# GUJARAT TECHNOLOGICAL UNIVERSITY

## English for Research Paper Writing
## SUBJECT CODE: 3700001
## SEMESTER: I/II

**Type of course: Audit course**

**Prerequisite: -**

**Rationale: -**

**Teaching and Examination Scheme**:

| Teaching Scheme | | | Credits | Examination Marks | | | | Total Marks |
|---|---|---|---|---|---|---|---|---|
| | | | | Theory Marks | | Practical Marks | | |
| L | T | P | C | ESE(E) | PA (M) | PA (V) | PA (I) | |
| 2 | 0 | 0 | 0 | 50 | 0 | 0 | 0 | 50 |

**Content**

| Sl. No. | Topic | Teaching Hours | Module Weightage (%) |
|---|---|---|---|
| 1. | Planning and Preparation, Word Order, Breaking up long sentences, Structuring Paragraphs and Sentences, Being Concise and Removing Redundancy, Avoiding Ambiguity and Vagueness | 4 | 17 |
| 2. | Clarifying Who Did What, Highlighting Your Findings, Hedging and Criticising, Paraphrasing and Plagiarism, Sections of a Paper, Abstracts. Introduction | 4 | 17 |
| 3. | Review of the Literature, Methods, Results, Discussion, Conclusions, The Final Check | 4 | 17 |
| 4. | key skills are needed when writing a Title, key skills are needed when writing an Abstract, key skills are needed when writing an Introduction, skills needed when writing a Review of the Literature | 4 | 17 |
| 5. | skills are needed when writing the Methods, skills needed when writing the Results, skills are needed when writing the Discussion, skills are needed when writing the Conclusions | 4 | 16 |
| 6 | useful phrases, how to ensure paper is as good as it could possibly be the first- time submission | 4 | 16 |

**Reference Books:**


1. Goldbort R (2006) Writing for Science, Yale University Press (available on Google Books)
2. Day R (2006) How to Write and Publish a Scientific Paper, Cambridge University Press
3. Highman N (1998), Handbook of Writing for the Mathematical Sciences, SIAM. Highman'sbook
4. Adrian Wallwork , English for Writing Research Papers, Springer New York Dordrecht Heidelberg London, 2011

**Course Outcome:**

At the end of the course, the student will be able to:

1. Understand that how to improve your writing skills and level of readability
2. Learn about what to write in each section
3. Understand the skills needed when writing a Title
4. Ensure the good quality of paper at very first-time submission

# GUJARAT TECHNOLOGICAL UNIVERSITY

## Constitution of India
## SUBJECT CODE: 3700005
## SEMESTER: I/II

**Type of course: Audit course**

**Prerequisite: -**

**Rationale: -**

**Teaching and Examination Scheme**:

| Teaching Scheme | | | Credits | Examination Marks | | | | Total Marks |
|---|---|---|---|---|---|---|---|---|
| | | | | Theory Marks | | Practical Marks | | |
| L | T | P | C | ESE(E) | PA (M) | PA (V) | PA (I) | |
| 2 | 0 | 0 | 0 | 50 | 0 | 0 | 0 | 50 |

**Content**

| Sl. No. | Topic | Teaching Hours | Module Weightage (%) |
|---|---|---|---|
| 1. | **History of Making of the Indian Constitution**<br>History<br>Drafting Committee, ( Composition & Working) | 4 | 17 |
| 2. | **Philosophy of the Indian Constitution:**<br>Preamble<br>Salient Features | 4 | 17 |
| 3. | **Contours of Constitutional Rights & Duties:**<br>Fundamental Rights<br>Right to Equality<br>Right to Freedom<br>Right against Exploitation<br>Right to Freedom of Religion<br>Cultural and Educational Rights<br>Right to Constitutional Remedies<br>Directive Principles of State Policy<br>Fundamental Duties. | 4 | 17 |
| 4. | **Organs of Governance:**<br>Parliament<br>Composition<br>Qualifications and Disqualifications<br>Powers and Functions<br>Executive<br>President<br>Governor<br>Council of Ministers<br>Judiciary, Appointment and Transfer of Judges, Qualifications<br>Powers and Functions | 4 | 17 |

| | | | |
|---|---|---|---|
| 5. | **Local Administration:**<br>District's Administration head: Role and Importance,<br>Municipalities: Introduction, Mayor and role of Elected Representative,<br>CEO of Municipal Corporation.<br>Pachayati raj: Introduction, PRI: ZilaPachayat.<br>Elected officials and their roles, CEO ZilaPachayat: Position and role.<br>Block level: Organizational Hierarchy (Different departments),<br>Village level: Role of Elected and Appointed officials,<br>Importance of grass root democracy | 4 | 16 |
| 6 | **Election Commission:**<br>Election Commission: Role and Functioning.<br>Chief Election Commissioner and Election Commissioners.<br>State Election Commission: Role and Functioning.<br>Institute and Bodies for the welfare of SC/ST/OBC and women. | 4 | 16 |

**Reference Books:**


1. The Constitution of India, 1950 (Bare Act), Government Publication.
2. Dr. S. N. Busi, Dr. B. R. Ambedkar framing of Indian Constitution, 1st Edition, 2015.
3. M. P. Jain, Indian Constitution Law, 7th Edn., Lexis Nexis, 2014.


**Course Outcome:**

At the end of the course, the student will be able to:

1. Understand the premises informing the twin themes of liberty and freedom from a civil rights perspective
2. To address the growth of Indian opinion regarding modern Indian intellectuals constitutional role and entitlement to civil and economic rights as well as the emergence of nationhood in the early years of Indian nationalism
3. To address the role of socialism in India after the commencement of the Bolshevik Revolution in 1917 and its impact on the initial drafting of the Indian Constitution.

**GUJARAT TECHNOLOGICAL UNIVERSITY**
**Master of Engineering**
**Subject Code - 3720001**
**Semester II**
**Subject Name: Mini Project with Seminar**

**Type of course: Core**

**Teaching and Examination Scheme:**

| Teaching Scheme | | | Credits | Examination Marks | | | | Total Marks |
|---|---|---|---|---|---|---|---|---|
| | | | | Theory Marks | | Practical Marks | | |
| L | T | P | C | ESE (E) | PA (M) | ESE (V) | PA (I) | |
| 0 | 0 | 4 | 2 | 0 | 0 | 0 | 100 | 100 |

**Content:**

A mini project requires comparatively less time than major projects. They are comparatively simpler and have shorter duration. Mini Project helps students to explore and strengthen the understanding of fundamentals through practical application of theoretical concepts. Mini Project can help them to boost their skills and widen their horizon of thinking. It will act like a beginners guide to undertake the major project/dissertation during the final year and will ensure preparedness of students to undertake major projects/dissertation. Students will be required to select the topic relevant to their specialization and that has value addition. Students will get an opportunity to work in actual industrial environment if they opt for internship. Based on the selected topic student will also prepare seminar report based on the literature survey

Mini Project will have mid semester presentation and end semester presentation. Mid semester presentation will include identification of the problem based on the literature review on the topic referring to latest literature available. End semester presentation should be done along with the report on identification of topic for the work and the methodology adopted involving scientific research, collection and analysis of data, determining solutions highlighting individuals' contribution. Continuous assessment of Mini Project at Mid Sem and End Sem will be monitored by the departmental committee.

**Course Outcomes**: At the end of the course, the student will be able to:

1. Identify engineering problems reviewing available literature.
2. Study different techniques used to analyze complex systems.
3. Solve a live problem using software/analytical/computational tools and present solution by using his/her technique applying engineering principles.
4. Learn to write technical reports and develop skills to present and defend their work in front of technically qualified audience.

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Master of Engineering**
**Subject Code: 3720216**
**Semester – II**
**Subject Name: Advance Algorithms**

**Type of course: Regular**

**Prerequisite: UG level course in Algorithm Design and Analysis**

**Rationale:** This course will cover fundamental algorithms that operate on common data structures, for instance sorting and searching; advanced design and analysis techniques; advanced graph matching algorithms including minimum spanning trees and shortest paths; flow networks; and linear programming. In summary, this course will provide exposure to recent trends in problem solving paradigms.

**Teaching and Examination Scheme:**

| Teaching Scheme | | | Credits | Examination Marks | | | | Total Marks |
|---|---|---|---|---|---|---|---|---|
| | | | | Theory Marks | | Practical Marks | | |
| L | T | P | C | ESE (E) | PA (M) | ESE (V) | PA (I) | |
| 3 | 0 | 2 | 4 | 70 | 30 | 30 | 20 | 150 |

**Content:**

| Sr. No. | Content | Total Hrs |
|---|---|---|
| 1 | **Sorting:** Review of various sorting algorithms, topological sorting **Graph:** Definitions and Elementary Algorithms: Shortest path by BFS, shortest path in edge-weighted case (Dijkasra's), depth-first search and computation of strongly connected components, emphasis on correctness proof of the algorithm and time/space analysis, example of amortized analysis. | 6 |
| 2 | **Matroids:** Introduction to greedy paradigm, algorithm to compute a maximum weight maximal independent set. Application to MST.<br>**Graph Matching:** Algorithm to compute maximum matching. Characterization of maximum matching by augmenting paths, Edmond's Blossom algorithm to compute augmenting path. | 8 |
| 3 | **Flow-Networks:** Maxflow-mincut theorem, Ford-Fulkerson Method to compute maximum flow, Edmond-Karp maximum-flow algorithm.<br>**Matrix Computations:** Strassen's algorithm and introduction to divide and conquer paradigm, inverse of a triangular matrix, relation between the time complexities of basic matrix operations, LUP-decomposition. | 9 |
| 4 | **Shortest Path in Graphs:** Floyd-Warshall algorithm and introduction to dynamic programming paradigm. More examples of dynamic programming.<br>**Modulo Representation of integers/polynomials:** Chinese Remainder Theorem, Conversion between base-representation and modulo-representation. Extension to polynomials. Application: Interpolation problem. | 10 |

| | | |
|---|---|---|
| | **Discrete Fourier Transform (DFT):** In complex field, DFT in modulo ring. Fast Fourier Transform algorithm. Schonhage-Strassen Integer Multiplication algorithm | |
| **5** | **Linear Programming:** Geometry of the feasibility region and Simplex Algorithm<br>**NP-completeness:** Examples, proof of NP-hardness and NP-completeness.<br>**One or more of the following topics based on time and interest** Approximation algorithms, Randomized Algorithms, Interior Point Method, Advanced Number Theoretic Algorithm | **10** |
| **6** | Recent Trends in problem solving paradigms using recent searching and sorting techniques by applying recently proposed data structures. | **5** |

**Reference Books:**

1. "Introduction to Algorithms" by Cormen, Leiserson, Rivest, Stein.
2. "The Design and Analysis of Computer Algorithms" by Aho, Hopcroft, Ullman.
3. "Algorithm Design" by Kleinberg and Tardos

4. “Fundamentals of Algorithmics” by Gilles Brassard and Paul Bratley.

**Course Outcomes:**

| Sr. No. | CO statement | Marks % weightage |
|---|---|---|
| CO-1 | Analyze the time complexity/performance of different algorithms. | **20%** |
| CO-2 | Determining the appropriate data structure for solving a particular set of problem. | **20%** |
| CO-3 | Categorize the different problems in various classes according to their complexity. | **30%** |
| CO-4 | Insight of recent activities in the field of the advanced data structure. | **30%** |

**List of Experiments:**

- Minimum 10 experiments based on the above contents.
- Mini Project in a group of max. 3 students
- Writing a research paper on selected topic from content with latest research issues in that topic

**Major Equipments:**

- Latest PCs with related software

**List of Open Source Software/learning website:**

- https://www.coursera.org/specializations/algorithms
- https://visualgo.net/bn
- https://online.stanford.edu/courses/cs161-design-and-analysis-algorithms

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Master of Engineering**
**Subject Code: 3720217**
**Semester – II**
**Subject Name: Soft Computing**

**Type of course:** Core Course

**Prerequisite:** Basic Knowledge of Mathematics

**Rationale:** To introduce the soft computing concepts and techniques and to foster their abilities in designing appropriate technique for a given scenario. To implement soft computing based solutions for real world problems. T give students knowledge about non-traditional techniques and fundamentals of artificial neural networks, fuzzy logic and genetic algorithms. To provide students hands-on experience on MATLAB to implement various strategies.

**Teaching and Examination Scheme:**

| Teaching Scheme | | | Credits | Examination Marks | | | | Total Marks |
|---|---|---|---|---|---|---|---|---|
| L | T | P | C | Theory Marks | | Practical Marks | | |
| | | | | ESE (E) | PA (M) | ESE (V) | PA (I) | |
| 3 | 0 | 2 | 4 | 70 | 30 | 30 | 20 | 150 |

**Content:**

| Sr. No. | Content | Total Hrs | % Weightage |
|---|---|---|---|
| 1 | **Unit 1 : INTRODUCTION TO SOFT COMPUTING AND NEURAL NETWORKS:** Evolution of Computing: Soft Computing Constituents, From Conventional AI to Computational Intelligence: Machine Learning Basics | 7 | 14 |
| 2 | **Unit 2: FUZZY LOGIC:** Fuzzy Sets, Operations on Fuzzy Sets, Fuzzy Relations, Membership Functions: Fuzzy Rules and Fuzzy Reasoning, Fuzzy Inference Systems, Fuzzy Expert Systems, Fuzzy Decision Making. | 9 | 20 |
| 3 | **Unit 3: NEURAL NETWORKS:** Machine Learning Using Neural Network, Adaptive Networks, Feed forward Networks, Supervised Learning Neural Networks, Radial Basis Function Networks : Reinforcement Learning, Unsupervised Learning Neural Networks, Adaptive Resonance architectures, Advances in Neural networks | 10 | 20 |
| 4 | **Unit 4: GENETIC ALGORITHMS:** Goals of optimization, comparison with traditional methods, schemata, Terminology in GA – strings, structure, parameter string, data structures, operators, coding fitness function, algorithm, applications of GA in Machine Learning : Machine Learning Approach to Knowledge Acquisition. | 9 | 20 |
| 5 | **Unit 5: Matlab/Python Lib:** Introduction to Matlab/Python, Arrays and array operations, Functions and Files, Study of neural network toolbox and fuzzy logic toolbox, Simple implementation of Artificial Neural Network and Fuzzy Logic | 10 | 20 |
| 6 | **Unit 6 :** Recent Trends in various classifiers, neural networks and genetic algorithm | 3 | 06 |

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Master of Engineering**
**Subject Code: 3720217**

**Reference Books:**


1. Jyh Shing Roger Jang, Chuen Tsai Sun, Eiji Mizutani, Neuro-Fuzzy and Soft Computing, Prentice Hall of India, 2003.
2. George J. Klir and Bo Yuan, Fuzzy Sets and Fuzzy Logic: Theory and Applications, Prentice Hall, 1995.
3. MATLAB Toolkit Manual
4. Timothy J.Ross, Fuzzy Logic with Engineering Applications, McGraw-Hill
5. Goldberg, D. E, Genetic algorithm in search, optimization and machine learning, Addison-Wesley, Reading Mass
6. S.N.Sivanandam, S.N.Deepa , Principles of Soft Computing, 2e, Wiley India Pvt. Ltd.
7. S. RAJASEKARAN, G. A. VIJAYALAKSHMI PAI, NEURAL NETWORKS, FUZZY LOGIC AND GENETIC ALGORITHM: SYNTHESIS AND APPLICATIONS, PHI Learning Pvt. Ltd


**Course Outcome:**

| Sr. No. | CO statement | Marks % weightage |
|---|---|---|
| CO-1 | Identify and describe soft computing techniques and their roles in building intelligent machines. | **20** |
| CO-2 | Apply fuzzy logic and reasoning to handle uncertainty and solve various engineering problems. | **20** |
| CO-3 | Apply genetic algorithms to combinatorial optimization problems. | **20** |
| CO-4 | Evaluate and compare solutions by various soft computing approaches for a given problem. | **20** |
| CO-4 | Use various tools to solve soft computing problems. | **20** |

**Suggested List of Experiments:** If MATLAB is not available, the practical may be carried out in SCILAB or C/C++/Java

1. Introduction to MATLAB & its environment.
2. Introduction to MATLAB: Fuzzy Logic Toolbox, Fuzzy Logic Simulink Demos
3. Introduction to MATLAB: Neural Network (NN) Toolbox, NN Simulink Demos
4. MATLAB simulation: Artificial Neural Network (ANN) implementation
5. MATLAB simulation: NN Tool Artificial Neural Network (ANN) implementation
6. MATLAB simulation: Various structure of NN algorithms implementation
7. MATLAB simulation: Training Algorithms of ANN.
8. MATLAB simulation: Coding and minimizing a fitness function using GA.

**List of Open Source Software/learning website:**

1. http://www.iitk.ac.in/kangal/codes.shtml
2. http://lancet.mit.edu/ga/dist/galibdoc.pdf
3. https://books.google.co.in/books?hl=en&lr=&id=W5SAhUqBVYoC&oi=fnd&pg=PR11&dq=SOft+computing+course+&ots=et_2Nvjy_4&sig=jDXLrGIeD3zc4QUxvcEvC5FrFY#v=onepage&q=SOft%20computing%20course&f=false

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Master of Engineering**
**Subject Code: 3720217**

**Major Equipments / Software:**

Students may implement open ended problems on some Microprocessors / DSP boards. Computers with MATLAB / Scilab/ C/C++/Java software may serve the purpose.

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Master of Engineering**

**Subject Code: 3720218**

**SUBJECT NAME:** Secure Software Design and Enterprise Computing

**Semester II**

**Type of course: Elective**

**Prerequisite: Computer Programming and Software Engineering**

**Rationale:**. Students should be made aware with secure software design issues and understand how to fix software flaws and bugs in various software. Students will come to understand various issues like weak random number generation, information leakage, poor usability, and weak or no encryption on data traffic. They will be able to learn and apply techniques for successfully implementing and supporting network services on an Enterprise scale and heterogeneous systems environment. Study of methodologies and tools to design and develop secure software containing minimum vulnerabilities and flaws.

**Teaching and Examination Scheme:**

| Teaching Scheme | | | Credits | Examination Marks | | | | Total Marks |
|---|---|---|---|---|---|---|---|---|
| L | T | P | C | Theory Marks | | Practical Marks | | |
| | | | | ESE(E) | PA (M) | PA (V) | PA (I) | |
| 3 | 0 | 2 | 4 | 70 | 30 | 30 | 20 | 150 |

**Content:**

| Sr. No. | Content | Total Hrs | % Weightage |
|---|---|---|---|
| 1 | **Unit 1:**<br>**Secure Software Design**<br>Identify software vulnerabilities and perform software security analysis, Master security programming practices, Master fundamental software security design concepts, Perform security testing and quality assurance. | **9** | **20** |
| 2 | **Unit 2:**<br>**Enterprise Application Development**<br>Describe the nature and scope of enterprise software applications, Design distributed N-tier software application, Research technologies available for the presentation, business and data tiers of an enterprise software application, Design and build a database using an enterprise database system, Develop components at the different tiers in an enterprise system, Design and develop a multi-tier solution to a problem using technologies used in enterprise system, Present software solution. | **12** | **25** |
| 3 | **Unit 3: Enterprise Systems Administration**<br>Design, implement and maintain a directory-based server infrastructure in a heterogeneous systems environment, Monitor server resource utilization for system reliability and availability, Install and administer network services (DNS/DHCP/Terminal Services/Clustering/Web/Email). | **10** | **20** |
| 4 | **Unit 4: Enterprise Network**<br>Obtain the ability to manage and troubleshoot a network running multiple services, Understand the requirements of an enterprise network and how to | **10** | **20** |

| | | | |
|---|---|---|---|
| | go about managing them. | | |
| **5** | **Unit 5: Defending Applications**<br>Handle insecure exceptions and command/SQL injection, Defend web and mobile applications against attackers, software containing minimum vulnerabilities and flaws. | **7** | **15** |

**Reference Books:**

1. Theodor Richardson, Charles N Thies, Secure Software Design, Jones & Bartlett
2. Kenneth R. van Wyk, Mark G. Graff, Dan S. Peters, Diana L. Burley, Enterprise Software Security, Addison Wesley.

**Course Outcome:**

**Course Outcomes:**

| Sr. No. | CO statement | Marks % weightage |
|---|---|---|
| CO-1 | Understand Software process vulnerabilities for an organization and design and develop multi-tier solution of a problem. | **45** |
| CO-2 | Able to administer Enterprise System. | **20** |
| CO-3 | Troubleshoot Enterprise Network. | **20** |
| CO-4 | Develop secure software which can defend against attackers | **15** |

**Practicals & Assignments :**

1. Study of various open source security tools for Application testing, Code Review, Penetration Testing, Vulnerability Assessment, Vulnerability Scanner etc.
2. Design and develop multi tier application for an enterprise.
3. Installation of Directory based Server and monitoring resource utilization.
4. Practicals based on network services such as DNS/DHCP/Terminal Services/Clustering/Web/Email
5. Study of SQL Injection Problem.
6. Developing application that can defend SQL injection problem.

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Master of Engineering**
**Subject Code: 3720219**
**Semester – II**
**Subject Name: Computer Vision**

**Type of course: Elective**

**Prerequisite:** Linear Algebra, Vector Calculus, Data Structures and Programming.

**Rationale:**

- Be familiar with both the theoretical and practical aspects of computing with images.
- Have described the foundation of image formation, measurement, and analysis.
- Understand the geometric relationships between 2D images and the 3D world.
- Grasp the principles of state-of-the-art deep neural networks.

**Teaching and Examination Scheme:**

| Teaching Scheme | | | Credits | Examination Marks | | | | Total Marks |
|---|---|---|---|---|---|---|---|---|
| | | | | Theory Marks | | Practical Marks | | |
| L | T | P | C | ESE (E) | PA (M) | ESE (V) | PA (I) | |
| 3 | 0 | 2 | 4 | 70 | 30 | 30 | 20 | 150 |

**Content:**

| Sr. No. | Content | Total Hrs | % Weightage |
|---|---|---|---|
| **1** | Overview, computer imaging systems, lenses, Image formation and sensing, Image analysis, pre-processing and Binary image analysis. | **8** | 16% |
| **2** | Edge detection, Edge detection performance, Hough transform, corner detection | **9** | 19% |
| **3** | Segmentation, Morphological filtering, Fourier transform | **9** | 19% |
| **4** | Feature extraction, shape, histogram, color, spectral, texture, using CV IP tools, Feature analysis, feature vectors, distance /similarity measures, data preprocessing | **9** | 19% |
| **5** | Pattern Analysis: Clustering: K-Means, K-Medoids, Mixture of Gaussians Classification: Discriminant Function, Supervised, Un-supervised, Semi-supervised Classifiers: Bayes, KNN, ANN models; Dimensionality Reduction: PCA, LDA, ICA, and Non-parametric methods. | **9** | 19% |
| **6** | Recent trends in Activity Recognition, computational photography, Biometrics. | **4** | 8% |
| | **Total** | **48** | **100%** |

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Master of Engineering**
**Subject Code: 3720219**

**Reference Books:**


1. Computer Vision: Algorithms and Applications by Richard Szeliski.
2. Computer Vision – A modern approach, by D.Forsyth and J.Ponce, Prentice Hall Robot Vision, by B. K. P. Horn, McGraw-Hill.
3. Deep Learning, by Goodfellow, Bengio, and Courville.
4. Dictionary of Computer Vision and Image Processing, by Fisher et al.
5. Three-Dimensional Computer Vision, by Olivier Faugeras, The MIT Press.


**Course Outcomes:**

At the end of the module the student will be able to:

| Sr. No. | CO statement | Marks % weightage |
|---|---|---|
| CO-1 | Identify basic concepts, terminology, theories, models and methods in the field of computer vision. | 20% |
| CO-2 | Describe basic methods of computer vision related to multi-scale representation, edge detection and detection of other primitives, stereo, motion and object recognition. | 35% |
| CO-3 | Developed the practical skills necessary to build computer vision applications. | 30% |
| CO-4 | To have gained exposure to object and scene recognition and categorization from images. | 15% |

**List of Experiments:**

1. To perform variants of linear filter on an image.
2. To perform median filter on an image.
3. To perform all morphological filter operations on the image.
4. To perform edge detection on an image using Sobel, Prewitt, Roberts and Canny Method.
5. To perform Hough transform on an image.
6. Generate histogram and perform histogram equalization of an image.
7. To perform clustering using any technique on the image.
8. To perform various pattern classification and analysis techniques on images.
9. To perform various pattern clustering and analysis techniques on images.

**List of Open Source Software/learning website:** Open CV, Python

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Master of Engineering**
**Subject Code: 3720220**
**Semester – II**
**Subject Name: HPC Architecture and ECO system**

**Type of course: Elective**

**Prerequisite:** Computer Architecture and Organization, Data Structure

**Rationale:** Algorithmic processing performed in High Performance Computing environments impacts the lives of billions of people, and planning for exascale computing presents significant power challenges to the industry. The objective of the course is to impart in depth knowledge of parallel architectures and different shared/distributed memory architectures in support to exascale computing. MPI and OpenMP are discussed along with their applications.

**Teaching and Examination Scheme:**

| Teaching Scheme | | | Credits | Examination Marks | | | | Total Marks |
|---|---|---|---|---|---|---|---|---|
| | | | | Theory Marks | | Practical Marks | | |
| L | T | P | C | ESE (E) | PA (M) | ESE (V) | PA (I) | |
| 3 | 0 | 2 | 4 | 70 | 30 | 30 | 20 | 150 |

**Content:**

| Sr. No. | Content | Total Hrs |
|---|---|---|
| **1** | Overview of parallel system organization | 4 |
| **2** | Examples of Scientific Computing; Parallel Languages, Embarrassingly parallel problems; problem decomposition, graph partitioning and load balancing | 12 |
| **3** | Introduction to message passing and MPI programming | 7 |
| **4** | Introduction to shared memory and OpenMP programming | 7 |
| **5** | Recent trends in OpenMP and MPI programming, application areas of scientific computing. | 9 |
| **6** | Performance analysis tools; HPCToolkit, OpenSpeedShop,<br>Debugging and Profiling tools; GDB, PAPI, mpiP, ompP, Valgrind, MPI Program Profiler | 9 |
| | **Total** | **48** |

**Reference Books:**


1. 1. Parallel Programming for Multicore and Cluster Systems by Thomas Rauber and Gudula Runger.
2. Scientific Parallel Computing by Scott, Clark, and Bagheri.

3. Using OpenMP: Portable Shared Memory Parallel Programming by Chapman, Jost, and van der Pas.

**Course Outcomes:**

| Sr. No. | CO statement | Marks % weightage |
|---|---|---|
| CO-1 | Develop parallel algorithms for high performance systems. | 25% |
| CO-2 | Perform problem decomposition and load balancing using MPI and OpenMP. | 25% |
| CO-3 | Identify hot spots in the codes using performance analysis tools. | 25% |
| CO-4 | Evaluate the performance of the code using the tools | 25% |

Distribution of marks weightage for cognitive level

| Bloom's Taxonomy for Cognitive Domain | Marks % weightage |
|---|---|
| Recall | 5 |
| Comprehension | 5 |
| Application | 20 |
| Analysis | 25 |
| Evaluate | 25 |
| Create | 20 |

**Practical List:**

Use Valgrind, Vtune Amplifier, Nvidia Visual Profiler and Nvidia Nsight to identify hotspots and other parameters for detailed analysis of following the practicals.

1) Calculate standard deviation using Pthread, OpenMP and MPI.
2) Write parallel code for Matrix Matrix Multiplication using MPI cluster of 4 nodes, OpenMP, PVM cluster of 4 nodes, OpenACC and CUDA and compare and plot the performance in terms of execution time for Matrix size of 1000 x 1000, 5000 x 5000, 10,000 x 10,000, 20,000 x 20,000.
3) Write the programs in MPD or in C with the Pthreads library for the following:
   a) Sequential Jacobi iteration program
   b) Parallel Jacobi iteration program
   c) Sequential multigrid program
   d) Parallel multigrid program
4) Perform Monte Carlo simulation using NVIDIA's CURAND library for random number generation.

   Write your own small program to compute the average value of

   $az^2 + bz + c$

where z is a standard Normal random variable (i.e. zero mean and unit variance, which is what the random number generator produces) and a, b, c are constants which you should store in constant memory. It is suggested to use each thread to average over 100 values, and then write this to a device array which gets copied back to the host for the averaging over the contributions from each of the threads.

(Note: the average value should be close to a + c.)

5) Implement 3D Laplace Finite Solver using CUDA and OpenACC.

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Master of Engineering**
**Subject Code: 3720221**
**Semester – II**
**Subject Name: GPU Computing**

**Type of course:** Elective

**Prerequisite:**

**Rationale:** Almost, all modern processors are multi-core, and in order to make effective use of them, you need to run multiple threads. When accelerators, such as GPUs, are available, then the number of threads needs to be even larger. This course teaches you how to organize the computations of the threads so that they work together and performs the required computations efficiently, making good use of the available hardware resources. We will focus on using GPUs for general purpose computing, rather than for graphics.

**Teaching and Examination Scheme:**

| Teaching Scheme | | | Credits | Examination Marks | | | | Total Marks |
|---|---|---|---|---|---|---|---|---|
| | | | | Theory Marks | | Practical Marks | | |
| L | T | P | C | ESE (E) | PA (M) | ESE (V) | PA (I) | |
| 3 | 0 | 2 | 4 | 70 | 30 | 30 | 20 | 150 |

**Content:**

| Sr. No. | Content | Total Hrs | % Weightage |
|---|---|---|---|
| 1 | **Introduction:** History, GPU Architecture, Clock speeds, CPU / GPU comparisons, Heterogeneity, Accelerators, Parallel Programming, CUDA OpenCL / OpenACC, Kernels Launch parameters, Thread hierarchy, Warps/Wavefronts, Threadblocks/Workgroups, Streaming multiprocessors, 1D/2D/3D thread mapping, Device properties, Simple Programs | 13 | 27 |
| 2 | **Memory**: Memory hierarchy, DRAM / global, local / shared, private / local, textures, Constant Memory, Pointers, Parameter Passing, Arrays and dynamic Memory, Multi-dimensional Arrays, Memory Allocation, Memory copying across devices, Programs with matrices, Performance evaluation with different memories | 7 | 15 |
| 3 | **Synchronization**: Memory Consistency, Barriers (local versus global), Atomics, Memory fence. Prefix sum, Reduction. Programs for concurrent Data Structures such as Worklists, Linked-lists. Synchronization across CPU and GPU<br>**Functions**: Device functions, Host functions, Kernels functions, Using libraries (such as Thrust), and developing libraries. | 10 | 21 |
| 4 | **Support**: Debugging GPU Programs. Profiling, Profile tools, Performance aspects<br>**Streams**: Asynchronous processing, tasks, Task-dependence, Overlapped data transfers, Default Stream, Synchronization with streams. Events, Event-based-Synchronization - Overlapping data transfer and kernel execution, pitfalls. | 8 | 17 |

| | | | |
|---|---|---|---|
| 5 | **Case Studies**: Image Processing, Graph algorithms, Simulations, Deep Learning | 5 | 10 |
| 6 | **Advanced topics**: Dynamic parallelism, Unified Virtual Memory, Multi-GPU processing, Peer access, Heterogeneous processing | 5 | 10 |
| | **Total** | 48 | **100%** |

**Reference Books:**

1. David Kirk and Wen-mei Hwu, Programming Massively Parallel Processors: A Hands-On Approach, 2nd Edition, Publisher: Morgan Kaufman, 2012, ISBN: 9780124159921.
2. Shane Cook, CUDA Programming: A Developer's Guide to Parallel Computing with GPUs, Morgan Kaufman; 2012 (ISBN: 978-0124159334)

**Course Outcomes:**

After learning the course the students should be able to:

| Sr. No. | CO statement | Marks % weightage |
|---|---|---|
| CO-1 | Define terminology commonly used in parallel computing, such as efficiency and speedup. | 10% |
| CO-2 | Describe common GPU architectures and programming models. | 20% |
| CO-3 | Implement efficient algorithms for common application kernels, such as matrix multiplication. | 20% |
| CO-4 | Given a problem, develop an efficient parallel algorithm to solve it. | 25% |
| CO-5 | Given a problem, implement an efficient and correct code to solve it, analyze its performance, and give convincing written and oral presentations explaining the achievements. | 25% |

**Distribution of marks weightage for cognitive level**

| Bloom's Taxonomy for Cognitive Domain | Marks % weightage |
|---|---|
| Recall | 5 |
| Comprehension | 10 |
| Application | 20 |
| Analysis | 25 |
| Evaluate | 25 |
| Create | 15 |

**Practical List:**

1. Write CUDA code to compute the squares of the first N integers.
2. Write CUDA code to determine the following:
    i. data transfer bandwidth from host to device

ii. data transfer bandwidth from device to host
iii. data transfer bandwidth from host to device using pinned memory
iv. data transfer bandwidth from device to host using pinned memory
v. kernel creation overhead

3. Implement matrix multiplication on the CPU and GPU(without using shared memory), and compare their relative performances in terms of GFlop/s and report your performance results.
4. Write code for matrix multiplication using shared memory and compare its performance with CPU code. Determine the best tile-size for matrix multiplication using shared memory for your GPU machine.
5. Implement the 1-D convolution kernel and compare the performance with and without shared memory.
6. Perform 2-D convolution on an n x n matrix with an m x m mask and determine the number of halo cells required for it.
7. Implement 2-D convolution with data in shared memory. Also, analyze the reduction in bandwidth from use of shared memory in 2-D convolution.
8. Write code for histogramming without atomic operations and experiment with it to find out the fraction of times that it gives incorrect results and report it.
9. Rewrite program 8 so that it uses atomic operations, and compare its performance with it and report it.
10. Write code to perform reduction using atomic operations on global memory. (ii) Rewrite this code to use shared memory in order to reduce the overhead of atomic operations. (iii) Compare the performances of the above two with the reduction algorithm we had discussed earlier.
11. Write code with different threads performing atomic operations on different, but adjacent, locations in memory. Compare the performance with different data sizes.

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Master of Engineering**
**Subject Code: 3720222**
**Semester – II**
**Subject Name: Parallel Algorithms**

**Type of course: Elective**

**Prerequisite:** Data Structures, Design and Analysis of Algorithms

**Rationale:** Parallel computing has become mainstream and very affordable today. This is mainly because hardware costs have come down rapidly. Processing voluminous datasets is highly computation intensive. Parallel computing has been fruitfully employed in numerous application domains to process large datasets and handle other time-consuming operations of interest. As a result, unprecedented advances have been made in such areas as biology, scientific computing, modeling and simulations, and so forth. Hence the objective of this course is to introduce parallel algorithms and compare it with its sequential equivalent.

**Teaching and Examination Scheme:**

| Teaching Scheme | | | Credits | Examination Marks | | | | Total Marks |
|---|---|---|---|---|---|---|---|---|
| | | | | Theory Marks | | Practical Marks | | |
| L | T | P | C | ESE (E) | PA (M) | ESE (V) | PA (I) | |
| 3 | 0 | 2 | 4 | 70 | 30 | 30 | 20 | 150 |

**Content:**

| Sr. No. | Content | Total Hrs | % Weightage |
|---|---|---|---|
| 1 | Computers Models of computation, Need for parallelism, Complexity measure for parallel algorithms, parallel computational models such as PRAM, LMCC, Hypercube, Cube Connected Cycle, Butterfly, Perfect Shuffle Computers, Tree model, Pyramid model, Fully Connected model, PRAM, CREW, EREW models, simulation of one model from another one, Expressing parallel algorithms | 5 | 10 |
| 2 | Parallel combinatorial algorithms: permutations with and without repetitions combinations, derangements | 4 | 8 |
| 3 | Maximum/Minimum, Median, Kth Largest/Smallest element, Matrix Vector Multiplication,Matrix-Matrix Multiplication, Parallel discrete event simulation Image dithering, Dense LU factorization | 6 | 13 |
| 4 | Parallel sorting algorithms: Hyper quick sort, Merge sort, Bitonic merge Sort, odd even transposition, Enumeration sort (sorting on the CRCW model, CREW model and EREW model) | 11 | 23 |
| 5 | Parallel searching algorithms: Searching on a sorted sequence (EREW,CREW,CRCW), Searching on a random sequence (EREW, CREW, CRCW, Tree and Mesh) Sequential selection algorithm, Parallel selection algorithm (EREW parallel solution) | 11 | 23 |

| | | | |
|---|---|---|---|
| 4 | Parallel graph algorithms: parallel graph search &, tree traversal algorithms, Graph coloring, Minimal spanning tree, Shortest path algorithm | 11 | 23 |
| | **Total** | 48 | **100%** |

**Reference Books:**

1. Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar, Introductionto Parallel Computing, Second Edition, Addison Wesley, 2003. ISBN: 0-201-64865.
2. S. Akl. Design and Analysis of Parallel Algorithms, Prentice Hall Inc, 1992.
3. Michael Quinn, Parallel Computing Theory and Practice, McGraw Hill,Second Edition, 1994.
4. F.T.Leighton, Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes , MK Publishers, San Mateo California, 1992..
5. Wilkinson, M.Allen, Parallel Programming Techniques and Applications using networked workstations and parallel computers, Prentice Hall, 1999
6. Joseph Jaja. An Introduction to Parallel Algorithms, Addison Wesley, 1992.
7. H. Sparkias and A. Gibbon. Lecture notes on Parallel Computation, Cambridge University Press, 1993.
8. K. Hwang and F. A. Briggs.Computer Architecture and Parallel Processing, McGraw Hill Inc., 1985.

**Course Outcomes:**

After learning the course the students should be able to:

| Sr. No. | CO statement | Marks % weightage |
|---|---|---|
| CO-1 | Gain basic understanding of fundamental concepts in parallel computing. | 10% |
| CO-2 | Be able to identify and leverage common parallel computing patterns. | 20% |
| CO-3 | know about parallel computing model like PRAM, LMCC etc. | 20% |
| CO-4 | analyze the computational complexity of parallel algorithms | 20% |
| CO-5 | Be able to properly assess efficiency and scalability of a parallel algorithm/application. | 30% |

Distribution of marks weightage for cognitive level

| Bloom's Taxonomy for Cognitive Domain | Marks % weightage |
|---|---|
| Recall | 5 |
| Comprehension | 10 |
| Application | 15 |
| Analysis | 25 |
| Evaluate | 25 |
| Create | 20 |

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Master of Engineering**
**Subject Code: 3720222**

**Practical List:**

1) Compare the speedup of the parallel implementation of Quick sort using MPI (On a cluster of 5 Nodes) and OpenMP (Shared Memory Implementation on multicore machine).
2) Compare the speedup of the parallel implementation of Merge sort using MPI (On a cluster of 5 Nodes) and OpenMP (Shared Memory Implementation on multicore machine).
3) Compare the speedup of the parallel implementation of Bitonic sort using MPI (On a cluster of 5 Nodes) and OpenMP (Shared Memory Implementation on multicore machine).
4) Compare the speedup of the parallel implementation of Odd-Even transposition sort using MPI (On a cluster of 5 Nodes) and OpenMP (Shared Memory Implementation on multicore machine).
5) Implement parallel list ranking algorithm on MPI cluster of 5 Nodes and find it speedup over the sequential implementation.
6) Give parallel implementation (OpenMP/MPI) of Awerbuch-Shiloach algorithm for finding the connected components of a graph.

# GUJARAT TECHNOLOGICAL UNIVERSITY

**Master of Engineering**
**Subject Code: 3720223**
**Semester – II**
**Subject Name: Advance Machine Learning**

**Type of course: Regular**

**Prerequisite:** Machine Learning, Probability Theory

**Rationale:** This course will introduce key concepts in pattern recognition and machine learning; including specific algorithms for classification, regression, clustering and probabilistic modeling. In summary, this course will provide a broad view of the general issues arising in the application of algorithms to analyzing data, common terms used, and common errors made if applied incorrectly.

**Teaching and Examination Scheme:**

| Teaching Scheme | | | Credits | Examination Marks | | | | Total Marks |
|---|---|---|---|---|---|---|---|---|
| L | T | P | C | Theory Marks | | Practical Marks | | |
| | | | | ESE (E) | PA (M) | ESE (V) | PA (I) | |
| 3 | 0 | 2 | 4 | 70 | 30 | 30 | 20 | 150 |

**Content:**

| Sr. No. | Content | Total Hrs |
|---|---|---|
| 1 | **Key concepts**, Supervised/Unsupervised Learning, Loss functions and generalization, Probability Theory, Parametric vs Non-parametric methods, Elements of Computational Learning Theory<br>Ensemble Learning, Bagging, Boosting, Random Forest | 8 |
| 2 | **Kernel Methods** for non-linear data, Support Vector Machines, Kernel Ridge Regression, Structure Kernels, Kernel PCA, Latent Semantic Analysis | 8 |
| 3 | **Bayesian methods** for using prior knowledge and data, Bayesian inference, Bayesian Belief Networks and Graphical models, Probabilistic Latent Semantic Analysis, The Expectation-Maximisation (EM) algorithm, Gaussian Processes | 8 |
| 4 | **Dimensionality Reduction** - CCA, LDA, ICA, NMF - Canonical Variates - Feature Selection vs Feature Extraction | 10 |
| 5 | **Filter Methods** - Sub-space approaches - Embedded methods<br>Low-Rank approaches - Recommender Systems<br>.Application areas - Security - Business - Scientific | 9 |

| | | |
|---|---|---|
| **6** | Recent trends in supervised and unsupervised learning algorithm, dimensional reducibility, feature selection and extraction | **5** |

**Reference Books:**

1. Pattern Recognition and Machine Learning, Christopher M. Bishop
2. John Shawe-Taylor and NelloCristianini, Kernel Methods for Pattern Analysis.
3. The Elements of Statistical Learning, Springer 2009
4. Machine Learning Algorithms, 2nd Edition, Giuseppe Bonaccorso, Packt Publication
5. TensorFlow Machine Learning, Nick McClure, Packt Publication

**Course Outcomes:**

| Sr. No. | CO statement | Marks % weightage |
|---|---|---|
| CO-1 | Key concepts, tools and approaches for pattern recognition on complex data sets | **20%** |
| CO-2 | Kernel methods for handling high dimensional and non-linear patterns | **25%** |
| CO-3 | State-of-the-art algorithms such as Support Vector Machines and Bayesian networks. | **25%** |
| CO-4 | Solve real-world machine learning tasks: from data to inference | **15%** |
| CO-5 | Theoretical concepts and the motivations behind different learning frameworks | **15%** |

**List of Experiments:**

- Minimum 10 experiments based on the above contents.
- Mini Project in a group of max. 3 students
- Writing a research paper on selected topic from content with latest research issues in that topic

**Major Equipments:**

- Latest PCs with related software

**List of Open Source Software/learning website:**

- https://www.analyticsvidhya.com/blog/2016/01/complete-tutorial-learn-data-science-python-scratch-2/
- https://www.springboard.com/resources/learning-paths/machine-learning-python
- https://www.rstudio.com/online-learning/