Politechnika Krakowska im. Tadeusza Kościuszki

Karta przedmiotu

obowiązuje studentów rozpoczynających studia w roku akademickim 2020/2021

Wydział Inżynierii Materiałowej i Fizyki

Kierunek studiów: Fizyka Techniczna w Języku Angielskim — Profil: Ogólnoakademicki

Forma sudiów: stacjonarne — Kod kierunku: FTja

Stopień studiów: II

Specjalności: Computer modelling (modelowanie komputerowe w języku angielskim)

1 Informacje o przedmiocie

Nazwa przedmiotu	Introduction to Python
Nazwa przedmiotu w języku angielskim	Introduction to Python
Kod przedmiotu	WIMiF FTJA oIIS F10 20/21
Kategoria przedmiotu	Przedmioty wybieralne
Liczba punktów ECTS	3.00
Semestry	1

2 Rodzaj zajęć, liczba godzin w planie studiów

Semestr	Wykład	Ćwiczenia	Laboratorium	Laboratorium komputerowe	Seminarium	Projekt
1	30	0	0	15	0	15

3 Cele przedmiotu

Cel 1 Main aim of this course is to learn basic grammar and data structures of Python programming language.

Cel 2 Learning how to use basic scientific libraries.

Cel 3 Learning soft skills: team-work, communication in team.

4 Wymagania wstępne w zakresie wiedzy, umiejętności i innych kompetencji

1 Knowledge of how to use computer.

2 Knowledge of some programming language will be helpful but not necessary level will be adjusted to audience needs.

5 Efekty kształcenia

EK1 Wiedza Student knows basic grammar and data structures of Python programming language.

EK2 Umiejętności Student knows how to use basic grammar and data structures of Python programming language.

EK3 Wiedza Student knows selected libraries used in scientific computing.

EK4 Umiejętności Studnet knows how to apply selected libraries used in scientific computing.

EK5 Kompetencje społeczne Student has practice of team-work and communication in group.

6 Treści programowe

Wykład		
Lp	Tematyka zajęć Opis szczegółowy bloków tematycznych	Liczba godzin
W1	1. Introduction to programming History of computers and programming languages. How computers work von Neumann architecture, CPU, memory, system bus, IO devices. Algorithm. Programming paradigms: procedural, object oriented, and functional. Examples of languages and their use in IT and science. Difference between compiled and interpreted languages.	2
W2	2. Technical issues. Philosophy of Python. Webpage of Python. Basic software for Python programming for Windows, Linux and Android - installation, configuration and overview. Jupyter notebooks, Spyder IDE. Work with prompt. How to run Python scripts - Hello world.	2
W3	3. Basic Python. Variables. Dynamic (Duck typing) vs Static typing. Basic datatypes: integers, floats, strings, booleans. Making decisions if-else.	2
W4	4. Sequences Lists, Tuples, Dictionaries, Strings as tuples. Formating strings.	2
W5	5. Loops For loop, While loop, continue, break, Examples.	2
W6	6. Functions and procedural programming Defining functions. Global and local scope of variables. global - word. Default arguments.	2
W7	7. Working with files File object, opening files modes: read, write, append. Reading and writing files efficient techniques. Closing files.	2
W8	8. Exceptions and modules Catching and rising exceptions. Writing fault-safe code. Importing modules and packages. Creating libraries splitting code into modules. Use of self-defined modules.	2

Wykład		
Lp	Tematyka zajęć Opis szczegółowy bloków tematycznych	Liczba godzin
W9	9. Functional programing Lambda functions, map, reduce, filter as core of functional programming. Memorization.	2
W10	10. List comprehension and recursion List comprehension. Effective recursion programming. Tail recursion.	2
W11	11. Object Oriented Programming Class and objects attributes and methods. Encapsulation. Inheritance. Simple examples.	2
W12	12. Introduction to Numpy library Arrays, indexing, slicing, remarks on speed.	2
W13	13. Common computational tasks SciPy library Interpolation, Root finding, Model fitting, Numerical integration.	2
W14	14. ODE solutions using SciPy Ordinary Differential Equations theory and basic numerical schemes: Euler, Runge-Kutta of 4th order. Methods of integration from SciPy library. Examples: first order equations, harmonic oscillator, Lotka-Volterra model, Zombie apocalypse model.	2
W15	15. Students talks on selected topics. Presentations and discussion.	2

Laboratorium komputerowe		
Lp	Tematyka zajęć Opis szczegółowy bloków tematycznych	Liczba godzin
K1	1. Introduction to programming History of computers and programming languages. How computers work von Neumann architecture, CPU, memory, system bus, IO devices. Algorithm. Programming paradigms: procedural, object oriented, and functional. Examples of languages and their use in IT and science. Difference between compiled and interpreted languages.	1
K2	2. Technical issues. Philosophy of Python. Webpage of Python. Basic software for Python programming for Windows, Linux and Android - installation, configuration and overview. Jupyter notebooks, Spyder IDE. Work with prompt. How to run Python scripts - Hello world.	1
K3	3. Basic Python. Variables. Dynamic (Duck typing) vs Static typing. Basic datatypes: integers, floats, strings, booleans. Making decisions if-else.	1
K4	4. Sequences Lists, Tuples, Dictionaries, Strings as tuples. Formating strings.	1
K5	5. Loops For loop, While loop, continue, break, Examples.	1
K6	6. Functions and procedural programming Defining functions. Global and local scope of variables. global - word. Default arguments.	1
K7	7. Working with files File object, opening files modes: read, write, append. Reading and writing files efficient techniques. Closing files.	1

Laboratorium komputerowe		
Lp	Tematyka zajęć Opis szczegółowy bloków tematycznych	Liczba godzin
K8	8. Exceptions and modules Catching and rising exceptions. Writing fault-safe code. Importing modules and packages. Creating libraries splitting code into modules. Use of self-defined modules.	1
K9	9. Functional programing Lambda functions, map, reduce, filter as core of functional programming. Memorization.	1
K10	10. List comprehension and recursion List comprehension. Effective recursion programming. Tail recursion.	1
K11	11. Object Oriented Programming Class and objects attributes and methods. Encapsulation. Inheritance. Simple examples.	1
K12	12. Introduction to Numpy library Arrays, indexing, slicing, remarks on speed.	1
K13	13. Common computational tasks SciPy library Interpolation, Root finding, Model fitting, Numerical integration.	1
K14	14. ODE solutions using SciPy Ordinary Differential Equations theory and basic numerical schemes: Euler, Runge-Kutta of 4th order. Methods of integration from SciPy library. Examples: first order equations, harmonic oscillator, Lotka-Volterra model, Zombie apocalypse model.	1
K15	Introduction to Algorithms and Data Structures.	1

Projekt		
Lp	Tematyka zajęć Opis szczegółowy bloków tematycznych	Liczba godzin
P1	Basic methods of team-work.	5
P2	Selection and Analysis of real-world problem that can be solved using computers (Python and libraries).	5
P3	Team-work on selected problem. Preparing final presentation.	5

7 Narzędzia dydaktyczne

N1 Wykłady

N2 Dyskusja

N3 Prezentacje multimedialne

N4 Ćwiczenia laboratoryjne

N5 Praca w grupach

N6 Ćwiczenia projektowe

N7 Konsultacje

8 Obciążenie pracą studenta

Forma aktywności	Średnia liczba godzin na zrealizowanie aktywności
Godziny kontaktowe z nauczycielem akademickim, w tym:	
Godziny wynikające z planu studiów	60
Konsultacje przedmiotowe	0
Egzaminy i zaliczenia w sesji	0
Godziny bez udziału nauczyciela akademickiego wynikające z nakładu pracy studenta, w tym:	
Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	15
Opracowanie wyników	10
Przygotowanie raportu, projektu, prezentacji, dyskusji	5
Sumaryczna liczba godzin dla przedmiotu wynikająca z całego nakładu pracy studenta	**90**
Sumaryczna liczba punktów ECTS dla przedmiotu	3.00

9 Sposoby oceny

A1: Homeworks and laboratory exercises: Student is required to work out simple exercises given during the lectures.
A2: Final project: Student is required to prepare a presentation on one of the subjects connected with the subject.
Final grade = Average(Average(A1) + Average(A2))

Ocena formująca

F1 A1: Homeworks and laboratory exercises

F2 A2: Final project

Ocena podsumowująca

P1 Final grade = Average(Average(A1) + Average(A2))

Warunki zaliczenia przedmiotu

W1 Final grade $>$ 2

Ocena aktywności bez udziału nauczyciela

B1 A1 and A2

Kryteria oceny

Efekt kształcenia 1	
Na ocenę 2.0	Student does not know basic grammar and data structures of Python programming language.
Na ocenę 3.0	Student knows how basic programs that uses flow-control constructs works and can describe basic data structures.
Na ocenę 3.5	In addition, Student knows how modules and exeptions works.
Na ocenę 4.0	Student knows adavnced concepts of Python programming as Objects and terms from Functional Programming. Student can describe details of sceintific libraries.
Na ocenę 4.5	In addition, Student can read advanced programs.
Na ocenę 5.0	Student knows Python programming fulently.
Efekt kształcenia 2	
Na ocenę 2.0	Student does not know how to use basic grammar and data structures of Python programming language.
Na ocenę 3.0	Student knows how to write basic programs that uses flow-control constructs and can use basic data structures.
Na ocenę 3.5	In addition, Student know how to make modules and how to use them. Can use excepions.
Na ocenę 4.0	Student can use adavnced concepts of Python programming as Objects and terms from Functional Programming. Student can use sceintific libraries on a good level.
Na ocenę 4.5	In addition, Student can read advanced programs and modify them.
Na ocenę 5.0	Student can write advanced Python programs.
Efekt kształcenia 3	
Na ocenę 2.0	Student does not know selected libraries used in scientific computing.
Na ocenę 3.0	Student knows basics selected libraries used in scientific computing.
Na ocenę 3.5	Student can know how to find help in the Internet on selected component of libraries.
Na ocenę 4.0	Student know how to create basic programs using sceintific libraries.
Na ocenę 4.5	Student know how to use advanced features of scientitfic libraries.
Na ocenę 5.0	Students know how to write effective programs that use scientific libraries.
Efekt kształcenia 4	
Na ocenę 2.0	Student does not know how to use selected libraries used in scientific computing.
Na ocenę 3.0	Student knows how to write program that uses basics selected scientific libraries.

Na ocenę 3.5	Student can know how to find help in the Internet on selected component of libraries and use it to write basic programs.
Na ocenę 4.0	Student know how to write and run basic programs using sceintific libraries.
Na ocenę 4.5	Student use advanced features of scientitfic libraries.
Na ocenę 5.0	Students can write effective programs that use scientific libraries.
Efekt kształcenia 5	
Na ocenę 2.0	Student deas not know basic principles of team work.
Na ocenę 3.0	Student know principles of team-work and can work in group efficiently.

10 Macierz realizacji przedmiotu

Efekt kształcenia	Odniesienie danego efektu do szczegółowych efektów zdefiniowanych dla programu	Cele przedmiotu	Treści programowe	Narzędzia dydaktyczne	Sposoby oceny
EK1	K_W02b K_W05 K_U01b	Cel 1	W1 W2 W3 W4 W5 W6 W7 W8 W9 W10	N1 N2 N3 N6 N7	F1 F2 P1
EK2	K_U01b K_U02 K_U07b K_U08b K_K01 K_K03	Cel 1	K1 K2 K3 K4 K5 K6 K7 K8 K9 K10 P1 P2 P3	N2 N4 N5 N6 N7	F1 F2 P1
EK3	K_W02b K_W03 K_W05	Cel 2	W11 W12 W13 W14	N1 N2 N3 N7	F1 F2 P1
EK4	K_U01b K_U02 K_U04b K_U07b K_U08b K_K01 K_K03	Cel 2	K11 K12 K13 K14 K15 P1 P2 P3	N4 N5 N6 N7	F1 F2 P1
EK5	K_K01 K_K02 K_K03 K_K04	Cel 3	K1 K2 K3 K4 K5 K6 K7 K8 K9 K10 K11 K12 K13 K14 K15 P1 P2 P3	N4 N5 N6 N7	F1 F2 P1

11 Wykaz literatury

Literatura podstawowa

[1] M. Lutz — *Learning Python*, , 2013, O'Reilly Media

[2] H. Fangohr — *Computational Science and Engineering in Python*, , 2019, University of Southampton

[3] Various authors — *Scipy Lecture Notes*, , 2019,

Literatura uzupełniająca

[1] Various authors — *SciPy Cookbook*, , 2019,

12 Informacje o nauczycielach akademickich

Osoba odpowiedzialna za kartę

dr inż. Radosław Kycia (kontakt: `rkycia@pk.edu.pl`)

Osoby prowadzące przedmiot

1 dr Radosław Kycia (kontakt: `rkycia@pk.edu.pl`)

13 Zatwierdzenie karty przedmiotu do realizacji

(miejscowość, data) (odpowiedzialny za przedmiot) (dziekan)

Przyjmuję do realizacji (data i podpisy osób prowadzących przedmiot)

. . . . . .