scheduling, process synchronisation. Memory management and virtual memory. File systems; I/O systems; Security and protection; Distributed systems; Case studies.

Lab work: Practical hands-on engagement to facilitate understanding of the material taught in the course. All the process, memory, file and directory management issues will be demonstrated under the LINUX operating system. Also UNIX commands will be briefly discussed. Alternatively, hands-on exposure may be through the use of operating systems developed for teaching, like TempOS, Nachos, Xinu or MiniOS. Another possibility is through programming exercises that implement and simulate algorithms taught. Simulation of CPU scheduling algorithms, producer-consumer problem, memory allocation algorithms, file organisation techniques, deadlock algorithms and disk scheduling algorithms.

CSC 309: Artificial Intelligence (2Units C: LH 15; PH 45)

Learning Outcomes

At the end of this course, students should be able to:

1. explain Artificial Intelligence(AI) fundamentals, concepts, goals, types, techniques, branches, applications, AI technology and tools;

2. discuss intelligent agents, their performance, examples, faculties, environment and architectures, and determine the characteristics of a given problem that an intelligent system must solve;

3. describe the Turing test and the "Chinese Room" thought experiment, and differentiate between the concepts of optimal reasoning/behaviour and human-like reasoning/behaviour;

4. describe the role of heuristics and the trade-offs among completeness, optimality, time complexity, and space complexity;

5. analyse the types of search and their applications in AI and describe the problem of combinatorial explosion of search space and its consequences;

6. demonstrate knowledge representation, semantic network and frames along with their applicable uses;

7. practice Natural Language Processing, translate a natural language (e.g., English) sentence into a predicate logic statement, convert a logic statement into clause form, apply resolution to a set of logic statements to answer a query; and

8. analyse programming languages for AI and expert systems technology, and employ application domains of AI.

Course Contents

Overview of Artificial Intelligence. History of AI. Goals of AI. AI Technique. Types of AI. Branches and applications of AI. Advantages and Disadvantages. Introduction to Intelligent Agents. Agent Performance, Examples of Agents, Agent Faculties, Rationality, Agent Environment. Agent Architectures. Search. General Classes of AI Search Algorithm Problems. Problem Solving by Search. Types of AI Search Techniques and Strategies. Introduction to the types of problems and techniques in AI. Problem-Solving methods. Major structures used in AI programmes. Knowledge Representation. KR and Reasoning Challenges. KR Languages. Knowledge representation techniques such as predicate logic, non-monotonic logic, and probabilistic reasoning. Semantic Network - types of relationships, semantic network inheritance, types and components. Introduction to Frames. Natural Language Processing (NLP). Introduction to natural language understanding and various syntactic and semantic structures. Introduction to Expert Systems - characteristics, components, types, requirements, technology, development. Programming Languages for AI. Introduction to computer image recognition.

Lab work: Group practical in (i) Turing test practical - Students can act out their own version of the Turing test (iii) Facial recognition practical to aid in teaching students how machine learning works with students simulating a facial recognition algorithm. Practical applications of NLP in groups – (i) Question Answering focuses on building systems that automatically answer the questions asked by humans in a natural language (ii) Spam detection application for detecting unwanted e-mails getting to a user's inbox (iii) Sentiment analysis/opinion mining should be used on the web to analyse the attitude, behaviour, and emotional state of the sender, implemented through a combination of NLP and statistics (iv) Practical exercise of machine translation used to translate text or speech from one natural language to another natural language such as the Google Translator (v) Developing a model to provide word processor software for the spelling correction (vi) Developing a model for speech recognition for converting spoken words into text (vii) Implementing a Chatbot to provide the staff/student's chat services. OR Group Practical exercise on agents and its environment using simulation of a colony of ants foraging for food; model simulating a message between agents; model simulating the flocking behaviour of birds; model to apply standard search algorithm to the classic

search problem of missionaries and cannibals, and how to use communicating agents for searching networks. Some computer AI animation exercises for any branch of AI. Practical exercise on simple robots coupling and programming. Group project of building a lawn robot for trimming grasses, or any simple design and implementation of robotics.

SLU-CSC 322: Computer Science Innovation and New Technologies (2 Units C: LH 30)

Learning Outcomes

At the end of this course, students should be able to:

1. explain business models;

2. identify some entrepreneurial opportunities available in IT;

3. describe business plan and business startup process;

4. explain business feasibility and strategy;

5. explain marketing strategies; and

6. discuss business ethics and legal issues.

Course Contents

Fundamental concepts of innovation and business ideas in general. Product development. Business leadership. Digital marketing. Entrepreneurial opportunities in IT. Legal issues and Business ethics. New venture creation process. Business feasibility planning. Market research. Business strategy. Business models and Business plans. Technical presentations. Report on a successful entrepreneurial outfit.

CSC 399: SIWES II (3 Units C: PH 135)

Learning Outcomes

At the end of this training, students should be able to:

1. explain how a typical cybersecurity firm operates;

2. describe the various assignments carried out and the skills acquired during the SIWES period; and

3. submit a comprehensive report on the knowledge acquired and the experience gained during the exercise.

Course Contents

Students are attached to private and public organisations for a period of three months during the third year session-long break with a view to making them acquire additional practical experience in

all areas of Computer scence. Students are supervised during the training period and shall be expected to keep records designed for the purpose of monitoring their performance. They are also expected to submit a report on the experience gained and defend their reports.

CYB 201: Introduction to Cybersecurity and Strategy (2 Units C: LH 30)

Learning Outcomes

At the end of this course, students should be able to:

1. explain cybersecurity concepts, its methods, elements, and terminologies of cybersecurity -cyber, security, threat, attack, defence, and operations;

2. describe common cyber-attacks and threats, cybersecurity issues, challenges and proffered solutions, and build an enhanced view of main actors of cyberspace and cyber operations;

3. apply the techniques for identifying, detecting, and defending against cybersecurity threats, attacks and protecting information assets;

4. explain the impact of cybersecurity on civil and military institutions, privacy, business and government applications;

5. identify the methods and motives of cybersecurity incident perpetrators, and the countermeasures employed by organisations and agencies to prevent and detect those incidences and software application vulnerabilities; and

6. state the ethical obligations of security professionals, evaluate cybersecurity and national security strategies to the typologies of cyber-attacks that require policy tools and domestic response, and define the cybersecurity requirements and strategies evolving in the face of big risk.

Course Contents

Basic concepts: cyber, security, confidentiality, integrity, availability, authentication, access control, non-repudiation and fault-tolerant methodologies for implementing security. Security policies, best current practices, testing security, and incident response, Risk management, disaster recovery and access control. Basic cryptography and software application vulnerabilities. Evolution of cyber-attacks. Operating system protection mechanisms, intrusion detection systems, basic formal models of security, cryptography, steganography, network and distributed system security, denial of service (and other) attack strategies, worms, viruses, transfer of funds/value across networks, electronic voting, secure applications. Cybersecurity policy and guidelines. Government regulation of information technology. Main actors of cyberspace and cyber operations. Impact of cybersecurity on civil and military institutions, privacy, business and government applications; examination of the dimensions of networks, protocols, operating systems, and associated applications. Methods and motives of cybersecurity incident perpetrators, and the countermeasures employed by organisations and agencies to prevent and detect those incidences. Ethical obligations of security professionals.

Trends and development in cybersecurity. Software application vulnerabilities. Evolution of cybersecurity and national security strategies, requirements to the typologies of cyber-attacks that require policy tools and domestic response. Cybersecurity strategies evolving in the face of big risk. Role of standards and frameworks.

DTS 304: Data Management I (3 Units C: LH 30; PH 45)

Learning Outcomes

At the end of the course the students should be able to:

1. describe the components of a database system and give examples of their use;

2. describe the differences between relational and semi-structured data models;

3. explain and demonstrate the concepts of entity integrity constraint and referential integrity constraint;

4. apply queries, query optimisations and functional dependencies in relational databases;

5. describe properties of normal forms and explain the impact of normalisation on the efficiency of database operations;

6. describe database security and integrity issues and their importance in database design; and

7. explain the concepts of concurrency control and recovery mechanisms in databases.

Course Contents

Information Management Concepts. Information storage & retrieval. Information management applications. Information capture and representation. Analysis and indexing - search, retrieval, information privacy. Integrity and security. Scalability, Efficiency and Effectiveness. Introduction to database systems. Components of database systems. DBMS functions. Database architecture and data independence. Database query language. Conceptual models. Relational data models. Semi-structured data models. Relational theory and languages. Database Design. Database security and integrity. Introduction to query processing and optimisation. Introduction to concurrency and recovery.

Lab work: Practical exercise on information representation, capture, storage and retrieval. Learn how to analyse data and index for easy searching and indexing. Practical on creating database files and models. How to create and use various database designs. How to query the created database. Methods of oncurrency and recovery in database. Learn how to secure the database.

ICT 305: Data Communication Systems and Network (3 Units C: LH 30; PH 45)

Learning Outcomes

At the end of this course, students should be able to:

1. explain data transmission over layered networks;

2. list and explain common internet technologies and protocols; and

3. explain network operating system.

Course Contents

Types and sources of data. Simple communications network. Transmission definitions, one way transmission, half duplex transmission, transmission codes, transmission modes, parallel transmission, serial transmission, bit synchronisation, character synchronisation, synchronous transmission, asynchronous transmission, efficiency of transmission. Introduction to network protocol. Seven Layer ISO-OSI standard protocols and network architecture. Transport protocols, session services protocols, and other protocols. Institute of Electrical and Electronics Engineering 802 standards. Error control and Data Compression: Forward Error Control; error detection methods; parity checking; linear block codes, cyclic redundancy checking; feedback error control, data compression, Huffman coding and dynamic Huffman coding. Local Area Networks: medium access control techniques – Ethernet, token bus and token ring; fibre distributed data interface, metropolitan area network. Peer-to-peer, Client Server. Client-Server Requirements: GUI design standards, interface independence, platform independence, transaction processing, connectivity, reliability, backup, and recovery mechanisms. Features and benefits of major recovery mechanisms. Network OS: (e.g., Novell NetWare, UNIX/LINUX, OS/2 & Windows NT). INTERNET: Definition, architecture, services, internet addressing. Internet protocol, IPv4, IPv6.

Lab Work: Demonstration of simple communications networks. Illustration of applications at the various levels of the OSI model. Demonstration of different types of Local Area Networks (LANs). Illustration of Metropolitan Area Networks. Illustration of Error Detection and Error Correction techniques. Demonstration of Network Operating Systems.

SLU- CSC 303 Java Programming (2 Units C: LH 30; PH 45)

The learning objectives

Java programming course are designed to ensure that students:

1. Gain a solid understanding of Java syntax, data types, and control flow structures.
2. Develop proficiency in object-oriented programming principles including encapsulation, inheritance, and polymorphism.
3. Learn to handle exceptions effectively and implement error-handling strategies.
4. Master the usage of arrays, strings, and the collections framework for data manipulation.
5. Acquire practical skills in file handling, networking, and database connectivity using Java.
6. Explore multithreading concepts and understand how to create concurrent applications.
7. Gain experience in building graphical user interfaces (GUIs) using Swing and event handling mechanisms.

8. Understand the basics of JavaFX for creating modern, interactive user interfaces.
9. Learn best practices in Java programming, including coding standards, performance optimization, and error handling.
10. Apply their knowledge and skills through hands-on coding exercises, assignments, and a final project to develop a complete Java application.

Course Content:

Basics of Java syntax, data types, and control flow, the course progresses to cover object-oriented programming principles such as classes, inheritance, and polymorphism. Exception handling, file I/O, and collections framework are explored, followed by topics like multithreading, networking, GUI programming with Swing, and database connectivity with JDBC. Throughout the course, students engage in practical coding exercises and assignments to reinforce learning, culminating in a final project where they apply their skills to develop a complete Java application. Additionally, the course emphasizes best practices, and performance optimization techniques, and introduces advanced topics like Java 8 features and JavaFX for modern UI development.

SLU-CSC 305: Computer Graphics & Animation (2Units C: LH 15; PH 45)

Learning Outcomes

At the end of this course, students should be able to:

1. Explain the overall workflow and techniques involved in computer animation production.

2. understand the fundamental computer graphics topics including graphics pipeline architecture, transformations, modeling, viewing, shading, and texture mapping.

3. study basic mathematical backgrounds related to computer graphics including linear algebra and geometry.

4. Understand vividly, those computer graphic algorithms (such as object transformation, geometric representation, shading and illumination model, anti-aliasing and Ray tracing).

5. Study 3-D curve and surface algorithms, in rendering, surface and line removal algorithms

6. Explain the abstract mathematical model describing the way colors can be represented, and

7. Understand the methods of computing a digital image of what the virtual camera sees

Course contents

Definition and Concepts of Computer Graphics: Computer Graphics and Applications, Hardware, Software and Display Devices, Graphics Data Structure, Colour Theory, Image Representation.

Geometric Modelling: Basic Line drawing, Mathematics of CG, Curve and Surface Design, Ray Tracing, Texture Mapping. 3D Graphics Rendering: Geometric Transformation, Scan Conversion, Three-dimensional Viewing, 3D Transform and Animation, Hidden Surface Elimination.

LEVEL 400 Courses

COS 409: Research Methodology and Technical Report Writing (2 Units C: LH 30)

Learning Outcomes

At the end of this course, students should be able to:

1. explain research, types, approaches and significance of research;

2. discuss statement of problem, research methods, methodology, research process, criteria and strategy for good research;

3. discuss scientific investigation, problem formulation, and technique of the research problem;

4. discuss the guidelines for constructing questionnaire/schedule, guidelines for successful interviewing, research proposal and research plan; and

5. explain types of reports, technical report writing, procedures and guidelines.

Course Contents

Foundations of Research. Types of Research. Research Approaches. Significance of Research. Research Methods versus Methodology. Research Process. Criteria and Strategy for Good Research. Problems Encountered by Researchers in Nigeria. Principles of Scientific Research. Scientific investigation. Problem formulation. Definition and technique of the Research Problem. Selection of Appropriate Method for Data Collection- Primary Data and Secondary Data. Guidelines for Constructing Questionnaire/Schedule. Guidelines for Successful Interviewing. Difference between Survey and Experiment. Eloping Research Proposal and Research Plan. Formulation of working hypothesis and Testing. Literature review. Procedure for reviewing related relevant studies and referencing cited works. Types of Reports. Technical Report Writing. Layout and mechanics of writing a Research Report. Standard Techniques for Research Documentation. Sampling Design. Different Types of Sample Designs. Steps in Sampling Design. Criteria of Selecting a Sampling Procedure. Methods of analysis. Processing and Analysis of Data Elements/Types of Analysis. Interpretation and Presentation of results. How to prepare References and Bibliography.

CSC 401: Algorithms and Complexity Analysis (2 Units C: LH 30)

Learning Outcomes

At the end of the course, students should be able to:

1. explain the use of big-O, omega, and theta notation to describe the amount of work done by an algorithm,

2. use big-O, omega, and theta notation to give asymptotic upper, lower, and tight bounds on time and space complexity of algorithms,

3. determine the time and space complexity of simple algorithms,

4. deduce recurrence relations that describe the time complexity of recursively defined algorithms,

5. solve elementary recurrence relations,

6. for each of the strategies (brute-force, greedy, divide-and-conquer, recursive backtracking, and dynamic programming), identify a practical example to which it would apply,

7. use pattern matching to analyse substrings, and

8. use numerical approximation to solve mathematical problems, such as finding the roots of a polynomial.

Course Contents

Basic algorithmic analysis. Asymptotic analysis of Upper and average complexity bounds. Standard Complexity Classes. Time and space trade-offs in analysis recursive algorithms. Algorithmic Strategies. Fundamental computing algorithms. Numerical algorithms. Sequential and Binary search algorithms. Sorting algorithms, Binary Search trees. Hash tables. Graphs and their representation.

CSC 402: Ethics and Legal Issues in Computer Science (2 Units C: LH 30)

Learning Outcomes

At the end of the course, students should be able to:

1. state laws and regulations related to ethics;

2. identify and explain relevant codes of ethics for computing practice;

3. identify social and ethical issues in different areas of computing practice;

4. review real-life ethical cases and be able to develop ethical resolutions and policies;

5. explain the consequences of ignoring and non-compliance with ethical provisions; and

6. develop a sound methodology in resolving ethical conflicts and crisis.

Course Contents

Addresses social, ethical, legal and managerial issues in the application of Computer Science to the information technology industry. Through seminars and case studies, human issues confronting Computer Science graduates will be addressed. Topics include managerial and personal ethics, computer security, privacy, software reliability, personal responsibility for the quality of work, intellectual property, environment and health concerns, and fairness in the workplace.

INS 401: Project Management (2 Units C: LH 30)

Learning Outcomes

At the end of this course, students should be able to:

1. describe project management planning;

2. describe project scheduling;

3. explain the management of project resources;

4. discuss project procurement, monitoring, and execution; and

5. explain project communication and time management.

Course Contents

Introduction to Project Management. The Project Management Lifecycle: Project management and systems development or acquisition, The project management context, Technology and techniques to support the project management lifecycle, and Project management processes. Managing Project Teams: Project team planning, Motivating team members, Leadership, power and conflict in project teams, and managing global project teams; Managing Project Communication and enhancing team communication; Project Initiation and Planning. Managing Project Scope: Project initiation, How organisations choose projects, Activities, and Developing the project charter; Managing Project Scheduling: Common problems in project scheduling, and Techniques for project scheduling; Managing Project Resources: Types of resources (human, capital, time), and Techniques for managing resources. Project quality and tools to manage project quality. Managing project risk and tools for managing project risk. Managing Project Procurement: Alternatives to systems development, External acquisition, Outsourcing-domestic and offshore, Steps in the procurement process, and managing the procurement process. Project Execution, Control, and Closure: Managing project execution, monitoring progress, and managing change, Documentation, and communication, and common problems in project execution; Managing Project Control and Closure: Obtaining information, Cost control, Change control, Administrative closure, Personnel closure, Contractual closure, and Project auditing.

CSC 497: Final Year Project I (3 Units C: PH 135)

Learning Outcomes

At the end of this course, students should be able to:

1. identify a researchable project topic in cybersecurity, work on the introductory aspect, under the guidance of the supervisor;

2. review relevant recent literature pertinent to identified problem statement and establish the research gap;

3. acknowledge and reference sources of information used in the research report;

4. conceptualise and design a research methodology to address an identified problem;

5. determine tools for analysing data collected based on research objectives;

6. write a coherent proposal on the research project to be conducted; and

7. orally present the written project proposal.

Course Contents

An independent or group investigation of appropriate software, hardware, communication and networks or IT related problems in Computer Science carried out under the supervision of a lecturer. Before registering, the student must submit a written proposal to the supervisor to review. The proposal should give a brief outline of the project, estimated schedule of completion, and computer resources needed. A formal written report is essential and an oral presentation may also be required.

CSC 498: Final Year Project II (3 Units C:PH 135)

Learning Outcomes

Upon completion of the project, students should be able to:

1. demonstrate technical skills in Computer Science;

2. demonstrate generic transferable skills such as communication and team work;

3. produce a technical report in the chosen project;

4. defend the written project report; and

5. appreciate the art of carrying out full-fledged research.

Course Contents

This is a continuation of CSC 497. This contains the implementation and the evaluation of the project. A formal written report, chapters 4-5 have to be approved by the supervisor. A final report

comprising chapters 1 - 5 will be submitted to the department for final grading. An oral presentation is required.

SLU-CSC 403: Human-Computer Interface (HCI) (2 Units: LH 30)

Learning Outcomes

At the end of this course, students should be able to:

1. discuss the foundations and concept of the human-computer interface;

2. explain the principles of the human-computer interface;

3. explain the design and development of the human-computer interface; and

4. explain the importance of user feedback.

Course Contents

Foundations of HCI, principles of GUI, GUI toolkits; voice interface design, human-centred software evaluation and development; GUI design and programming, user experience.

SLU-CSC 405: Database Design & Management II (3 Units: LH 30; PH 45)

Learning Outcomes

At the end of this course, students should be able to:

1. Define the concepts database, database design, and database management

2. List and explain the importance and characteristics of databases;

3. Identify and explain the types and categories of databases;

4. Explain database design, development, maintenance, and Implementation strategies;

5. Describe SQL, and Data Modelling; normalization

6. List and explain 1NF, 2NF, 3NF, and Storage Management of databases;

Course Contents

Rational Databases: Mapping conceptual schema to relational Schema; Database Query Languages (SQL) Concept of Functional dependencies & Multi-Valued dependencies. Transaction processing; Distributed databases. Text: CJ Date.

SLU-CSC 404: Organization of Programming Languages (3 Units: LH 45)

Learning Outcomes

At the end of this course, students should be able to:

1. To increase the student's capacity to use different constructs

2. To enable students to choose language more intelligently

3. To make learning new languages easier

Course Contents

Language definition structure. Data types and structures, Review of basic data types, including lists and tress, control structure and data flow, Run-time consideration, interpretative languages, lexical analysis and parsing.

SLU-CSC 406: Compiler Construction (3 Units: LH 45)

Learning Outcomes

At the end of this course, students should be able to:

1. Understand the basic techniques used in compiler construction such as lexical analysis, top-down, bottom-up parsing, context-sensitive analysis, and intermediate code generation.

2. Understand the basic data structures used in compiler construction such as abstract syntax trees, symbol tables, three-address code, and stack machines.

3. Design and implement a compiler using a software engineering approach.

4. Use generators (e.g. Lex and Yacc)

Course Contents

Review of compilers assemblers and interpreters, structure and functional aspects of a typical compiler, syntax semantics and, functional relationship between lexical analysis, expression analysis and code generation. Internal form of source program. Use of a standard compiler (FORTRAN).

SLU-CSC 408: Object-Oriented Programming (3 Units C: LH 15; PH 45)

Learning Outcomes

At the end of this course, students should be able to:

1. explain the concept of the object-oriented approach to modelling;

2. describe the conceptual model of the UML-based software development life cycle;

3. demonstrate how to use the major UML diagrams for object-oriented analysis and design;

4. demonstrate the use of UML-based CASE tools.

Course Contents

Basic OOP Concepts: Classes, Objects, inheritance, polymorphism, Data Abstraction, Tools for developing, Compiling, interpreting and debugging, Java Programs, Java Syntax and data objects, operators. Central flow constructs, objects and classes programming, Arrays, methods. Exceptions, Applets and the Abstract, OLE, Persistence, Window Toolkit, Laboratory exercises in an OOP Language.

SLU-CSC 410: Internet of Things (2 Units: LH 20, P 30)

Learning Outcomes:

At the end of this course, students should be able to:

1. structural features and applications of the most common IoT compatible device families;

2. general knowledge about digital interfaces for sensors;

3. basics of programming for ARM Cortex devices (Use libraries);

4. general principles and approaches to debugging and verification of embedded systems.

5. choose the most appropriate class of embedded system to solve a problem;

6. perform modeling, optimizing and debugging for a IoT projects.

7. possess the necessary skills to design IoT compatible embedded devices and systems using computer aided tools and techniques.

Course Contents

Compiling, downloading, and running simple programs on an evaluation board. CAD tools and libraries (HAL/LL) for STM32-based design. C Language Programming, a popular programming language among embedded-system developers. Programming I/O, investigates some of the functions that configure I/O devices, gaining an understanding of what is involved in writing I/O interfaces for other targets. Digital interfaces. Serial interface, UART. I2C, SPI, CAN interfaces – review. Real-Time Signal Processing, introducing to Digital Signal Processing (DSP) and review the ARM Cortex M4 instruction set support for DSP applications. Real-Time Embedded Systems, writing a multithreaded program using flags for communication and ensuring mutual exclusion when accessing shared resources. Session-layer Protocols: MQTT and CoAP. HTTP-based M2M and the Web of Things. The W3C Web of Things.

Lab: A Practical Introduction to ARM® Cortex®; introduce to writing programs in C, Hello world by STM. Pulse generator by LED and Switches. Digital ports GPIO, Timers and PWM – mode. UART, I2CInterface. DMA application using the codec, followed by designing a low-pass filter. Programing of Real-Time Embedded Systems.

Summary of Graduation Requirement for B.Sc. Computer Science

LEVEL	CCMAS 70%				Additional 30%			Required units	
	Specialization	Faculty	General	Total	Core	Elect.	Total	Min	Max
100	-	5	17	22	8	-	8	30	30
200	11	8	4	23	7	2	8	30	32
300	15	6	4	25	6	-	6	31	31
400	12	4	-	16	14	-	14	30	30
Total	38	23	25	86	39	2	39	121 DE (91)	123 DE (93)

.