AMERICAN UNIVERSITY
OF CYPRUS

Course Code	**Course Name**	**ECTS Credit**
MSE 362	Parallel Systems	7.5
Pre-Requisite	**Course Type**	**Language of Instruction**
	Major Elective	English
Year of Study	**Level of Course**	**Mode of Delivery**
4th / 8th	BSc/1st Cycle	On Campus

Course Objectives:

The objective of the course is to present state of the art parallel systems and architectures and their main programming techniques and tools. Also to learn basic parallelization techniques for distributed memory environments and to design and implement corresponding algorithms for classical computational problems. Especially the objective is to provide the necessary knowledge and skills for the development of efficient parallel applications in modern parallel environments

Learning Outcomes:

Upon successful completion of the course, students will be able to:

- Recognize the most important of the modern categories/ architectures of parallel systems and describe their main features
- Distinguish between parallelism techniques in shared and distributed memory environments
- Design parallel algorithms for distributed memory environments
- Describe the different models of parallel programming models and distinguish their differences and main advantages
- Apply the most important of the taught programming models in practice
- Use modern parallel programming tools to implement efficient parallel applications
- Program in modern parallel environments, such as multicore systems, supercomputers, clusters, NOWs, etc.
- Measure the performance of implemented parallel and applications with modern measures, techniques and tools

Teaching Methodology:

Lectures 42 hours

Labs 30 hours

Course Content

Modern Parallel Systems and Architectures – supercomputers, multicore systems clusters, hybrid systems and architectures, GPUs

Parallelization Techniques in Distributed Memory Environment

Solution of classical computational problems in distributed memory environment (sorting algorithms, matrix multiplication algorithms and algorithms for solving linear systems)

Parallel Programming Models & Design of Parallel Programs

Programming in shared address space (OpenMP), programming by exchanging messages (MPI) and combining the above (hybrid model).

Programming GPUs (CUDA programming model)

Programming in clusters & NOWs / MPI, Condor environments

Measure the Performance of Parallel Programs

Assessment Methods:

Final Exam

Mid-term/Lab Exam

Assignments

Required Textbooks/Reading:

Title	**Author(s)**	**Publisher**	**Year**
Introduction to Parallel Computing	Z.J. Czech	Cambridge University	2017
Parallel Programming in C with MPI and OpenMP	Quinn M.J.	McGraw-Hill	2003