Study programme(s): Applied Mathematics – Data Science

Level: Master studies

Course title: Programming for Data Science

Lecturer: Nataša Krklec Jerinkić

Status: obligatory

ECTS: 6

Requirements: Linear Algebra, Basic Programming Skills

Learning objectives

- Introducing the fundamental principles of data science and data-analytic thinking
- Learning Python coding skills for modelling and analysing of a broad range of datasets - numerical, string, and more complex data formats
- Translate a simple algorithm into a Python code
- Learning how to effectively visualise results

Learning outcomes

- Introduction into analysis and processing of data
- Ability to write scripts in Python with basic programming concepts like loops, arrays, dictionaries, strings, if statements, functions and classes
- Exploratory data analysis: create plots and summary statistics
- Develop practical skills in problem solving by working on diverse data

Syllabus

- Develop skills necessary to use Python for data analysis:
  - Learn data structures: lists, tuples, dictionaries
  - Learn to write, test, and debug Python code
  - Learn scientific libraries in Python: NumPy (multidimensional array objects, linear algebra operations), SciPy (matrix decompositions, sparse matrices, statistical tests), Networkx (structure and analysis measures for graphs), Pandas (structured data, slicing, aggregating, and selecting subsets of data), Seaborn and Matplotlib (drawing attractive statistical graphics and visualizations)
- Develop skills necessary for data-driven applications and decision making

Literature

16. Pang-Ning Tan, Michael Steinbach, Vipin Kumar, "Introduction to data mining", Pearson Addison Wesley, 2006.
17. Wes McKinney, "Python for Data Analysis, O'Reilly Media", 2012.
18. Ron Zacharski, "A Programmer's Guide to Data Mining", 2012.

Weekly teaching load				Other: 0
Lectures: 2	Exercises: 3	Other forms of teaching: 0	Student research: 0	

Teaching methodology

Lectures; revisions of the material; active students' participation in problem solving; homework assignments; application of the taught material on real-world examples.

Grading (maximum number of points 100)

Pre-exam obligations		**Points**	**Final exam**	**points**
Programming Test	Solved homework assignments	60 = 40 (Homework assignments) + 20 (Test)	Preparation and defence of course project	40