COURSE OUTLINE

(1) GENERAL

SCHOOL	SCHOOL OF SCIENCES		
ACADEMIC UNIT	DEPARTMENT OF MATHEMATICS		
LEVEL OF STUDIES	UNDERGRADUATE PROGRAM		
COURSE CODE		**SEMESTER**	**E**
COURSE TITLE	MATHEMATICAL SOFTWARE		
INSTRUCTOR			
INDEPENDENT TEACHING ACTIVITIES		**WEEKLY TEACHING HOURS**	**CREDITS**
		3	4,5
COURSE TYPE	Specialised general knowledge		
PREREQUISITE COURSES:	NO		
LANGUAGE OF INSTRUCTION and EXAMINATIONS:	GREEK		
IS THE COURSE OFFERED TO ERASMUS STUDENTS	YES		
COURSE WEBSITE (URL)	http://www.math.aegean.gr/index.php/en/academics/undergraduate-programs		

(2) LEARNING OUTCOMES

Learning outcomes
This course introduces students to the basic mathematical modeling with the use of Mathematica and MATLAB. It consists of two parts: At the first part, the students are introduced to the basic structures of Mathematica: front-end, kernel, notebooks, data types, arithmetic computations, symbolic computations, graphic presentations and programming methods (procedural, functional, rule-based programming), built-in functions. These structures are used in the development of a model for various applications. At the second part, the students are introduced to the basic structures of MATLAB: variables, operators, flow control structures, matrices, vectors, arithmetic computations, built-in functions for vectors and matrices, programming methods and script files. These structures are used in the development of a model for various applications. The aim of this course is to introduce the students into the mathematical modelling by using the Mathematica and MATLAB which the most popular software tools for modelling. At the end of the course the students will be able to model simple mathematical problems.
General Competences
Working independently. Team working. Working in an interdisciplinary environment. Working in an international environment.

(3) SYLLABUS

Introduction to *MATHEMATICA*: The structure of MATHEMATICA (front end and kernel), variables, constants, operators, math types, arithmetic and symbolic computations, and some important system functions. Development of user functions, list Processing and use of Help Browser.

Procedural Programming: assignment and delayed assignment, input-output command, flow control commands, commands for loops and nested loops and the command Continue and Break. How to create a Module.

Functional Programming: Use of functions FullForm, TreeForm and Level, Map, Head, Apply, Through, Thread, MapThread, Nest, NestList.

Rule-based Programming: Global rules, use of functions Replace All, Replace Repeated, Replace, Rule, Rule Delayed, local rules, rules priority, conditional, unconditional and unlabeled patterns.

Polynomials: Functions that manipulates polynomial expressions and rational functions manipulation. Solving Equations and Systems. Matrix Operations, eigenvalues and eigenvectors of a matrix and diagonalization of a matrix.

Computation of function limit, derivative and integral. Series, solving differential equations and graphical representations.

Introduction to *MATLAB:* simple arithmetic calculations, matrix creation and manipulation, creating and using script files. Variables, constants, data types and operators.

Programming Structures of MATLAB: Basic embedded functions, loop commands, flow control commands. Functions that process vectors and matrices with emphasis to functions that are used to arithmetic calculations. Graphical representations in MATLAB.

Modular and Structural Programming in MATLAB: user defined functions, parameter passing, and connecting different functions. Solving problems from Linear Algebra and Number Theory.

Solving Linear Systems: Gauss, Gauss-Jordan, LU. Iterative Methods: Jacobi, Gauss-Seidel.

Data Interpolation: Lagrange, Direct Method, Cubic Splines, Cubic Hermite polynomials.

Data Approximation: Min Square Method.

Derivative and Integral of a function (using symbolic variables). Differentiation and Integration of piecewise polynomials. Solving ordinary differential equations.

TEACHING MATERIAL DISTRIBUTION	The teaching material of the course is uniformly distributed during the semester.

(4) TEACHING and LEARNING METHODS - EVALUATION

DELIVERY	Face-to-face lectures
USE OF INFORMATION AND COMMUNICATIONS TECHNOLOGY	• Use of ICT in teaching • Communication with students via e-mail. • Uploading course material on moodle system.
TEACHING METHODS	

Activity	***Semester workload***
Lectures	26
Laboratory practice	13
Independent study	73,5
Course total (25 per ECTS)	***112,5***

COURSE COMMITMENTS	Attending course and lab sessions is not obligatory.
STUDENT PERFORMANCE EVALUATION	Student's evaluation is done in Greek through a written examination which includes short-answers questions, problem solving and laboratory work. For students with disabilities, evaluation takes place via oral exams.

(5) ATTACHED BIBLIOGRAPHY

1. Contemporary Mathematical Software MATLAB-MATHEMATICA, Papageorgiou G., Tsitouras X., Famelis I, Symeon Editions, Athens 2004.