

DESCRIPTION OF THE COURSE OF STUDY

Course code	0719-2DE-C22-OOP	
Name of the course in	Polish	Programowanie obiektowe
	English	Object Oriented Programming

1. LOCATION OF THE COURSE OF STUDY WITHIN THE SYSTEM OF STUDIES

1.1. Field of study	Data Engineering
1.2. Mode of study	Full-time
1.3. Level of study	Undergraduate engineering study
1.4. Profile of study	General academic
1.5. Person/s preparing the course description	Dr. Eng. Przemysław Ślusarczyk
1.6. Contact	pslusarczyk@ujk.edu.pl

2. GENERAL CHARACTERISTICS OF THE COURSE OF STUDY

2.1. Language of instruction	English
2.2. Prerequisites	Programming fundamentals

3. DETAILED CHARACTERISTICS OF THE COURSE OF STUDY

3.1. Form of classes	lectures, laboratories	
3.2. Place of classes	Courses in the UJK teaching rooms of the Faculty of Exact and Natural Sciences	
3.3. Form of assessment	credit with grade (lectures, laboratories, project)	
3.4. Teaching methods	lectures– informative lectures laboratories – laboratory method (practical classes using Python development tools)	
3.5. Bibliography	Required reading	1. M. Lutz, Learning Python: Powerful Object-Oriented Programming, 5th Edition, O'Reilly Media 2013, url: http://learning-python.com/about-lp5e.html 2. Real Python Website, url: https://realpython.com
	Further reading	3. B. Slatkin, Effective Python: 59 Specific Ways to Write Better Python. Addison-Wesley Professional 2015 4. L. Ramalho, Fluent Python: Clear, Concise, and Effective Programming, O'Reilly Media 2015

4. OBJECTIVES, SYLLABUS CONTENT AND INTENDED TEACHING OUTCOMES

4.1. Course objectives (including form of classes)
<p>Knowledge (lectures and laboratories) C1. To give the students a knowledge of the object oriented paradigm. C2. Presentation of Python language syntax in the field of object oriented programming.</p> <p>Abilities (laboratories) C3. Developing skills to implement software based on object oriented paradigm.</p> <p>Social competence (laboratories) C4. Developing competence to provide expert knowledge related to the object oriented programming.</p>

4.2. Detailed syllabus (including form of classes)

Lectures and laboratories:

Fundamentals of the object oriented paradigm: inherit, customize and extend, classes and instances, reusing code. Coding classes: attributes and methods definition, attribute access control, customizing constructors. Class objects: making instances, method calls. Inheritance: inheritance and composition, subclasses, coding class tree, polymorphism, operator overloading, namespaces, multiple inheritance. Advanced class topics: extending build-in types, static and class methods, decorators and metaclasses. Exceptions: coding, raising and catching, user-defined exceptions, try/except/else statement, coding termination action with try/finally statement, build-in exception classes. Iterators

Project:

Students cooperates in groups to design and implement low complexity software based on the object oriented paradigm.

4.3. Education outcomes in the discipline

Code	A student, who passed the course	Relation to teaching outcomes
within the scope of KNOWLEDGE:		
W01	has knowledge of the object oriented paradigm, understands class relations;	ID1A_W07
W02	defines class tree and object communication methods	ID1A_W07
within the scope of ABILITIES:		
U01	performs basic object-oriented analysis (OOA) and object oriented design (OOD) for specified task	ID1A_U07 ID1A_U08 ID1A_U13
U02	has skill to implement, debug and test object oriented software in Python	ID1A_U07 ID1A_U08 ID1A_U13
within the scope of SOCIAL COMPETENCE:		
K01	has competence to provide expert knowledge related to the object oriented programming	ID1A_K03 ID1A_K04

4.4. Methods of assessment of the intended teaching outcomes

Teaching outcomes (code)	Method of assessment (+/-)																				
	Oral exam			Project			Self-study			Group work											
	Form of classes			Form of classes			Form of classes			Form of classes			Form of classes			Form of classes					
	L	C	P	L	C	P	L	C	P	L	C	P	L	C	P	L	C	P	L	C	P
W01	+																				
W02	+																				
U01						+			+			+									
U02						+			+			+									
K01	+								+			+									

4.5. Criteria of assessment of the intended teaching outcomes		
Form of classes	Grade	Criterion of assessment
lecture (L)	3	at least 50% and not more than 60% of the total number of available points
	3,5	more than 60% and not more than 70% of the total number of available points
	4	more than 70% and not more than 80% of the total number of available points
	4,5	more than 80% and not more than 90% of the total number of available points
	5	more than 90% of the total number of available points
classes (C)	3	at least 50% and not more than 60% of the total number of available points
	3,5	more than 60% and not more than 70% of the total number of available points
	4	more than 70% and not more than 80% of the total number of available points
	4,5	more than 80% and not more than 90% of the total number of available points
	5	more than 90% of the total number of available points

5. BALANCE OF ECTS CREDITS – STUDENT’S WORK INPUT

Category	Student's workload	
	Full-time studies	Extramural studies
<i>NUMBER OF HOURS WITH THE DIRECT PARTICIPATION OF THE TEACHER /CONTACT HOURS/</i>		
<i>Participation in lectures</i>	30	
<i>Participation in laboratories</i>	30	
<i>Preparation for the test</i>		
<i>Others</i>		
<i>INDEPENDENT WORK OF THE STUDENT/NON-CONTACT HOURS/</i>		
<i>Preparation for the lecture</i>	5	
<i>Preparation for the laboratories</i>	25	
<i>Preparation for the test</i>	10	
<i>Gathering materials for the project</i>		
<i>Preparation of multimedia presentation</i>		
<i>Others*</i>		
TOTAL NUMBER OF HOURS	100	
ECTS credits for the course of study	4	

Accepted for execution (date and signatures of the teachers running the course in the given academic year)

.....