| Title | Computer Organization and Assembly Language |
|---|---|
| **Code** | COMP3137 |
| **Credit Hours** | 4(3+1) |
| **Prerequisite** | Digital Electronics and Logic Design |
| **Course Description** | The course aims to understand how device dependent programming can be performed. To understand the programming model of Intel 8086 and above processors. To be able to develop low level programs for an Intel based machine. This course includes detailed discussion on various topics, Comparison of Low Level and High Level Languages, Basic Organization of a Conceptual Computer, Organization of an Intel based Micro Computer, Programming Model of Intel 8086 and above, Memory Organization, Intel Based Computers and Addressing Modes, programs with Interactive Input and Output , utilizing of arithmetic and logical expressions ,Transfer Instructions, Unconditional & Conditional Program Transfer Instructions, Procedure and Recursion, String instruction, Software Interrupt, Intercepting Interrupts, and File operations. |
| **Aims & Objectives** | • Understanding about low and high level language<br>• Understanding about Intel 8086 architecture.<br>• Understanding about Intel processor family<br>• Understanding about MASM (Microsoft assembler)<br>• Writing code in assembly language using MASM<br>• Understanding about File Operations |
| **Learning Outcomes** | • Understanding of the major components of a computer and basics of computer organization<br>• Understanding of interconnection between computer modules and functions<br>• Understanding of the internal working of central processing unit<br>• Understanding of Instruction set architecture. Addressing modes and formats<br>• Understanding of Assembly Language Programming Concepts for intel x86 processor (Data types, Functions Control Structures, Loops, Procedures, Arrays, Pointers, Stack, Strings, Interrupts, file handling)<br>• Understanding of direct Hardware interaction using assembly language |
| **Recommended Text Book** | • Irvine, Assembly Language for Intel-based Computers, 6th end, Prentice Hall. |
| **Reference Books & Material** | • W.Stallings, "Computer Organization & Architecture", 8th ed, Prentice HALL, 2007.<br>• Barry B. Bray, ‐The Intel Microprocessors‖ 8th ed, Pearson, 2009.<br>• The Intel Microprocessors, latest Edition by Barry b. Bray<br>• Assembly Language Programming and Organization of the IBM PC by Yalta Yu.<br>• Computer Organization and Architecture, 9th Edition, William Stallings |
| **Course Content** | **Introduction:**<br>Introduction about computer Organization, assembly language, Comparison of low level and high level languages, Registers types (General) (16-bit): General purpose register ,Special purpose register, Introduction and usage of RAM, Processor, |

| | |
|---|---|
| | Registers, System Bus, Instruction Execution Cycle, Assembly and Machine Language, Assembler, Linker and Link Libraries, Programmer‘s View of a Computer System, RISC and CISC architecture, Physical address calculation, Basic Memory Organization, CPU organization.<br>**Top Level View of Computer Function and Interconnection, Assembler:**<br>Instruction Cycle, Execute Cycle, Interrupts, Interrupt Cycle, Memory Connection, Input / Output Connection, CPU Connection, MASM, MIPS, Defining Data in MASM Assembler<br>**Elements of Assembly Language, Functions:**<br>Integer Constants, Integer Expressions, Real Number Constants, Character Constants, String Constants, Reserved Words, Identifiers, Directives, Instructions, The NOP (No Operation) Instruction, Adding and Subtracting Integer, INC and DEC Instructions, NEG Instruction, how to move integer number in register? Adding and subtracting numbers in registers, Declaration and initialization of variables, moving data from variable to register, Data Definition Statement, BYTE and SBYTE Data, WORD and SWORD Data, Defining DWORD and SDWORD Data, Knowledge about different data types<br>**Operations, Array & Loops**:<br>Division and Multiplication in Assembly, Jumps Based on Specific Flags, Jumps Based on Equality, Simple jump statements, Jumps based on specific condition, Some code examples, Practice on MASM<br>**Procedures, File Operations Procedures:**<br>Labels in procedures, Stack, Runtime Stack, Conditional Control Flow Directives, Compound Expressions, Data Representation & Conversion<br>**Architecture:**<br>Data path, Control unit, critical path, General principles of pipelining, pipelined Y86 implementations |

| **Title** | **Design and Analysis of Algorithms** |
|---|---|
| **Code** | COMP3138 |
| **Credit Hours** | 3(3+0) |
| **Prerequisite** | Data Structure and Analysis of Algorithms |
| **Course Description** | This is a theory course directed at non-theory students with the standard undergrad background. The goal is to survey the key theory topics that every computer science graduate student should know. In about two weeks for each selected topic, we will gain insights into the basics and study one two example in depth. |
| **Aims & Objectives** | The goal of this course is to develop the appropriate background, foundation, and experience for advanced study in Computer Science. Students will develop the necessary skills from both a theoretical perspective as well as applying their knowledge on various problem sets. Particularly, this course will help them in<br>Developing mathematical skills for algorithm design, analysis, evaluation and computational cost,<br>Developing the skills to design and implement efficient programming solutions to |

| | |
|---|---|
| | various problems,<br>Developing data structure techniques for various aspects of programming.<br>At the end of the course, the students should be able to comprehend Design and Analysis of Algorithms and be able to develop efficient optimized algorithms for complex real world problems. |
| **Learning Outcomes** | • Prove the correctness and analyze the running time of the basic algorithms forthose classic problems in various domains;<br>• Apply the algorithms and design techniques to solve problems<br>• Analyze the complexities of various problems in different domains. |
| **Recommended Text Book** | • Introduction to Algorithms, (2nd ed. 2007 MIT Press) by Thomas H. Carmen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein. |
| **Reference Books & Material** | • Data Structures and Algorithm Analysis in C++, Mark Allen Weiss.<br>• Data Structures using C++, Tenanbaum |
| **Course Content** | **Introduction:**<br>Algorithm Design, Data Structures, Efficiency, Analysis of Algorithms, Mathematical Review<br>Mathematical Analysis of Algorithms, types of functions, Order of Growth and Asymptotic Notations<br>**Sorting Algorithms:**<br>Selection Sort algorithm, example and Analysis, Insertion Sort algorithm, Divide and Conquer Algorithms, Merge Sort algorithm, Quick Sort algorithm, Bucket Sort algorithm, Radix Sort algorithm, Counting Sort algorithm,<br>Heap Sort basics, Algorithms, properties and examples, Heap operations and Heap Sort Algorithm analysis, Heap insertion and deletion<br>**Tree based Algorithms and Hashing:**<br>Red Black Tree and Binary Search Tree Basics, Tree Searching Algorithms and Analysis, Analysis of Insertion and Deletion algorithms in BST<br>Hashing Basics, Examples and Hash functions, Analysis of Collision Resolution Techniques in Hashing<br>**Dynamic Programming:**<br>Dynamic Programming – 0-1 Knap Sack, Fractional Knap Sack, Longest Common Subsequence, Shortest Path Finding, Matrix Chain Multiplication, Assembly line chain problem,<br>**Greedy Algorithms:**<br>Prim‘s Algorithm and Kruskal Algorithm, Dijkstra's Algorithm, Huffman Coding<br>**NP-Completeness and Randomized Algorithms:**<br>NP Completeness - Polynomial time verification, Reducibility , NP-Completeness Proofs, Randomized Algorithms—Particle Swarm Optimization, Randomized Algorithms--Genetic Algorithms |

| **Title** | **Parallel & Distributed Computing** |
|---|---|
| **Code** | COMP3139 |

| | |
|---|---|
| **Credit Hours** | 3(3+0) |
| **Prerequisite** | None |
| **Course Description** | This is a theory course directed at non-theory students with the standard undergrad background. The goal is to survey the key theory topics that every computer science graduate student should know. In about two weeks for each selected topic, we will gain insights into the basics and study one two example in depth. |
| **Aims & Objectives** | • The course is an introduction to distributed computing. It discusses the basic principles, concepts and techniques used in distributed systems. The course assumes knowledge of computer communication, Operating Systems, Sockets and Threads Programming.<br>• Provides an undergraduate- level introduction to parallel programming and parallel and distributed systems<br>• Shared-memory parallel computers and distributed-memory multi computers (e.g., clusters) will be studied<br>• Aspects of the practice and research issues in parallelism will be covered<br>• Aim of this course is to equip high quality and ambitious engineering graduates with the necessary advanced technical and professional skills for an enhanced career either in industry or leading-edge research in the area of distributed /grid computing systems |
| **Learning Outcomes** | • Understand and apply theoretical foundations of programming concurrent distributed computing systems<br>• Develop portable programs for parallel or distributed architectures using Message-Passing Interface (MPI)<br>• Analytical modelling and performance of parallel programs and problems with shared memory<br>• Critique research papers on distributed computing over the Internet and identify open research problems |
| **Recommended Text Book** | • Parallel Programming: Techniques and Applications Using Networked Work stations and Parallel Computers‖, Wilkinson and M. Allen, 1stEdition |
| **Reference Books & Material** | • Distributed Systems: Principles and Paradigms, A. S. Tanenbaum and M. V. Steen, Prentice Hall, 2nd Edition, 2007<br>• Distributed and Cloud Computing: Clusters, Grids, Clouds, and the Future Internet, K Hwang, J Dongarra and GC. C. Fox, Elsevier, 1st Ed |
| **Course Content** | **Introduction:**<br>Why use parallel and distributed systems? Speedup and Amdahl's Law. Hardware architectures: multi processors (shared memory), networks of work stations (distributed memory), clusters (latest variation)<br>**Software architectures:** threads and shared memory, processes and message passing, distributed shared memory (DSM), distributed shared data (DSD)<br>**Parallel Algorithms:**<br>Concurrency and synchronization(review), Data and work partitioning, Common parallelization strategies, Granularity, Load balancing, Examples: parallel search, parallel sorting, etc.<br>**Shared-Memory Programming:** |

| | |
|---|---|
| | Threads, P threads, Locks and semaphores24<br>**Distributed-Memory Programming:**<br>Message Passing and Map Reduce, Distributed-Memory Programming, PI, Google's Map Reduce, Hadoop<br>**Other Parallel Programming Systems:**<br>Tread Marks, Distributed shared memory. Aurora: Scoped behavior and abstract data types. S Enterprise: Process templates |

| **Title** | **Artificial Intelligence** |
|---|---|
| **Code** | COMP4114 |
| **Credit Hours** | 4(3+1) |
| **Prerequisite** | None |
| **Course Description** | This course will introduce the basics of artificial intelligence (AI), its scope and application domain. The course will cover topics such as knowledge representation and reasoning formalisms, propositional logic, search methods, learning paradigms, automated reasoning, knowledge based systems, knowledge application and machine learning techniques and some of deep learning techniques etc. |
| **Aims & Objectives** | • To introduce the principles of AI methods.<br>• To equip students with the developments, justifications, implementation, and use of representational, formalism and search methods.<br>• To provide an opportunity to students to learn methods most useful under complex computational uncertain, and vague situations. |
| **Learning Outcomes** | • Enabling students to study advanced courses in the field such as machine learning, neural networks, text and data mining.<br>• Demonstrate the ability to apply AI and Computational Intelligence techniques to a variety of research and application projects. |
| **Recommended Text Book** | • Artificial Intelligence: A modern approach. Russell and Norvig, 3rd edition Pearson Education Series in AI |
| **Reference Books & Material** | • Luger, George & Stubblefield, William, Artificial Intelligence: Structures and Strategies for Complex Problem Solving (6th ed.),<br>• Mathematical Methods in Artificial Intelligence. (Edward A. Bender).<br>• Principals of Artificial Intelligence and Expert Systems Development. (David W. Rolston)<br>• Nils J Nilson, Artificial Intelligence – A New Synthesis, Morgan Kaufman Publishers, Elsevier, USA.<br>• Patrick Henry Winston, Artificial Intelligence, Third Edition, Pearson Education Series in AI. |
| **Course Content** | **Introduction:**<br>History, Applications and Future, Agents and environments, Rationality, Agent Types, Propositional Logic, Predicate Calculus<br>Problem solving through uninformed search methods; State Space Search, Depth First Search, Breath first search, Comparisons.<br>**Problem solving through informed search, Expert Systems:** |

| | |
|---|---|
| | enable them to excel in their academics |
| **Learning Outcomes** | At the completion of the course student will be able to:<br>• Demonstrate improvement in their communicative performance<br>• Self-evaluate their communication skills leading to effective communication |
| **Recommended Text Book** | • Practical Business English, Collen Vawdrey, 1993 |
| **Reference Books & Material** | • Hand book of Communications Skills, Hargie, O. (Ed.)<br>• Mandel, S. 2000. Effective Presentation Skills: A Practical Guide Better Speaking<br>• Communication for Business Success (Canadian Edition) (v.1.0).<br>• Reading and Study Skills by John Langan<br>• Study Skills by Riachard Yorky<br>• Improve Your Communication Skills. London: KoganPage, Barker, A,2003<br>• Practical Business English, Collen Vawdrey |
| **Course Content** | Definition & types of communication (verbal & non-verbal), Components of communication, Barriers in Effective Communication, Listening Skills: Listening to individuals, listening strategies in group discussion, listening news reports, speeches etc. and getting the gist. Speaking Skills: Presentations, Formal and informal Conversation, Interviews and strategies to make interview successful. Reading Skills: Skimming, Scanning, Intensive and Extensive Reading, Reading short stories, comics and excerpts. Writing Skills: Writing applications, official letters, resume, precis writing, Changing narration-converting a story into a news report etc., Writing report/story by looking at an image |

| **Title** | **Technical Writing and Presentation Skills** |
|---|---|
| **Code** | ENGL2115 |
| **Credit Hours** | 3(3+0) |
| **Prerequisite** | None |
| **Course Description** | The course is designed to help students clearly convey the right information in their reports, elicit the business response they seek. |
| **Aims & Objectives** | To understand the stages of written communication in a formal setting. |
| **Learning Outcomes** | • Understand fundamental structures of technical/business formats<br>• Compose grammatically/technically correct academic/business writing<br>• Develop their written business communication |
| **Recommended Text Book** | • Technical Report Writing, by Pauley and Riordan, Houghton Mifflin Company, 8th Edition |
| **Reference Books & Material** | • Effective Technical Communication by Ashraf Rizvi, Tata McGraw-Hill |

| | |
|---|---|
| **Course Content** | **Introduction:**<br>Business Communication Employment interviews (types of interviews, preparation for interviews, follow up communication in interviews)<br>**Business Correspondence:**<br>Formal Letters (Letters of inquiry, order, complaint, invitation, business request), Formal Letters Formal Letters (Letters of inquiry, order, complaint, invitation, business request, Replies to queries and requests) Communication in team (group work business project)<br>**Memoranda and Proposal Writing:**<br>Memoranda Introduction and practice Proposal Writing Making Business proposal through critical thinking and research Making Business proposal through critical thinking and research<br>**Presentations Skills**:<br>Presentations (How to prepare a presentation) Presentations (How to present a business proposal) Presentations (Do's and Don'ts of a good presentation), Practical Group Project |

| **Title** | **Professional Practices in IT** |
|---|---|
| **Code** | ITEC4141 |
| **Credit Hours** | 3(3+0) |
| **Prerequisite** | None |
| **Course Description** | This course aims to develop student understanding about historical, social, economic, ethical, and professional issues relevant to computing and opinion about professionalism and ethics. |
| **Aims and Objectives** | The main objectives of this course are to:<br>• Determine and identify ethical procedures and behaviours in the organization.<br>• Provide the information about computer and internet crimes.<br>• Enable understanding of the issues related to intellectual freedom, intellectual property, and copyright law.<br>• Enable understanding the ethical issues associated with gathering, storing and accessing information in databases. |
| **Learning Outcomes** | At the completion of the course, students will be able to:<br>• Understand and appreciate the meaning of ethics, values and attitudes.<br>• Be guided in their ethical thinking and considerations as they relate in the cyber world.<br>• Be aware of the different ethical dilemma/issues in the cyber world. Appreciate and internalize the code of conduct of an I. T. Professional.<br>• Be familiar with the various Laws that penalizes cybercrimes |
| **Recommended Text Book** | Professional Issues in Software Engineering by Frank Bott, Allison Coleman, Jack Eaton and Diane Rowland, CRC Press; 3rd Edition (2000). |
| **Reference Books** | • Computer Ethics by Deborah G. Johnson, Pearson; 4th Edition (January 3,2009).<br>• A Gift of Fire: Social, Legal, and Ethical Issues for Computing and the Internet |

| Title | Multivariable Calculus |
|---|---|
| **Code** | MATH3122 |
| **Credit Hours** | 3(3+0) |
| **Prerequisite** | Calculus - I |
| **Course Description and objectives** | This course has been designed to provide foundation and basic ground for multivariable calculus and analytical background with multi variables |
| **Learning Outcomes** | At the completion of the course student will be able to:<br>• Understand the basic concepts and know the basic techniques of differential and integral calculus of functions of several variables<br>• Apply the theory to calculate the gradients, directional derivatives, arc length of curves, area of surfaces, and volume of solids<br>• Solve problems involving maxima and minima, line integral and surface integral, and vector calculus |
| **Recommended Text Book** | • Multivariable Calculus, 6th edition James, Stewart 2007 Cengage Learning publishers.<br>• Calculus and Analytical Geometry, 6th edition. Swokowski, Olinick and Pence.1994. Thomson Learning EMEA, Ltd. 3.<br>• Multivariable Calculus, 5th edition Howard, A. Albert, H. 1995, John Wiley. |
| **Course Content** | Functions of Several Variables and Partial Differentiation. Multiple Integrals, Line and Surface Integrals. Green‘s and Stoke‘s Theorem. Fourier Series: periodic functions, Functions of any period P-2L, Even & odd functions, Half Range expansions, Fourier Transform; Laplace Transform, Z-Transform |

| Title | Numerical Computing |
|---|---|
| **Code** | MATH1122 |
| **Credit hours** | 3(3+0) |
| **Prerequisite** | Calculus - I |
| **Course Description and objectives** | This course has been designed to provide a systematic introduction to aspects of numerical methods as they apply to Computer Science problems and structure, focusing on providing a solid theoretical foundation for further work. In this course more emphasis shall be given to statistical and probabilistic formulation with respect to computing aspects. |
| **Learning Outcomes** | At the completion of the course student will be able to:<br>• The student would understand the fundamental concepts of Scientific Programming using programing Language(s)<br>• Use a computer algebra system to investigate and solve mathematical problems relating to integration, differential equations and approximation |
| **Recommended Text Book** | • Numerical Analysis (9th Edition) by Richard L. Burden, J. Douglas Faires by Brooks/Cole Boston USA, 2011<br>• Numerical Methods for Scientific Computing by J.H. Heinbockel Trafford |

| | |
|---|---|
| | Concept of classification, Decision tree approach, Naïve Bays Approach, Accuracy of the system? Bayesian classification, Rule-based classification, Classification by back propagation,<br>**Clustering:**<br>Basic terminology, Partitioning methods: K-Means, K-Medoids, CALARANS Hierarchical methods: BIRCH, CURE etc Density based methods: DBSCAN, OPTICS Grid based methods: STING, Wave Cluster, CLIQUE<br>**Evaluation of Classifiers:**<br>Classification with Train, Test, and Validation sets Evaluation Criteria, Metrics for Performance, Accuracy, Cost Matrix, Cost Sensitive Measures, Methods for Performance Evaluation (Holdout, Random Sampling, Cross Validation, Stratified sampling, Bootstrap) ROC Curve<br>**Mining Complex Data Types and Applications:**<br>Mining text databases, mining the Web, mining time-series and sequence datasets. Targeted Marketing and Customer Modeling<br>Data Mining and Society; Future Directions |

| **Title** | **Introduction to Data Science** |
|---|---|
| **Code** | COMP3115 |
| **Credit hours** | 3(3+0) |
| **Prerequisite** | None |
| **Course Description and objectives** | This course has been designed to provide an introduction about data science, its application and needs in today‘s world. This course also introduces the students about cutting-edge tools used by data scientists around the world. These tools include Python, Numpy, Pandas, Jupyter Notebooks, Scikit learn and a bunch more. The aim is not to just introduce data analysis but to make students aware how they can apply it in real world to make an impact. |
| **Learning Outcomes** | At the completion of the course student will be able to:<br>• Find useful datasets, for research questions<br>• Perform data analysis to help answer the research questions and present its findings |
| **Recommended Text Book** | • Data Science from Scratch by Joel Grus<br>• Python for Data Analysis |
| **Course Content** | **Introduction:**<br>Welcome and overview of the course. Introduction to data science and value of learning data science, Big Data, Modern data science skills, Why Python<br>Data Science Life cycle & Process (Asking Right Questions, Obtaining Data, Understanding Data, Building Predictive Models, Generating Visualizations) For Building Data Products, Introduction to Data (Types of Data and Datasets), Data Quality (Measurement and Data Collection Issues)<br>**Introduction to Python:**<br>Python basics, variables, data types, objects, loops, conditions, functions, string functions, lists, tuples, dictionaries, sets<br>Jupiter: the most used tool in data science, key features, getting started, documenting with markdown text, Numpy for data analysis, time and space efficient functions, and array basics.<br>Pandas data analysis functionality, benefits, pandas data structure, pandas series, pandas data frame, descriptive statistics functions: describe(), corr(), min(), max(), mode(), median(), data cleaning with pandas, merging data frames, frequent string |

| | |
|---|---|
| | operations: split(), contains(), extract()<br>**Data visualization and analysis:**<br>Plotting with pandas, bar chart, box plot, histogram, Data analysis, frequent data operations, Data analysis in detail using Jupyter Notebooks, numpy, pandas etc, data visualization, Matplotlib Library, Relational Algebra & SQL, Scraping & Data Wrangling (assessing, structuring, cleaning & munging of data), Basic Descriptive & Exploratory Data Analysis, Introduction to Text Analysis (Stemming, Lemmatization, Bag of Words, TF-IDF)<br>**Machine learning:**<br>Introduction to Prediction and Inference (Supervised & Unsupervised) Algorithms, Introduction to Scikit Learn, Bias-Variance Tradeoff, Model Evaluation & Performance Metrics (Accuracy, Contingency Matrix, Precision-Recall, F-1 Score, Lift, etc.) |

| **Title** | **Object Oriented Analysis and Design** |
|---|---|
| **Code** | COMP3117 |
| **Credit Hours** | 3(3+0) |
| **Prerequisite** | None |
| **Course Description and objectives** | The object-oriented paradigm provides solutions to many development problems and provides a reliable design, complexity control, and reusability etc. Object-Oriented Analysis and Design includes the overall goals of the object paradigm, the selection of classes, the relationships among them, and their realization to implement systems. This course has been designed to provide:<br>• Introduction basic concepts of the object oriented paradigm, elicitation of requirements through use case<br>• Identification of domain concepts through domain model<br>• selection of classes and assigning roles and responsibilities to various classes and identification of relationships among them, and their realization to implement systems.<br>• Analysis and application a range of concepts, principles, and practices of the subject in the context of loosely specified problems Appreciation of the interplay between theory and practice.<br>• Hands on experience on various tools Enterprise Architect/ Papyrus/ Rational Software Architect to create and manipulate UML Models. |
| **Learning Outcomes** | At the completion of the course student will be able to:<br>• Critically analyze and apply a range of concepts, principles, and practices of the subject in the context of loosely specified problems, showing effective judgment in the selection and use of tools and techniques.<br>• Produce work involving problem identification, analysis, design, and development of a software system, along with appropriate documentation showing problem-solving and evaluation skills using supporting evidence and demonstrate quality.<br>• Gain significant project experience<br>• Demonstrate knowledge and understanding of essential facts, concepts, principles, and theories and apply them in the context of well-defined scenarios, showing judgment in the selection and application of tools and techniques.<br>• Gain understanding of unification and RUP |