PYTHON and MATLAB Programming

Course description sheet

Basic information

Field of study Geospatial Computer Science	**Didactic cycle** 2024/2025
Major Remote Sensing and GIS	**Course code** DGEITGS.IIi1.07196.24
Organisational unit Faculty of Geo-Data Science, Geodesy, and Environmental Engineering	**Lecture languages** English
Study level Second-cycle (engineer) programme	**Mandatoriness** Obligatory
Form of study Full-time studies	**Block** Major Modules
Profile General academic	**Course related to scientific research** Yes

Course coordinator	Mariusz Twardowski
Lecturer	Mariusz Twardowski

Period Semester 1	**Method of verification of the learning outcomes** Completing the classes **Activities and hours** Lectures: 15 Project classes: 30	**Number of ECTS credits** 3

Course's learning outcomes

Code	Outcomes in terms of	Learning outcomes prescribed to a field of study	Methods of verification
Knowledge - Student knows and understands:			
W1	rules and differences of Python and Matlab programming language.	GEI2A_W02	Test

Code	Outcomes in terms of	Learning outcomes prescribed to a field of study	Methods of verification
Skills – Student can:			
U1	create programs using Python language.	GEI2A_U02	Test
U2	extend geoinformation tools functionality using Python language.	GEI2A_U02	Test
Social competences – Student is ready to:			
K1	Creativity in programming.	GEI2A_K03	Activity during classes

Program content ensuring the achievement of the learning outcomes prescribed to the module

The module allows to acquire familiarity with Python language in comparison with Matlab for solving computing problems.

Student workload

Activity form	Average amount of hours* needed to complete each activity form
Lectures	15
Project classes	30
Realization of independently performed tasks	25
Contact hours	1
Preparation of project, presentation, essay, report	10
Student workload	**Hours** 81
Workload involving teacher	**Hours** 45

* hour means 45 minutes

Program content

No.	Program content	Course's learning outcomes	Activities
1.	Lectures: 1. Introduction to Python Interpreter basics. Other programming languages comparison. Variable assignment, mathematical and logical operators. Base data types and their representation. Keywords and complex structures. Mutable objects. 2. Scripts and program flow control Correct script formatting. Conditional structures, loops and function definitions. Difference between interpreter and compiler. Exception handling. Input/output procedures. 3. Classes, objects, and modules Introduction to object-oriented programming in Python. Class, object and method construction, inheritance and polymorphism. Script modularisation methods and library import. 4. Language libraries selection Standard libraries and most important elements. Geospatial libraries GDAL, OGR, OSR. Reading, writing and processing raster and vector data. 5. QT interface introduction. Rapid Application Development tools, automatic code generation, connecting interface elements with implementation code. Event-driven programming. 6. Python as a Matlab replacement introduction to PIL and matplotlib libraries. Data Visualization algorithms.	W1, K1	Lectures
2.	Project classes: 1. Introduction to Python Active interpreter usage for variable assignment and expression evaluation. Type checking. String slicing. A list, set and dictionary allocation. Keyword usage. 2. Scripts and program flow control. Script authoring tools. Error correction and interpretation. Conditionals, loops and exception handling. 3. Classes, objects, and modules. Class creation and method implementation. Inheritance examples. Object initialization and designator usage in method calls. Module import. 4. Selected libraries. Common standard library examples. Spatial library usage. Raster, vector and text data operations. 5. QT interface design introduction. Basic GUI design tool usage. Tools for automatic code generation. Connecting basic interface elements with methods. Simple interface testing. 6. Visualization libraries. Connecting PIL and Matplotlib libraries with QT interface. Usage of libraries in geoinformation problems.	U1, U2, K1	Project classes

Extended information/Additional elements

Teaching methods and techniques :

Lectures

Activities	Methods of verification	Credit conditions
Lectures	Activity during classes, Test	
Project classes	Activity during classes, Test	

Conditions and the manner of completing each form of classes, including the rules of making retakes, as well as the conditions for admission to the exam

Ćwiczenia zaliczane są na podstawie kolokwium. W przypadku braku zaliczenia kolokwium możliwa będzie jego poprawa.

Method of determining the final grade

Lectures and project classes final grade will be based on the final test at the end of the semester. In case of a failed attempt, there will be possible 2 more tries. Retrying failed attempt does not have the impact on final grade.

Manner and mode of making up for the backlog caused by a student justified absence from classes

W przypadku nieobecności na wykładach lub ćwiczeniach student zobowiązany jest nadrobić materiał samodzielnie.

Prerequisites and additional requirements

Ability to understand English.
Computer usage knowledge.
Any basic computer language familiarity.
Ability to use a web browser.

Rules of participation in given classes, indicating whether student presence at the lecture is obligatory

Lectures: Studenci uczestniczą w zajęciach poznając kolejne treści nauczania zgodnie z syllabusem przedmiotu. Studenci winni na bieżąco zadawać pytania i wyjaśniać wątpliwości. Rejestracja audiowizualna wykładu wymaga zgody prowadzącego.
Project classes: Studenci wykonują prace praktyczne mające na celu uzyskanie kompetencji zakładanych przez syllabus. Ocenie podlega sposób wykonania projektu oraz efekt końcowy.

Literature

Obligatory

1. Literatura:
2. 1. Lutz Mark: "Learning Python, 5th edition". 2013,
3. 2. Dawson Michael: "Python Programming for the Absolute Beginner, 3rd Edition", 2010
4. 3. Documentation and online courses: http://pl.python.org
5. 4. Summerfield Mark: "Rapid GUI Programming with Python and Qt". Prentice Hall 2008
6. 5. PyQt documentation: http://pyqt.sourceforge.net/Docs/PyQt4/

Scientific research and publications

Publications

1. 1. K. Pyka, M. Twardowski: “Miejsce wolnego oprogramowania w nauczaniu geoinformatyki”. Archiwum Fotogrametrii, Kartografii i Teledetekcji. 2007.

2. 2. K.Pyka, M. Słota, M. Twardowski -“Usage of stereo orthoimage in GIS: old concept, modern solution”. XXII ISPRS congress. 2012

Learning outcomes prescribed to a field of study

Code	Content
GEI2A_K03	aktywnego i kreatywnego włączenia się w dynamiczny rozwój geoinformacji, wzmacniania jej roli w społeczeństwie oraz popularyzowania powszechnego korzystania z danych przestrzennych.
GEI2A_U02	programować, modyfikować i rozbudowywać istniejące aplikacje oraz łączyć różne technologie informatyczne w zakresie geoinformacji.
GEI2A_W02	złożone zasady programowania oraz konstruowania algorytmów, niezbędne do czytania ze zrozumieniem, pisania, uruchamiania i weryfikacji programów.