

(1) GENERAL

SCHOOL	SCHOOL OF INFORMATION SCIENCES & TECHNOLOGY		
ACADEMIC UNIT	DEPARTMENT OF STATISTICS		
LEVEL OF STUDIES	POSTGRADUATE		
COURSE CODE		TRIMESTER	1
COURSE TITLE	Introduction to R and Python		
INDEPENDENT TEACHING ACTIVITIES		WEEKLY TEACHING HOURS	CREDITS
		3	5
		30	5
COURSE TYPE	Specialization		
PREREQUISITE COURSES:			
LANGUAGE OF INSTRUCTION and EXAMINATIONS:	ENGLISH		
COURSE DELIVERY METHOD	Distance		
IS THE COURSE OFFERED TO ERASMUS STUDENTS	NO		
COURSE WEBSITE (URL)	https://cloud.aueb.gr/index.php/s/4Rzjft7BXJbDE7r		

(2) LEARNING OUTCOMES

Learning outcomes
• Understanding the basic principles of programming and algorithm development. • Become familiar with the R environment and the management of objects and data. • Application of arithmetic operations, loops, and functions in R. • Development of skills in data visualization through charts and graphs in R. • Understanding the basic syntax and data types of Python. • Use of conditional structures, loops, and functions in Python. • Introduction to basic data processing techniques, such as reading/writing files, sorting, and filtering. • Implementation of a simple, integrated application using fundamental programming concepts
General Competences
• Searching, analyzing, and synthesizing data and information, including the use of necessary technologies. • Early-stage data processing. • Ability to work independently. • Ability to work in an international environment.

(3) SYLLABUS

Fundamentals of Programming. Introduction to R: basic package elements, command-line interface, and graphical interface. Arithmetic operations and expressions. Objects, object types, and data types. Loop syntax: for, while, and repeat loops. Program creation and result lists. Special commands. Plotting in R, including multiple plots. Functions. Introduction to Python. History, uses, and advantages. Basic syntax. Variables and data types: numbers, strings, lists, tuples, and dictionaries. Basic operations: assignment, arithmetic operations, printing text. Conditional statements: if-else. Loops: for and while. Creating and calling functions. Introduction to data processing: reading and writing files, data manipulation (sorting, filtering, etc.). Application of programming concepts: development of a simple end-to-end project as an example.

(4) TEACHING and LEARNING METHODS - EVALUATION

DELIVERY <i>Face-to-face, Distance learning, etc.</i>	Distance	
METHOD AND FREQUENCY OF COMMUNICATING WITH THE STUDENTS	Remotely via email and through weekly office hours (and whenever necessary) via teleconferencing or in-person office visits.	
ENSURING COMMUNICATION AMONG STUDENTS	Teleconference, Chat via eclass and/ or TEAMS, and QA sessions	
USE OF INFORMATION AND COMMUNICATIONS TECHNOLOGY	MS Teams, e-class, MS Outlook, R, Python, WayGround or Kahoot educational games and quizzes	
REQUIRED EQUIPMENT AND TECHNOLOGY KNOWLEDGE	Camera, mic, PC, MS Office, and TEAMS	
COURSE POLICY ON PLAGIARISM AND PLAGIARISM DETECTION TOOLS	Turnitin	
COURSE POLICY ON USE OF AI TOOLS	The use of Artificial Intelligence tools is prohibited in all cases (1).	
TEACHING METHODS	Activity	Semester workload
	Lectures	30
	Assignment writing	40
	Lab Exercise	20
	Self-Study hours	40
	Course total	130
STUDENT PERFORMANCE EVALUATION	Evaluation language is English. Evaluation methods include lab exercises, written assignments, and/or written examinations. Oral examinations of the assignments may be conducted if clarifications are needed or if there are suspicions of plagiarism or unauthorized use of Artificial Intelligence tools. Assessment criteria are provided on the course eClass platform and in the course materials.	

(5) RECOMMENDED BIBLIOGRAPHY

- Wickham, H., & Grolemund, G. (2017). R for Data Science. O'Reilly Media.
- Matthes, E. (2019). Python Crash Course: A Hands-On, Project-Based Introduction to Programming. No Starch Press.