MULTICORE ARCHITECTURE AND PROGRAMMING	
Course Code: CI71	**Credits: 3:0:1**
Prerequisite: Nil	**Contact Hours: 42L+14P**
Course Coordinator: Dr. Mohana Kumar S	

Course Contents

Unit I

Introduction to Multi-Core Architecture: Motivation for Concurrency in Software, Parallel Computing Platforms, Parallel Computing in Microprocessors, Differentiating Multi-Core Architectures from Hyper- Threading Technology, Multi-Threading on Single-Core Versus Multi-Core Platforms Understanding Performance, Amdahl's Law, Growing Returns: Gustafson's Law. **System Overview of Threading:** Defining Threads, System View of Threads, Threading Above the Operating System, Threads Inside The OS, Threads Inside the Hardware, What Happens When A Thread Is Created, Application Programming Models and Threading, Virtual Environment: VMs And Platforms, Runtime Virtualization, System Virtualization.

Unit II

Fundamental Concepts of Parallel Programming: Designing for Threads, Task Decomposition, Data Decomposition, Data Flow Decomposition, Implications of Different Decompositions, Challenges You'll Face, Parallel Programming Patterns, A Motivating Problem: Error Diffusion, Analysis Of The Error Diffusion Algorithm, Alternate Approach: Parallel Error Diffusion, Other Alternatives.

Unit III

Threading and Parallel Programming Constructs: Synchronization, Critical Sections, Deadlock, Synchronization Primitives, Semaphores, Locks, Condition Variables, Messages, Flow Control-Based Concepts, Fence, Barrier, Implementation-Dependent Threading Features. Threading API's: Threading API's For Microsoft Windows, Win32/MFC Thread API's, Threading API's For Microsoft. NET Framework, Creating Threads, Managing Threads, Thread Pools, Thread Synchronization, POSIX Threads, Creating Threads, Managing Threads, Thread Synchronization, Signalling, Compilation and Linking.

Unit IV

Open MP: A Portable Solution For Threading: Challenges In Threading A Loop, Loop-Carried Dependence, Data-Race Conditions, Managing Shared And Private Data, Loop Scheduling And Portioning, Effective Use Of Reductions, Minimizing Threading Overhead, Work-Sharing Sections, Performance-Oriented Programming, Using Barrier And No Wait, Interleaving Single-Thread And Multi-Thread Execution, Data Copy-In And Copy-Out, Protecting Updates Of Shared Variables, Intel Task Queuing Extension to Open MP, OpenMP Library Functions, OpenMP Environment Variables, Compilation, Debugging, Performance.

Unit V

Solutions To Common Parallel Programming Problems: Too Many Threads, Data Races, Deadlocks, And Live Locks, Deadlock, Heavily Contended Locks, Priority Inversion, Solutions For Heavily Contended Locks, Non-Blocking Algorithms, ABA Problem, Cache Line Ping-Ponging, Memory Reclamation Problem, Recommendations, Thread-Safe Functions And Libraries, Memory Issues, Bandwidth, Working In The Cache, Memory Contention, Cache-Related Issues, False Sharing, Memory Consistency, Current IA-32 Architecture, Itanium Architecture, High-Level Languages, Avoiding Pipeline Stalls On IA-32,Data Organization For High Performance.

Text Books:

1. Shameem Akhter and Jason Roberts, Multicore Programming, Increased Performance through Software Multi-threading, Intel Press, 2006
2. Hennessey and Patterson, Computer Architecture A Quantitative Approach, 4th Edition, Elsevier, 2012.

Reference Book:

1. Kai Hwang and Naresh Jotwani, Advanced Computer Architecture - Parallelism, Scalability, Programmability, 2nd Edition, Tata McGraw Hill, 2011.

Course Outcomes (COs):

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

1. Understand performance-related parameters in the field of Computer Architecture.
2. Identify the limitations of ILP and the need for multi-core architectures.
3. Solve the issues related to multiprocessing and suggest solutions.
4. Point out the salient features of different multi-core architectures and how they exploit parallelism.
5. Understand the concept of multi-threading and OpenMP.

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	**Marks**	**Course outcomes addressed**
Internal test-I	30	CO1, CO2, CO3
Internal test-II	30	CO3, CO4, CO5
The average of two internal tests shall be taken for 30 marks.		
Other components		
Assignment 1	10	CO1, CO2, CO3
Assignment 2	10	CO1, CO2, CO3, CO4, CO5
Semester End Examination (SEE)	100	CO1, CO2, CO3, CO4, CO5

FOUNDATIONS OF COMPUTER VISION	
Course Code: CI72	**Credits: 2:1:0**
Prerequisite: Nil	**Contact Hours: 28L+14T**
Course Coordinator: Dr. Nithya N	

Course Contents

Unit I

Introduction: Computer vision, Imaging modalities, Fundamental steps in image processing, Applications of computer vision. Digital Image Fundamentals: Image formation model, Sampling and quantization, Relationships between pixels. Mathematical tools used in image processing.

Unit II

Spatial Filtering: Intensity transformation functions, Histogram processing (Histogram equalization, Histogram matching), Fundamentals of spatial filtering (Mechanisms of spatial filtering, correlation and convolution), Smoothing spatial filters, Sharpening spatial filters.

Unit III

Image Segmentation: Fundamentals, Detection of isolated points, line and basic edge, Thresholding, Region-based segmentation. Representation and Description: Representation (border following, chain codes, minimum-perimeter polygons) Boundary descriptors (simple descriptors, shape numbers), Region descriptors (simple descriptors, topological descriptors, texture).

Unit IV

Object Recognition: What Should Object Recognition Do? Feature, Geometric, and semantic questions, Patterns, and pattern classes, Recognition based on decision-theoretic methods, Matching, Optimum statistical classifier, Neural networks

Unit V

Morphological Processing: Erosion and Dilation, Opening and Closing, Hit-or-miss transform, Morphological algorithms (Boundary extraction, Hole filling, Extraction of connected components). Compression Techniques: Fundamentals, Compression methods (Huffman, Arithmetic, Run-length coding)

Text Book:

1. Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing, 3rd edition., Pearson, 2010.
2. D.A. Forsyth and J. Ponce, Computer Vision: A modern approach, Pearson Education, 2015.

References:

1. Anil K. Jain, Fundamentals of Digital Image Processing, Pearson Education, 2001.
2. B. Chanda and D. Dutta Majumdar, Digital Image Processing and Analysis, PHI, 2003.

CONTAINERIZATION LABORATORY	
Course Code: CIL74	**Credits: 0:0:1**
Prerequisite: Nil	**Contact Hours: 14P**
Course Coordinator: Dr. Siddesh G M	

Course Contents

1. Overview of virtualization, AWS Platform, Load Balancing
2. Amazon Virtual Private Cloud.
3. Introduction to GIT Workflow.
4. Working with various commands in GIT.
5. Recording Changes to the Repository.
6. Creating a Service in Kubernetes, Installing Kubernetes Dashboard.
7. Deploying an App using Dashboard. Using Rolling Updates in Kubernetes.
8. Containers and Container Orchestration.
9. Working with Docker Containers. Docker Command Line Interphase.
10. Deploying an application to an EC2 fleet using AWS.
11. Automating code deployments using AWS Code.
12. Pipeline and Amazon Elastic Container Service (Amazon ECS).

References:

1. Chris Richardson: Microservices Patterns with Examples in Java, Manning Publications Co., First Edition, 2019.
2. Moises Macero: Learn Microservices with Spring Boot: A Practical Approach to RESTful Services using RabbitMQ, Eureka, Ribbon, Zuul and Cucumber, A Press, First Edition, 2017.
3. Sourabh Sharma: Mastering Microservices with Java 9, Packt Publishing Ltd, Second Edition, 2017.

Course Outcomes (COs):

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

1. Illustrate the importance of Microservices as an Architecture Implementation.
2. Develop a Containers and Container Orchestration.
3. Deploy application on docker and Access the Kubernetes.

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tools	**Marks**	**Course Outcomes (COs)addressed**
Lab Test-I	10	CO1, CO2
Lab Test-II	10	CO3
Weekly Evaluation+Lab Record	30	CO1, CO2, CO3
The Final CIE out of 50 Marks = Marks of Lab Record + Marks scored in Lab Test-I + Marks scored in Lab Test-II		
Semester End Examination (SEE)	50	CO1, CO2, CO3

SKILL ENHANCEMENT LAB - GENERATIVE AI	
Course Code: CIL75	**Credits: 0:1:2**
Prerequisite: Nil	**Contact Hours: 14T+28P**
Course Coordinator:	

Course Contents

Skill Development aims to create convergence across engineering streams in terms of skill training initiatives that would be best designed and implemented. Skill Development is the process of identifying the skills gap and providing skilling training & employment benefits to students. This includes the incorporation of skilling in addition to the academic curriculum, providing opportunities for quality long & short-term skill training, by facilitating gainful employment and ensuring career progression that meets the aspirations of our students.

Introduction to Generative AI Tools

- Setting up the environment: Jupyter notebooks, TensorFlow/PyTorch installation
- Introduction to Colab for cloud-based computing
- Basic walkthrough of TensorFlow and PyTorch for generative modeling
- **Exercise:** Implement a simple feed-forward neural network

Generative Adversarial Networks (GANs)

- Building a basic GAN: Discriminator and Generator architecture
- Training GANs and handling common issues: mode collapse, instability
- Experimenting with DCGAN for image generation
- **Exercise:** Create a GAN to generate synthetic images from a dataset (e.g., MNIST or CIFAR-10)

Variational Autoencoders (VAEs)

- Implementing a VAE: Encoder, Decoder, and Latent Space
- Training VAEs and visualizing latent space representations
- Applications of VAEs: data augmentation, anomaly detection
- **Exercise:** Build a VAE for image reconstruction and analyse latent space

Transformers and Text Generation

- Implementing a Transformer-based model (e.g., GPT)
- Fine-tuning a pre-trained model for text generation
- Generating creative content such as poetry, code, or dialogue
- **Exercise:** Fine-tune a GPT model on a custom text dataset for a specific task

Diffusion Models and Image Generation

- Understanding the principles of Diffusion Models
- Implementing a simple Diffusion Model for image generation
- Experimenting with Denoising Diffusion Probabilistic Models (DDPMs)
- **Exercise:** Use a Diffusion Model to generate high-quality images

Real-World Application Project

- Project planning and ideation: Selecting a real-world problem
- Dataset collection and pre-processing
- Model selection and initial implementation

- **Exercise:** Start a project applying generative AI to a chosen domain (e.g., art, healthcare, text generation)

Project Development and Iteration

- Iterative development: refining models, improving performance
- Model validation and performance evaluation
- Addressing ethical considerations and potential biases
- **Exercise:** Continue developing the project, incorporating feedback and improvements

Project Presentation and Deployment

- Finalizing the project: model deployment strategies
- Preparing for project presentation: slides, demo
- **Exercise:** Present the project to the class and submit the final report

Tools and Software:

- Python programming language
- TensorFlow or PyTorch
- Google Colab or local Jupyter notebooks
- Libraries such as Hugging Face Transformers, OpenAI's Gym for text generation

Text Books:

1. Joseph Babcock and Raghav Bali, Generative AI with Python and TensorFlow2, Packt, 2021.
2. Aarushi Kansal, Building Generative AI-Powered Apps: A Hands-on Guide for Developers, Apress, 2024.

References:

1. François Chollet, Deep Learning with Python, Manning Publications Co., 2017.
2. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools and Techniques to Build Intelligent Systems, 3rd edition, O'Reilly, 2022.
3. Online tutorials and documentation for TensorFlow and PyTorch.

Course Outcomes (COs):

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

1. Illustrate how GenAI improves document processing.
2. Design the prompt engineering for Language Models.
3. Develop UIs for GenAI-driven applications.

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	**Marks**	**Course outcomes addressed**
Internal test-I	20	CO1, CO2, CO3
Regularity/Continuous Assessment	10	CO1, CO2, CO3
The average of the two internal tests shall be taken for 30 marks.		
Other components		
Project Implementation	20	CO1, CO2, CO3
The Final CIE out of 50 marks=CIE TEST + Continuous Assessment Project Implementation		
Semester End Examination (SEE)		
Course End Examination (SEE)	50	CO1, CO2, CO3

INFORMATION RETRIEVAL	
Course Code: CIE731	**Credits: 3:0:0**
Prerequisite: Nil	**Contact Hours: 42L**
Course Coordinator: Dr. S Aarathi	

Course Contents

Unit I

Introduction: Basic concepts in Information Retrieval (IR), Future developments of IR, IR retrieval process, Document Pre-processing: Tokenization, stop words, Normalization Stemming, Lemmatization-Faster Posting, Positional Posting.

- Pedagogy/Course delivery tools: Chalk and talk
- Links:https://www.youtube.com/watch?v=8himX4LmQJQ&list=PLg38siQiGwWHdLUMo_UIPlYTh5cxTWRSB&index=3

Unit II

Classic IR Models: Boolean Model, Vector space model: tf-idf weighting, Probabilistic Model, Language models for IR: The language model and the query likelihood model, comparison of Classic models, Evaluation **Measures:** Precision, Recall **Text Indexing:** Blocked sort-based indexing, TF-IDF, K-gram Indexing.

- Pedagogy/Course delivery tools: Chalk and talk
- Links:https://www.youtube.com/watch?v=SU5a4wkWdHg&list=PLg38siQiGwWHdLUMo_UIPlYTh5cxTWRSB&index=9&ab_channel=VenkateshVinayakarao

Unit III

Index Compression: Statistical properties of terms in information retrieval, Dictionary compression, Postings file compression, **Text data Compression**: Statistical Methods, Huffman tree algorithm, Parametric and zone indexes, **Text Classification:** The text classification problem, Naive Bayes algorithm, Flat clustering: Clustering in information retrieval, k-means algorithm. Machine learning Algorithms: Machine- learned scoring.

- Pedagogy/Course delivery tools: Chalk and talk
- Links:https://www.youtube.com/watch?v=BPmlYOVLr0A&list=PLg38siQiGwWHdLUMo_UIPlYTh5cxTWRSB&index=4&ab_channel=VenkateshVinayakarao

Unit IV

String Matching algorithms: Knuth Morris Pratt and Rabin Karp, stemming algorithm: Porter **Web search basics:** web characteristics, Architecture of web search engine, **Web Crawling and Indexing:** Overview, Crawling, Distributing Indexes. Link analysis: Web as a graph, Page Rank.

- Pedagogy/Course delivery tools: Chalk and talk
- Links: https://www.youtube.com/watch?v=UGYhpa7Ufi0&ab_channel=TechCS%26IT

Unit V

XML retrieval: Basics and Challenges in XML retrieval, A vector space model for XML retrieval, Evaluation of XML retrieval, Web Scraping: Introduction, HTML parsing, Request library, Handling HTML forms, Basics: working with API, Excel, CSV file. Semantic Web: Purpose, Semantic Web Stack.

- Pedagogy/Course delivery tools: Chalk and talk
- Links: https://nptel.ac.in/courses/127105390

Text Books:

1. Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, An Introduction to Information Retrieval, Cambridge University Press, Cambridge, England,2009.
2. Ricardo Baeza-Yates, Berthier Ribeiro-Neto, Modern Information Retrieval, Second Edition, Pearson Education, 2024.

References:

1. Simon Munzert, Christian Rubba, Peter Meibner, Dominic Nyhuis Automated Data Collection with R: A Practical Guide to Web Scraping and Text Mining, John Wiley & Sons Inc, 2015.
2. Ricci F, Rokach, L Shapira, B.Kantor, Recommender Systems Handbook, First Edition, 2011.

Course Outcomes (COs):

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

1. Describe text operations and various information retrieval models.
2. Evaluate an IR system using various evaluation measures.
3. Apply various algorithms such as string matching, map reduce, classification, and clustering.
4. Design web search engine, web crawling, and link analysis techniques.
5. Explain XML Retrieval and various semantic web technologies.

Course Assessment and Evaluation:

Continuous Internal Evaluation (CIE): 50 Marks		
Assessment Tool	**Marks**	**Course outcomes addressed**
Internal test-I	30	CO1, CO2, CO3
Internal test-II	30	CO3, CO4, CO5
The average of the two internal tests shall be taken for 30 marks.		
Other components		
Assignment 1	10	CO1, CO2, CO3
Assignment 2	10	CO1, CO2, CO3, CO4, CO5
Semester End Examination (SEE)	100	CO1, CO2, CO3, CO4, CO5