CS-653 MJ Systems – IV (4 Credits)

Objective: This is the fourth course in the collection on systems. At the end of the course a successful student should be able to appreciate the issues involved in developing a software subsystem in which the primary activity is fundamentally concurrent and not only data but computations can be divided across multiple possibly heterogeneous computing devices.

- Overview of concurrency control – Serializable scheduling, correctness of schedules, generation of efficient and correct schedules; atomicity, consistency, isolation and durability (ACID) properties; lock-based concurrency control, transactions as a concurrency control mechanism.
- Concurrent programing - concurrency as a fundamental aspect of programing high performance systems, concurrent programing is not just parallel programing, concurrent programing is not just distributed programing.
- Composable concurrency – impossibility of composability of lock based concurrent programs, introduction to software transactional memory (STM), STM based composition.
- Distributed computations – concurrent systems with lack of global clock, Message passing model, consensus problem, mutual exclusion in distributed systems, problem of atomic commit, 2-phase commit, 3-phase commit.

References:

- Principles of concurrent and distributed programming - Mordechai Ben-Ari
- Concurrent programming: principles and practice - Gregory R Andrews
- On Concurrent Programming - Fred B. Schneider
- Concurrent Programming: Algorithms, Principles, and Foundations - Michel Raynal
- Principles of Transactional Memory (Synthesis Lectures on Distributed Computing Theory) - Rachid Guerraoui, Michal Kapalka
- Control Flow and Data Flow: Concepts of Distributed Programming: International Summer School - E. W. Dijkstra et el
- A Discipline of Multiprogramming: Programming Theory for Distributed Applications (Monographs in Computer Science) - Jayadev Misra

Note: the topic "Overview of concurrency control", should be covered in brief as an overview of topics learnt in the earlier year.

Suggested further reading:

- Functional Programming for Loosely-Coupled Multiprocessors (Research Monographs in Parallel and Distributed Computing) - Paul H. J. Kelly
- Parallel and Concurrent Programming in Haskell: Techniques for Multicore and Multithreaded Programming - Simon Marlow
- Erlang Programming: A Concurrent Approach to Software Development - Simon Thompson
- Programming ERLANG: Software for a Concurrent World - Joe Armstrong
- Communicating sequential processes - C.A.R. Hoare

CS-681 RP: Research Project II (6 credits)

- The student is expected to work for a semester (at an average of 15 hours per week) on a project assigned by the guide. Presentation will be given by the students. Student is encouraged to publish his/her work to earn extra credits.

The departmental faculty will decide on the syllabus for the following subjects which may be offered as electives in first year and Second year as per the need and demands of the current industry/research institutes' requirements. Each elective will be of 4 credits.

Title
Foundations of Artificial Intelligence
Foundations of Machine Learning
Foundations of Human Software Interface
Foundations of Software Security
Foundations of Pervasive Computing
Foundations of Natural Language Processing
Foundations of Computational Text Processing
Foundations of Computational Finance
Foundations of Data Science
Foundations of Big Data
Introduction to Database systems

Title (to be offered in later semesters)
Artificial Intelligence and Machine Learning
Block-chain technology
Parallel Algorithms
Natural Language Processing
Computational Finance
Data Science
Quantum Computing
Advanced Typing
Typing for better programs
Advanced data structures
Advanced algorithms
File system implementation
Advanced Numerical Methods
Advanced Low Level Programming
Advanced Database Systems
Higher order logic
Software Engineering
Pragmatic Software Construction
Problem Solving
Modern Web Programming