Syrian Arab Republic

Higher Institute for Applied Sciences and Technology

HIAST

Course Specification Document

Title	Parallel Computing

Credits	3.5 ECTS

Aims	This course aims to introduce the student to parallel computing and parallel programming techniques on multi-processors (using shared memory) and clusters of computers (using message-passing), enabling him to efficiently develop parallel computations on available parallel computer architectures.

Intended learning outcomes

On successful completion of this course, the student will be able to:

- Understand parallel computing and high-performance computing.
- Understand message-passing-based computing and implement it using the MPI library.
- Understand shared-memory-based computing and implement it using the OpenMP library.
- Calculate the complexity of parallel programs for various computational problems.
- Familiarize himself with different parallel computing techniques.
- Familiarize himself with parallel algorithms for various scientific problems.
- Implement software projects and testing them on a larger scale using the SimGrid environment.

Syllabus

- **Parallel computing:** The concept of parallel computation and its challenges, performance metrics for parallel programs, types of parallel computers, Flynn's taxonomy of computers.
- **High-Performance computing:** Shared memory architecture, hybrid architectures, an overview of different programming models (Cluster computing, Grid computing with examples, Desktop grid computing, and volunteer computing, cloud computing).
- **Message-Passing programming:** SPMD and MPMD models, standard communication patterns in MPI, non-standard communication patterns in MPI, collective communication operations in MPI, performance evaluation parameters for parallel programs, development and debugging tools for parallel programs.
- **Embarrassingly parallel computation:** The concept of embarrassingly parallel computation, examples with sequential and parallel implementations, complexity analysis, static and dynamic task assignment.
- **Programming using shared memory:** Initialization and constructs for synchronization and shared data, introduction to OpenMP, examples of programming using shared memory.
- **Decomposition and Divide-and-Conquer techniques:** Fundamental concepts, examples, and applications of parallel implementation for decomposition and divide-and-conquer: bucket sort, numerical integration, N-Body problem.

- **Pipeline computing:** The concept of pipeline computing and its three types, examples of each type with a complete complexity analysis.
- **Synchronization in parallel computing:** The concept of synchronization, barriers, and various synchronization mechanisms, global and local synchronization, parallel data computations with examples, synchronized iterations, examples of synchronized computations with complexity analysis, partial synchronization.
- **Digital algorithms:** Matrix multiplication, solving linear equations, faster convergence methods, parallel sorting algorithms.
- **Image processing algorithms:** Low-level image processing.
- **Sorting algorithms:** Implementing some sorting algorithms in both sequential and parallel versions.
- **Load balancing and termination detection:** Different load balancing models, distributed termination detection algorithms, applications.
- **Search and optimization:** Sequential and parallel versions of branch-and-bound, genetic algorithms, hill-climbing.
- **Applying parallel programming techniques on cluster computers using MPI library.**
- **Applying parallel programming techniques on multi-processor architectures using OpenMP library.**
- **Implementing and testing software projects on a larger scale using the SimGrid environment.**