

# Python language in numerical calculations - opis przedmiotu

Informacje ogólne	
Nazwa przedmiotu	Python language in numerical calculations
Kod przedmiotu	13.2-WF-FizP-PraZa-S17
Wydział	<a href="#">Wydział Fizyki i Astronomii</a>
Kierunek	Fizyka
Profil	ogólnoakademicki
Rodzaj studiów	Program Erasmus pierwszego stopnia
Semestr rozpoczęcia	semestr zimowy 2017/2018

Informacje o przedmiocie	
Semestr	5
Liczba punktów ECTS do zdobycia	6
Typ przedmiotu	obowiązkowy
Język nauczania	angielski
Sylabus opracował	<ul style="list-style-type: none"><li>dr Marcin Kośmider</li></ul>

Formy zajęć					
Forma zajęć	Liczba godzin w semestrze (stacjonarne)	Liczba godzin w tygodniu (stacjonarne)	Liczba godzin w semestrze (niestacjonarne)	Liczba godzin w tygodniu (niestacjonarne)	Forma zaliczenia
Wykład	30	2	-	-	Egzamin
Laboratorium	30	2	-	-	Zaliczenie na ocenę

## Cel przedmiotu

The course aim is to introduce the Python as the scientific programming tool. Python is a general purpose, high-level and modern programming language and the capabilities of its standard library as well as the external modules to handle the numerical analysis in physics and related fields will be presented.

## Wymagania wstępne

Basic knowledge in programming and object oriented programming.

## Zakres tematyczny

### Metody kształcenia

1) General Python introduction

- Language syntax and data types

- Flow-control and exceptions

- Interactive shell

- Scripts

- Functions

- Modules

2) File I/O operations

- Writing to and saving files

- Data serialization

- Typical I/O operations errors

3) Object Oriented Programming

- Classes and objects

- Inheritance and polymorphism

- Abstractions

4) Introduction to software engineering

- Version control systems

- Linux as IDE
- Introduction to unit-tests
- Software efficiency and profiling

## 5) Numerical analysis and computer simulations introduction

- The math module
- NumPy's arrays
- Random numbers
- Basic linear algebra operations in NumPy
- Differential equations solvers in NumPy

- Data visualisations in the matplotlib module

- Introduction to parallel computing with mpi4py

## 6) Visualization, animations and image processing

- The canvas and graphical primitives
- Plots
- Animations
- Image processing with openCV (computer vision) module

## Efekty uczenia się i metody weryfikacji osiągnięcia efektów uczenia się

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Student is able to use Python and its standard library to implement a simple software that numerically solves a classical physics problem.		<ul style="list-style-type: none"> <li>bieżąca kontrola na zajęciach</li> <li>dyskusja</li> <li>egzamin - ustny, opisowy, testowy i inne</li> <li>projekt</li> </ul>	<ul style="list-style-type: none"> <li>Wykład</li> <li>Laboratorium</li> </ul>
Student is able to find out, learn and use the external Python libraries that will help him with the numerical analysis of physical problems.		<ul style="list-style-type: none"> <li>bieżąca kontrola na zajęciach</li> <li>dyskusja</li> <li>egzamin - ustny, opisowy, testowy i inne</li> <li>projekt</li> </ul>	<ul style="list-style-type: none"> <li>Wykład</li> <li>Laboratorium</li> </ul>
Student is able to perform graphical data analysis and data visualization using Python and its modules.		<ul style="list-style-type: none"> <li>bieżąca kontrola na zajęciach</li> <li>dyskusja</li> <li>egzamin - ustny, opisowy, testowy i inne</li> <li>projekt</li> </ul>	<ul style="list-style-type: none"> <li>Wykład</li> <li>Laboratorium</li> </ul>
Student is able to solve and present (in spoken and printed form) the outcomes of assigned project.		<ul style="list-style-type: none"> <li>bieżąca kontrola na zajęciach</li> <li>dyskusja</li> <li>projekt</li> </ul>	<ul style="list-style-type: none"> <li>Laboratorium</li> </ul>

## Warunki zaliczenia

### Lecture:

To pass the exam the student will be asked to numerically solve a certain problem of the classical physics or data analysis. The examined knowledge fields and the final exam grade will be evaluated using the following aspects: the problem analysis, presentation of the algorithms used in the problem solution, the presentation of the source code and the validity of the results.

### Laboratory:

30% - tests ad activity during laboratories

70% - final project

Before taking the exam the student must obtain a pass from the laboratory.

**Score:** weighted average rating of the exam (60%) and exercise (40%).

## Literatura podstawowa

[1] Mark Lutz, Python. Wprowadzenie, Wydanie IV, Helion, Gliwice 2010.

[2] <http://python.org>

[3] <http://python-ebook.blogspot.com/>

[4] <http://numpy.scipy.org>

[5] Hans Petter Langtangen, A primer on scientific programming with Python, Springer, Berlin 2009.

## Literatura uzupełniająca

[1] Internet

## Uwagi

Zmodyfikowane przez dr hab. Maria Przybylska, prof. UZ (ostatnia modyfikacja: 09-07-2018 23:01)

Wygenerowano automatycznie z systemu SylabUZ