ASPIRE
COLLEGE

Advanced Diploma in IT & Computing

COURSE CURRICULUM

Table of Contents

1.0 Course Objective

The Advanced Diploma of IT & Computing is a unique program that will prepare you for a well-paying career in the computer science/engineering sector. Focusing on emerging tools and technologies, and offering hands-on work on real industry projects, you'll be equipped for exciting and evolving tech careers across this growing industry.

This 60-credit diploma program provides graduates with diverse computing knowledge to meet the computer and information system needs of organizations of all sizes.

2.0 Course Duration

The 60-credit Advanced Diploma of IT & Computing is a full-time course that is delivered in 4 semesters that include 2400 equivalent instructional hours.

3.0 Admission Requirements

- Secondary School Diploma or equivalent, or a mature applicant
- English: Grade 12 - C, or equivalent course
- Mathematics: Grade 12 - C, or equivalent course

For students whose English is not their first language, English language requirements are outlined as follows:

Test	Required Minimum Scores
TOEFL (Test of English as a Foreign Language)	80 (20L,W,S,R) iBT (Internet-based TOEFL) **or** 550 Paper-based
iTEP Academic**	3.7
IELTS -Academic Module	6 – Overall with no Band below 5.5
PTE A	58 - Overall Minimum Score (49L,W,S,R)
FCE (Cambridge English First)	174 - Cambridge English Scale or FCE B
CAE (Cambridge English: Advanced)	180 - Cambridge English Scale or CAE C
CPE (Cambridge English Proficiency)	200 - Cambridge English Scale or CPE C
CAEL (Canadian Academic English Language)	60 - Overall score with no band below 50

4.0 Course Learning Outcomes

Students who complete the Advanced Diploma in IT & Computing should expect to gain the following competencies:

- Identify, analyze, design, develop, implement, verify and document the requirements for a computing environment.
- Diagnose, troubleshoot, document and monitor technical problems using appropriate methodologies and tools.
- Analyze, design, implement and maintain secure computing environments.
- Analyze, develop and maintain robust computing system solutions through validation testing and industry best practices.
- Communicate and collaborate with team members and stakeholders to ensure effective working relationship.
- Select and apply strategies for personal and professional development to enhance work performance.
- Apply project management principles and tools when responding to requirements and monitoring projects within a computing environment.
- Adhere to ethical, social media, legal, regulatory and economic requirements and/or principles in the development and management of the computing solutions and systems.
- Investigate emerging trends to respond to technical challenges.
- Analyze, plan, design, implement and administer computer systems and cloud solutions.

5.0 Assessments and grading standards

For each course module, the aim is to assess a student's competence in a range of IT and Computing management skills relevant to the course content and learning outcomes.

Students are required to demonstrate an understanding of key concepts, the application of skills to real work environments and an ability to conduct independent research and analysis. Here are the following assessment tools utilized by Aspire College:

Quizzes
Quizzes are meant to test a student's comprehension of all lessons as they progress through the course.

Assignments
Assignments are intended to assess the student's application, analysis, and critical thinking skills in relation to the concepts you learn in the course. One assignment is required for this course

Final Exam
The final exam is a cumulative test designed to ensure that students have mastered the material in the course.

The grading standards used by Aspire College are defined in the following charts and apply to all courses unless otherwise stated on the course outline.

Alpha Grade	Grade Point	Low Range	High Range	Description	Guideline
A+ A	4.00 3.75	90 80	100 89	Achievement in course outcomes is outstanding and warrants unique and distinguished recognition.	Considerable evidence of original thinking; demonstrated capacity to analyze, integrate and extend concepts; outstanding grasp of subject matter; consistent demonstration of practice with no supervision and guidance.
B+ B	3.50 3.00	75 70	79 74	Achievement in course outcomes is acceptable and meets above average standards.	Evidence of solid grasp of subject matter; reasonable understanding of relevant ideas; some evidence of critical capacity and analysis; clear connections of ideas and numerous extensions; reasonable demonstration of practice with minimal supervision and guidance.
C+ C	2.50 2.00	65 60	69 64	Achievement in course outcomes meets average standards.	Evidence of some understanding of subject matter and relevant ideas; ability to develop/apply solutions to simple problems; minimal connections of ideas and extensions; demonstration of practice with periodic supervision and guidance.
D	1.00	55	59	Achievement in course outcomes is marginal and meets minimal standards.	Evidence of minimal understanding of subject matter and relevant ideas; minimal connections of ideas; demonstration of practice with constant supervision and guidance.
F	0.00	0.00	54	Achievement in course outcomes is inadequate and fails to meet minimal standards.	Insufficient evidence of understanding subject matter and relevant ideas; inability to connect and extend ideas; inability to demonstrate practice.

6.0 Program Core Units

SEMESTER 1	CREDIT HOURS	SEMESTER 2	CREDIT HOURS
Introduction to Programming in Java	3	Discrete Mathematics	3
Introduction to Linux	3	Computer Concepts and Applications	3
Introduction to Computing	3	Database Fundamentals	3
Programming in C	3	Programming in C++	3

SEMESTER 3	CREDIT HOURS	SEMESTER 4	CREDIT HOURS
Data Structures and Algorithms	3	Database Programming	3
Programming in Python	3	Network and System Security	3
Introduction to Networking	3	Systems Analysis and Design	3
Introduction to Operating Systems	3		

NOTE: In addition to the Core Units students are required to take at least 5 elective courses to complete 60 credit hours. Students may take more than 5 or all elective courses as per their convenience.

PROGRAM ELECTIVE UNITS

ELECTIVE COURSES	CREDIT
Fundamentals of Information Technology	3
Introduction to Javascript	3
Data Science Ethics	3
Current Trends in Computer Science & IT	3
Introduction to Cybersecurity	3
Building Web Applications	3
Calculus	3
Software Engineering	3
Introduction to Artificial Intelligence	3

7.0 Course Objectives, Learning Outcomes & Course Topics

7.1 Introduction to Computing

Course Objective	The course objective is to understand the fundamentals of computing, including database systems, the Internet, software development and programming.
Learning outcomes	Upon completion of this course, you will be able to: • Summarize and describe the application software used for personal, business, and workgroup use • Break down how software controls the computing environment • Outline and define the components of computer hardware, including input and output devices • Summarize the history of computing, including how computer have impacted society • Define and appraise the different types of database systems and data types • Examine and describe the basics of Internet programming, scripting languages, search engines, and Internet protocols • Summarize the networking options available to interconnect computers and systems • Diagram and evaluate the lifecycle of developing software, such as applications, drivers, or operating systems • Describe and define the five basic elements of programming and what programmers do
Course topics	1. Application Software 2. Systems Software 3. Computer Hardware 4. Social Impacts and History of Computing 5. Data Communications 6. World Wide Web 7. Networks Access and Architecture 8. Software Development 9. Programming Methodology 10. Studying for Business 109

7.2 Fundamentals of Information Technology

Course Objective	The course objective is for you to understand basic IT topics that are essential to computer science such as database systems, the Internet, e-commerce, software development and computer security.
Learning outcomes	Upon completion of this course, you will be able to: • Identify the different elements of an information system, including input, output and processing elements. • Identify the components of a computer system, including hardware components and software applications. • Explain the differences between operating system software and application software. • Describe the purpose and components of database management, knowledge management systems and specialized information systems. • Analyze potential security risks when using an information system, including personal and professional computer systems and networks, and describe forms of computer network and internet security. • Identify jobs related to information systems careers and describe the impact of technology on careers. • Describe the system development life cycle (SDLC), define object-oriented programming, and describe alternate methods of development (e.g., Agile programming).
Course topics	1. Information Systems Overview 2. Computer Hardware & Software 3. Database Systems Overview 4. Networks & the Internet 5. Mobile & Electronic Commerce 6. Information Support Systems 7. Knowledge Management & Artificial Intelligence 8. Software Development & Project Management 9. Impact of Information Technology 10. Computer & Internet Security 11. Information Systems & Technology Careers

7.3 Computer Concepts and Applications

Course Objective	The course objective is to identify and understand computer hardware and software, the Internet, databases, computer security, programming, and system development.
Learning outcomes	Upon completion of this course, you will be able to: • Describe the historical development of computers and explain the impact of computers on work environments • Identify and describe computer systems components and their function • Differentiate between system and application software and provide examples of each • Describe the function of operating systems and identify different types of operating systems • Explain how the World Wide Web works and describe how the internet can solve business problems • Create documents, spreadsheets, and presentations with Microsoft Office applications • Identify components of telecommunications systems and describe types of networks and network architecture • Identify types of data and explain how businesses use big data, data mining, and database management systems • Recognize computer, network, internet security, and privacy threats and explain how they can be addressed by individuals and businesses • Identify and describe systems development life cycles and explain the purpose of systems analysis.
Course topics	1. Basic Computer Concepts 2. Computer Hardware Components & Functions 3. Application & System Software 4. Operating Systems for Computers 5. Telecommunications Systems 6. Understanding Computer Files 7. Troubleshooting Computer Issues 8. Web Browsers & Technology 9. Using Microsoft Word 10. Using Microsoft Excel 11. Using Microsoft PowerPoint 12. Database Fundamentals 13. Computer Programming Fundamentals 14. Systems Development & Analysis 15. Computer Threats & Security 16. Technology's Ethical, Legal & Privacy Issues

7.4 Introduction to Operating Systems

Course Objective	The course objective is to outline the Operating Systems such as the basics of memory management, storage management, file management, networking, computer security risks, virtual machines.
Learning outcomes	Upon completion of this course, you will be able to: • Describe the various operating system models • Describe techniques used by operating systems to manage processes • Describe how hardware is managed by the operating system • Describe how network functions are managed by an operating system • Explain how users are managed • Explain how operating systems manage memory • Describe mass storage systems and explain how they work • Explain file management and file systems within operating systems • Summarize security in operating systems design and use • Describe virtual machines and explain their use
Course topics	1. Operating System Fundamentals 2. Processes, Threads & Concurrency in Operating Systems 3. Basic Memory Management 4. Input/Output & Storage Management 5. File Management in Operating Systems 6. Networking in Operating Systems 7. Computer Security Risks & Policies 8. Virtual Machines

7.5 Introduction to Linux

Course Objective	The course objective is to analyze fundamentals of the Linux operating system, including installation, configuration, administration, file management, and security.
Learning outcomes	Upon completion of this course, you will be able to: • Explain some of the different distribution of Linux and the reason for open source. • Use Linux commands to manage files and file systems. • Create and execute BASH scripts. • Explain the structure of the Linux operating system. • Establish user accounts and permissions. • Configure basic Linux network services.
Course topics	1. Introduction to Linux & Open Source Software 2. Linux Installation & Configuration 3. Linux Filesystems, Commands & Text Editors 4. Linux Filesystem Management 5. Linux Filesystem Administration 6. The Bash Shell in Linux 7. Linux System Initialization & X Windows 8. Managing Linux Processes 9. Administrative Tasks in Linux 10. Linux Servers & Networks

7.6 Database Fundamentals

Course Objective	The objective of this course is to introduce you to the fundamentals of databases by reviewing relational database models and designs, the foundations of SQL (structured query language), database modifications, and business intelligence applications.
Learning outcomes	Upon completion of this course, you will be able to: • Describe the purpose of a database and its structure. • Explain the difference in various database models. • Utilize data modeling tools to design a database. • Use Structured Query Language (SQL) to create a database and manipulate data in a database. • Explain database security and policy development. • Explain the different methods to connect and display data from a database.
Course topics	1. Introduction to Database Fundamentals 2. Relational Database Model 3. Structured Query Language 4. Data Types in SQL 5. Variations of SQL 6. Relational Data Modeling 7. Relational Database Design 8. Creating, Modifying & Deleting Databases 9. Database Administration & Security 10. DBMS & Data Processing Applications 11. Business Intelligence & Data Warehouses 12. Non-Relational Databases

7.7 Introduction to Networking

Course Objective	The course objective is to introduce students to the foundations of computer networking. Students will learn about basic network standards, different network types, topologies, hardware, troubleshooting strategies and security methods.
Learning outcomes	Upon completion of this course, you will be able to: • Define and describe the different types of networks, topologies, and protocols • Explain the OSI model and define each layer and its function: physical, data link, network, transport, session, presentation, application • Explain the TCP/IP model and define services and protocols provided in the model • Define the key components of computer networks, including hardware and software components • Describe network types and their features, including local area networks (LAN) and wide area networks (WAN) • Define wireless networks and explain their function and use, including hardware, software, and standards used • Describe uses and applications for wireless networks, including VLAN; describe options for remote access • Explain the importance of network security, and describe solutions to prevent unwanted access to networks and network resources • Describe the use of voice over IP (VOIP) and video-over-IP, including protocols and quality of service (QoS) assurance.
Course topics	1. Computer Networking Fundamentals 2. Networking Standards & the OSI Model 3. Transmission & Networking Media 4. The Transmission Control Protocol/Internet Protocol Model 5. TCP/IP Mail Services & Network Troubleshooting 6. Network Topologies & Ethernet Standards 7. Network Hardware 8. Wide Area Networks 9. Wireless Networking 10. Virtual Networks 11. Network Security 12. Voice & Video Communication Over IP

7.8 Introduction to Programming in Java

Course Objective	The course objective is to develop core programming skills through the use of Java as the core language. The purpose is to prepare students for subsequent courses in both computer science and other science areas. Students will learn program design, control structures, data types, arrays, algorithms, documentation, testing, debugging; and principles of object-oriented design, including encapsulation, polymorphism, and abstraction.
Learning outcomes	Upon completion of this course, you will be able to: • Design a program using flowcharts and pseudocode • Write a simple program that includes best practices and documentation/comments • Write conditional statements and loops • Define and create arrays • Write statements that display output and gather input • Define functions and create functions in a program • Explain the tenets of object-oriented programming (OOP) and create classes, objects and methods • Create data structures and abstract data types (ADT)
Course topics	1. Computer Programming Basics 2. Built-In Data Types for Programming 3. Conditionals & Loops in Programming 4. Arrays in Programming 5. Input & Output in Programming 6. Functions & Modules in Programming 7. Object-Oriented Programming Overview 8. Algorithmic Analysis, Sorting & Searching 9. Stacks & Queues for Data Structures 10. Tables, Maps & Graphs in Programming

7.9 Cybersecurity

Course Objective	The course objective is to introduce students to the basics of cybersecurity and cybercrime. Students will learn about cybersecurity policies and procedures, the impact of cybercrime on communication, cryptography, operating system and network security, and disaster recovery.
Learning outcomes	Upon completion of this course, you will be able to: • Define the term 'cybersecurity,' the reasons it is necessary to be aware of it, and related significant concepts of the field of cybersecurity. • Define the term 'computer crime' and identify the types of computer crimes that have historically taken place. • Identify the triumvirate of cybersecurity: People, Technology, and Practices • Identify the term 'threat' in terms of characteristics and attributes, and its effect on the triumvirate of cybersecurity. • Describe, according to general principles of remediation how threats can be remedied or even avoided. • Describe prevention and remediation of software-based threats. • Describe prevention and remediation of hardware-based threats. • Describe prevention and remediation of personnel-based threats. • Describe protection of operating system and networking resources
Course topics	1. Introduction to Cybersecurity 2. Basics of Cybercrime 3. The Role of People in Cybercrime 4. Policies & Procedures for Cybersecurity 5. Tools & Techniques for Cybersecurity 6. Impact of Cybercrime on Electronic Communication 7. Cryptography in Cybersecurity 8. Understanding & Preventing Computer Viruses 9. Cybercrime & Software Development 10. Operating System & Network Security 11. Disaster Recovery in Cybersecurity

7.10 Programming in C

Course Objective	The course objective is to provide you with fundamental knowledge of the C programming language. Students will study arrays, strings, characters, input data and more.
Learning outcomes	Upon completion of this course, you will be able to: • Define programming, identify the elements of programming and describe the history and formation of ANSI C • Explain memory usage by different numeric types and create and execute a simple C program • Formulate a program using selection and utilize selection based algorithms • Create a program that utilizes repetition and use looping to solve mathematical problems • Describe useful standard library functions, create functions and declare parameters • Define one-dimensional and multi-dimensional arrays, initialize arrays and pass arrays as function arguments • Use and validate string input and format strings into readable output • Read, write and declare data files in C • Describe the relationship between arrays and pointers and explain how pointers can be manipulated to pass addresses • Create and use structures and work with unions, linked lists, and memory.
Course topics	1. Introduction to Computer Programming Basics 2. Programming Basics 3. Programming Using Selection in C 4. Programming Using Repetition in C 5. Programming Functions in C 6. Arrays, Characters & Strings in C 7. Data Files & Streams in C 8. Arrays, Addresses & Pointers in C 9. Data Structures in C Programming

7.11 Data Science Ethics

Course Objective	This course provides a framework to analyze these concerns as you examine the ethical and privacy implications of collecting and managing big data. As you work through the course, you'll refresh your knowledge of cybersecurity methods, as well as a variety of legal/ethical issues related to technology. When you're finished with the course, you should be able to: 1) Discuss the basics of cybersecurity, data security and network security 2) Explain privacy matters surrounding cyberspace, big data and data mining 3) Recognize national threats against cybersecurity 4) Assess the fundamentals of malware, viruses and cybercrime 5) Describe legal issues in the world of cyberspace 6) Understand the functions of encryption in cybersecurity 7) Address the ethics of technology, technological innovation and the IT profession
Learning outcomes	Upon completion of this course, you will be able to: • Define, explain and analyze “what is ethics?” • History, Concept of Informed Consent • Cyberspace and privacy • Domestic and international security • Legal issues in cyberspace • Technological innovation and ethics • Professional ethics
Course topics	Upon completion of this course, you will be able to: • Define, explain and analyze “what is ethics?” • History, Concept of Informed Consent • Cyberspace and privacy • Domestic and international security • Legal issues in cyberspace • Technological innovation and ethics • Professional ethics

7.12 Building Web Applications

About the course	In this course, you'll explore the basic structure of a web application, and how a web browser interacts with a web server towards the end. You'll also gain an introductory understanding of Hypertext Markup Language (HTML), as well as the basic syntax and data structures of the PHP language. An introduction to Cascading Style Sheets (CSS) will allow you to style markup for webpages. Lastly, you'll gain the skills and knowledge to install and use an integrated PHP/MySQL environment like XAMPP or MAMP.
Learning outcomes Topics	These goals will be met when this course has been completed: • Understand the basics of HTML • Outline HTML lists and elements • Describe how images can be used in HTML • Detail links and tables in HTML • Discuss user interaction with HTML forms • Style web pages with CSS • Identify CSS methods for advanced styling • Explain HTML 5 and web development How It Helps: • **Simplifies your duties:** This course can be used for many training purposes and works whether you have employees at one location or many, on the same schedule or working varied hours. • **Promotes learning:** You'll feel confident your IT teams are prepared to do their jobs correctly when you provide them with this course. • **Encourages understanding:** After finishing this course, your employees will have examined information on HTML links and tables as well as ways in which they are created and managed.
Course topics	1. Introduction to HTML 2. HTML Elements & Lists 3. Images in HTML 4. HTML Links & Tables 5. User Interaction with HTML Forms 6. Styling Web Pages with CSS 7. Advanced Styling Using CSS 8. Web Development & HTML 5 9. PHP and XAMPP

7.13 Programming in C++

Course Objective	The course objective is to cover the fundamentals of C++ programming, providing both theoretical and hands-on experience in developing programs. This course will provide you with the necessary building blocks for building C++ knowledge and skill.
Learning outcomes	Upon completion of this course, you will be able to: • Write, test, compile, and run a procedural C++ program • Explain data types used in C++ • Initialize and assign variables • Develop programs that branch: Use relational, logical, and Boolean logic • Develop programs that use loops: For, while, and Do-While loops • Create arrays and vectors; iterate through arrays; combine loops and vectors • Define strings and the C-style character string; create and modify strings in C++ • Write functions in C++, pass data by value and by reference; overload a function • Define classes, objects, and structs: write classes that group data; develop structs • Define streams in C++: read input from a string with streams; input/output to a file • Describe pointers and memory in C++; allocate and deallocate memory; define memory leaks and ways to avoid them • Create a class that inherits from another class; define polymorphism and encapsulation
Course topics	1. Computer Programming Elements & Basics 2. Programming Basics in C++ 3. Programming Using Branching in C++ 4. Programming Using Loops in C++ 5. Arrays & Vectors in C++ Programming 6. Strings in C++ Programming 7. C++ Programming Functions 8. Classes in C++ Programming 9. File Streams in C++ Programming 10. Pointers & Memory in C++ Programming 11. Inheritance, Polymorphism & Encapsulation in C++ Programming

7.14 Programming in Python

Course Objective	The course objective is to introduce students to the Python programming language. Students will learn the fundamentals of strings, lists, objects, graphics, functions, decision structures, object-oriented programming, data collections, and algorithm design.
Learning outcomes	Upon completion of this course, you will be able to: • Write simple programs in Python • Write programs that perform programmatic numerical computations. • Explain the concept of objects. • Process strings, lists, and files. • Use functions. • Use decision structures. • Use programming loops. • Apply Boolean algebra, expressions, and data types.
Course topics	1. Introduction to Python Programming 2. Computing with Numbers 3. Using Strings, Lists & Files in Python 4. Objects & Graphics in Python 5. Using Functions in Python 6. Decision Structures in Python 7. Iteration & Control Structures in Python 8. Object-Oriented Programming 9. Data Collections in Python 10. Algorithm Design & Data Analysis

7.15 introduction to Javascript

Course Objective	The course objective is to contribute to students' acquisition of skills and knowledge relevant to Computer Programming in Java.
Learning outcomes	Upon completion of this course, you will be able to: • Give an overview of JavaScript • Detail HTML and JavaScript • Outline the language basics of JavaScript • Discuss loops, arrays and conditionals in JavaScript • Identify the functions in JavaScript • Describe JavaScript and forms • Explain how the document object model is used in JavaScript • Define debugging, error handling and events in JavaScript
Course topics	1. JavaScript Overview 2. JavaScript & HTML 3. JavaScript Language Basics 4. Conditionals, Arrays & Loops in JavaScript 5. JavaScript Functions 6. Forms & JavaScript 7. The Document Object Model & JavaScript 8. Error Handling, Debugging & Events in JavaScript

7.16 Data Structures and Algorithms

Course Objective	The objective of this course is to introduce the essentials of data structures using Java-focused data structures. Students will use Java to complete programming exercises and gain hands-on experience with object-oriented design, algorithms, trees, priority queues, text processing, memory management
Learning outcomes	Upon completion of this course, you will be able to: • Define object-oriented principles of design • Design abstract data types • Analyze various algorithms using the Big-O notation • Explain and develop recursive algorithms • Explain how search and sorting algorithms work • Explain how stacks, queues and linked lists work • Explain how trees, hash tables and graphs work • Explain maps, graphs, and text processing • Explain memory management
Course topics	1. Basics of Java 2. Object-Oriented Design Fundamentals 3. Core Data Structures 4. Analyzing Algorithms 5. Recursion & Recursive Algorithms 6. Stacks, Queues & Lists in Java 7. List & Iterator Abstract Data Types 8. Trees in Data Structure 9. Priority Queues in Java 10. Maps & Hash Tables in Data Structures 11. Search Trees & Types 12. Sorting & Selection in Java 13. Text Processing 14. Graph Data Structures 15. Memory Management

7.17 Network and System Security

Course Objective	The course objective is to review the basics of information security, as well as important topics pertaining to network encryption, firewall systems, authentication, auditing and configurations.
Learning outcomes	Upon completion of this course, you will be able to: • Identify and assess current and anticipated security risks and vulnerabilities • Explain a network security plans and policies • Configure and maintain Cisco ASA 5505 to meet requirements outlined in security policies • Explain static address translation • Examine how instruction detection system (IDS) and intrusion prevention system (IPS) can be used for network defense • Implement wireless network security • Protect network from internal external threats • Describe the role of virtual private networks (VPN)
Course topics	1. Introduction to Network Security 2. Information Security Overview 3. Network Encryption 4. Network Firewall Systems 5. Wireless Network Security 6. Web Application Security 7. Network Authentication 8. Network Auditing & Configuration

7.18 Database Programming

Course Objective	The course objective is to teach students the requisite skills they'll need to create and manage relational database management systems. Students will learn how to populate, manipulate, and retrieve data; write queries for multiple tables; modify tables; and use regular expressions.
Learning outcomes	Upon completion of this course, you will be able to: • Define relational databases and explain core concepts of database design (normalization, normal forms, queries, and basic SQL commands) • Explain data definition language, define SQL data types and describe their use, describe constraints, and create primary and foreign keys in a database • Use SQL commands to populate and retrieve data from a database • Define data manipulation language (DML) and explain its use, write SQL statements to update, delete, and manipulate data in a database. • Write SQL commands to retrieve data from multiple tables using joins: inner, left, right, outer, cross, and self joins; define and explain the use of aliases • Write SQL subqueries to query data from multiple tables • Describe views and indexes in databases, Create views and indexes using SQL • Explain how to modify tables in SQL; write SQL commands to drop columns, rows, constraints, tables, temporary tables, and indexes using SQL. • Define sets in databases; write SQL commands to manipulate sets and group data in databases; perform multi-table inserts. • Define regular expressions and explain their use in Oracle, MySQL, and SQL Server • Explain the importance of database security and describe methods for securing databases through management, use SQL tools for database security, and describe auditing and forensic methods for databases.
Course topics	1. Intro to Relational Database Management Systems 2. Creating & Managing Database Tables 3. Populating & Retrieving Data in Databases 4. Manipulating Data in Databases 5. Data Queries in Multiple Tables Using Joins 6. Data Queries in Multiple Tables Using Subqueries 7. Views & Indexes in Databases 8. Modifying Tables in Databases 9. Manipulating Large Data Sets in Databases 10. Developing & Using Regular Expressions in Databases 11. Database Security

7.19 Systems Analysis & Design

Course Objective	The course objective is to understand and analyze the systems development life cycle, system modeling, system implementation, and systems security.
Learning outcomes	Upon completion of this course, you will be able to: • Analyze the duties and activities of a systems analyst. • Explain the purpose and various phases of the systems development life cycle (SDLC). • Demonstrate an understanding of project management. • Assess analysis and design tools and techniques. • Evaluate case studies for real-life aspects of systems analysis and design. • Analyze and use one of the popular systems development processes. • Evaluate system support models and describe methods for securing systems.
Course topics	1. Introduction to Systems Analysis & Design 2. The Systems Development Life Cycle (SDLC) 3. System Modeling & Development 4. System Design & Architecture 5. System Implementation in SDLC 6. Systems Support & Security 7. Project Management in System Design

7.20 Current Trends in Computer Science and IT

Course Objective	The course objective is to teach students about the current trends in various computer science and information technology fields. Students will learn about the fields of application design, database design, data science, artificial intelligence, machine learning, the internet of things, and blockchain technology.
Learning outcomes	Upon completion of this course, you will be able to: • Identify the skills and knowledge of today's IT leaders and describe the role IT leaders play in modern organizations. • Describe current technology trends to improve data management and security. • Describe the evolution of data science and identify emerging trends in data science. • Describe data visualization techniques and trends and analyze how organizations use data visualization to make strategic marketing decisions. • Define artificial intelligence and machine learning, differentiate between the two, and evaluate current trends and practices across different industries. • Define Internet of Things (IoT) and to identify current and future trends in IoT. • Identify trends associated with blockchain technology and evaluate the future impact of blockchain on data security. • Describe the positive and negative socioeconomic impact of technology integration and analyze the ethical challenges facing today's technology leaders.
Course topics	1. Introduction to Systems Analysis & Design 2. Information Technology Leadership & Systems 3. Application Design & Development 4. Database Design & Management 5. Data Science & Analytics 6. Techniques & Trends in Data Visualization 7. Artificial Intelligence Overview 8. Machine Learning Overview 9. Internet of Things Overview 10. Blockchain Technology 11. Ethical, Economical & Social Impact of Current Technology Trends

7.21 Discrete Mathematics

Course Objective	The course objective is to provide students with an overview of discrete mathematics. Students will learn about topics such as logic and proofs, sets and functions, probability, recursion, graph theory, matrices, Boolean algebra and other important discrete math concepts.
Learning outcomes	Upon completion of this course, you will be able to: • Use logical notation • Perform logical proofs • Design and evaluate Euler and Hamilton circuits • Apply recursive functions and solve recurrence relations • Determine equivalent logic expressions • Use graphs and trees • Apply basic and advanced principles of counting • Define sets and sequences. • Calculate discrete probabilities.
Course topics	1. Introduction to Logic & Proofs 2. Sets & Functions in Discrete Math 3. Counting Rules, Combinations & Permutations 4. Discrete Probability 5. Binomial Probability 6. Recursion & Advanced Counting 7. Principles of Graphs & Graph Theory 8. Trees in Discrete Mathematics 9. Matrices in Discrete Math 10. Boolean Algebra & Logic Gates

7.22 Calculus

Course Objective	The course objective is to master the basics of calculus with an emphasis on limits, derivatives and integrals.
Learning outcomes	Upon completion of this course, you will be able to: • Identify continuities and discontinuities in functions and graphs • Define and apply the Intermediate Value Theorem • Determine the limits of functions and use a graph to define limits • Summarize the formal definition of a derivative and appraise graphical representations of derivatives • Calculate derivatives of trigonometric functions, polynomial equations, and exponential equations • Calculate higher order derivatives • Use Newton's Method to find roots of equations • Define the Fundamental Theorem of Calculus • Calculate integrals of trigonometric and exponential functions • Solve integrals using substitution and trigonometric substitution
Course topics	1. Graphing and Functions 2. Continuity 3. Vectors in Calculus 4. Geometry and Trigonometry 5. How to Use a Scientific Calculator 6. Limits 7. Rate of Change 8. Calculating Derivatives and Derivative Rules 9. Graphing Derivatives and L'Hopital's Rule 10. Applications of Derivatives 11. Series 12. Area Under the Curve and Integrals 13. Integration and Integration Techniques 14. Integration Applications 15. Differential Equations

7.23 Software Engineering

Course Objective	The course objective is to teach students an in-depth understanding of software engineering. Students will learn about topics such as process models, development & testing, agile development, requirements modeling, software design & reuse, user interface design, and verification & validation. In addition, students will be working on a personalized software project in this class using Agile methodologies.
Learning outcomes	Upon completion of this course, you will be able to: • Define software engineering, system analysis, and explain the software development life cycle (SDLC) • Explain process models in application development (waterfall, incremental, iterative, agile, etc.) • Describe requirements gathering and requirements modeling tools • Describe and demonstrate user interface design, including best practices and processes • Demonstrate an understanding of quality assurance and critical systems testing • Describe project and software management methodologies in software engineering • Analyze the impact of security and risk management in software engineering • Describe emerging technologies in software engineering • Students would have built a personalized software project
Course topics	1. Introduction to Systems Analysis & Design 2. Introduction to Software Engineering 3. Process Models in Software Engineering 4. Development & Testing Methods in Software Engineering 5. Agile Development in Software Engineering 6. Software Requirements & System Models 7. Software Requirements Engineering 8. Software Design & Software Reuse 9. Software Architectural Design 10. Component-Level Design in Software Engineering 11. User Interface Design in Software Engineering 12. Software Verification & Validation 13. Software Management 14. Emerging Technologies in Software Engineering

7.24 Introduction to Artificial Intelligence

Course Objective	The objective of this course is to teach students the foundations and applications of artificial intelligence. Students will learn about intelligent agents, constraint satisfaction, logical agents, first-order logic, learning, reasoning, and AI trends.
Learning outcomes	Upon completion of this course, you will be able to: • Explain the foundations, history and ethical perspectives of Artificial Intelligence, and the science of agent design. • Illustrate the use of problem-solving techniques, such as the various search methods, gaming and self-learning agents. • Demonstrate AI's use of knowledge representation, through logic agents and first-order logic to address AI problems. • Discuss the philosophical foundations of AI and explain the possibilities for the future of AI • Recognize the different types of artificial intelligent systems in use today and how they are affecting our lives • Distinguish machine code and higher-level computer languages, how they are interpreted and compiled • Define computer security risks, explain how to prevent security breaks and maintain protection • Analyze algorithms and intelligent programs, write pseudocode and intelligent agents • Design decision tree algorithms for data mining using Markov processes, localization, SLAM and LISP • Create neural networks to include image processing, reasoning, and advanced planning
Course topics	1. Introduction to Systems Analysis & Design 2. Fundamentals of Artificial Intelligence 3. Intelligent Agents 4. Using Artificial Intelligence in Searches 5. Constraint Satisfaction in Artificial Intelligence 6. Logical Agents & First-Order Logic 7. Learning & Reasoning in Artificial Intelligence 8. The Present & Future of Artificial Intelligence