

OPIS PRZEDMIOTU/MODUŁU KSZTAŁCENIA (SYLABUS)

1.	Nazwa przedmiotu/modułu w języku polskim Języki skryptowe - Python	
2.	Nazwa przedmiotu/modułu w języku angielskim Scripting languages - Python	
3.	Jednostka prowadząca przedmiot Wydział Fizyki i Astronomii	
4.	Kod przedmiotu/modułu	
5.	Rodzaj przedmiotu/modułu (<i>obowiązkowy lub fakultatywny</i>) obowiązkowy	
6.	Kierunek studiów Informatyka stosowana i systemy pomiarowe	
7.	Poziom studiów (<i>I lub II stopień lub jednolite studia magisterskie</i>) I stopień	
1.	Rok studiów (<i>jeśli obowiązuje</i>) 1	
8.	Semestr (<i>zimowy lub letni</i>) zimowy	
9.	Forma zajęć i liczba godzin Wykład – 30 godz. Laboratorium komputerowe – 30 godz.	
10.	Imię, nazwisko, tytuł/stopień naukowy osoby prowadzącej zajęcia dr Janusz Szwabiński	
11.	Wymagania wstępne w zakresie wiedzy, umiejętności i kompetencji społecznych dla przedmiotu/modułu oraz zrealizowanych przedmiotów	
12.	Cele przedmiotu Celem kursu jest wprowadzenie do programowania z naciskiem na tworzenie przejrzystego, w miarę wydajnego i odpornego na błędy kodu. W ramach kursu student zapozna się z dużym podzbiorem języka skryptowego Python. Omówione zostaną techniki odgórnego projektowania aplikacji, analizy nieformalnej oraz techniki testowania i debugowania kodu.	
13.	Zna zasadę działania języków skryptowych	Symbole kierunkowych

	<p>(interpretowanych).</p> <p>Zna obszary zastosowań języków skryptowych, w tym Pythona.</p> <p>Zna środowiska programistyczne dedykowane Pythonowi.</p> <p>Zna podstawowe elementy języka Python.</p> <p>Zna popularne techniki programowania w Pythonie.</p> <p>Zna cykl życia programu komputerowego.</p> <p>Rozumie zasady prawnej ochrony dóbr koncepcyjnych i odpowiedzialności za jej naruszenie.</p> <p>Zna ograniczenia języków skryptowych, w tym Pythona.</p> <p>Potrafi tworzyć proste programy w Pythonie.</p> <p>Potrafi operować na złożonych strukturach danych.</p> <p>Potrafi korzystać z biblioteki standardowej oraz modułów zewnętrznych.</p> <p>Potrafi rozwiązywać proste praktyczne zadania przy pomocy programów napisanych w Pythonie.</p> <p>Potrafi tworzyć proste interfejsy graficzne.</p> <p>Potrafi skutecznie debugować własne programy.</p> <p>Potrafi optymalizować własne programy.</p> <p>Jest świadomy celowości ustawicznego kształcenia w obszarze IT.</p> <p>Jest świadomy korzyści płynących z zastosowania języków skryptowych do rozwiązywania niektórych problemów informatycznych.</p> <p>Ma świadomość korzyści wynikających z udostępniania wyników własnej pracy oraz korzystania z wyników pracy innych osób zgodnie z koncepcjami otwartego oprogramowania.</p>	<p>efektów kształcenia:</p> <p>K_W04, K_W10, K_U02, K_U06, K_U11, K_U17, K_K03, K_K05</p>
14.	<p>Treści programowe</p> <ul style="list-style-type: none"> • Środowiska do programowania w Pythonie. • Podstawowe typy danych, literały, operatory i wyrażenia. • Wyrażenia warunkowe i pętle. • Funkcje, skrypty i moduły. • Projektowanie odgórne, testowanie i debugowanie. • Listy, krotki, zbiory i słowniki. • Programowanie obiektowe: klasy i obiekty. • Grafika w Tkinter. • Operacje na plikach. • Analiza wydajności programów, profilowanie. 	
15.	<p>Zalecana literatura (<i>podręczniki</i>)</p> <ul style="list-style-type: none"> • A. B. Downey, "Think Python. How to Think Like a Computer Scientist", (http://www.greenteapress.com/thinkpython/) • M. Pilgrim, "Dive into Python", (http://www.diveintopython.net/) 	

16.	Forma zaliczenia poszczególnych komponentów przedmiotu/modułu, sposób sprawdzenia osiągnięcia zamierzonych efektów kształcenia: wykład: egzamin (praktyczny, przy komputerach) seminarium: laboratorium: listy zadań konwersatorium: inne:	
17.	Język wykładowy polski	
18.	Obciążenie pracą studenta	
	Forma aktywności studenta	Średnia liczba godzin na zrealizowanie aktywności
	Godziny zajęć (wg planu studiów) z nauczycielem: - wykład: - ćwiczenia: - laboratorium: - inne:	- 30 - 0 - 30 - 0
	Praca własna studenta np.: - przygotowanie do zajęć: - opracowanie wyników: - czytanie wskazanej literatury: - napisanie raportu z zajęć: - przygotowanie do egzaminu:	- 30 - 10 - 30 - 0 - 20
	Suma godzin	150
	Liczba punktów ECTS	5

*objaśnienie symboli:

K (przed podkreśleniem) - kierunkowe efekty kształcenia

W - kategoria wiedzy

U - kategoria umiejętności

K (po podkreśleniu) - kategoria kompetencji społecznych

01, 02, 03 i kolejne - numer efektu kształcenia

COURSE/MODULE DESCRIPTION (SYLLABUS)

1.	Course/module	
2.	University department	
3.	Course/module code	
4.	Course/module type – mandatory (compulsory) or elective (optional)	
5.	University subject (programme/major)	
6.	Degree: (<i>master, bachelor</i>)	
7.	Year	
8.	Semester (<i>autumn, spring</i>)	
9.	Form of tuition and number of hours	
10.	Name, Surname, academic title	
11.	Initial requirements (knowledge, skills, social competences) regarding the course/module and its completion	
12.	Objectives	
13.	Learning outcomes	Outcome symbols, e.g.: <i>K_W01*, K_U05, K_K03</i>
14.	Content	
15.	Recommended literature	
16.	Ways of earning credits for the completion of a course /particular component, methods of assessing academic progress: lecture: class: laboratory: seminar:	

	other:	
17.	Language of instruction	
18.	Student's workload	
	Activity	Average number of hours for the activity
	Hours of instruction (as stipulated in study programme) : - lecture: - classes: - laboratory: - other:	
	student's own work, e.g.: - preparation before class (lecture, etc.) - research outcomes: - reading set literature: - writing course report: - preparing for exam:	
	Hours	
	Number of ECTS	

* Key to symbols:

K (before underscore) - learning outcomes for the programme

W - knowledge

U - skills

K (after underscore) - social competences

01, 02, 03 and subsequent - consecutive number of learning outcome