

OPIS PRZEDMIOTU/MODUŁU KSZTAŁCENIA (SYLABUS)

1.	Nazwa przedmiotu/modułu w języku polskim oraz angielskim Języki skryptowe – Python / Scripting languages - Python
2.	Dyscyplina Informatyka techniczna i telekomunikacja - 5
3.	Język wykładowy Polski
4.	Jednostka prowadząca przedmiot Wydział Fizyki i Astronomii
5.	Kod przedmiotu/modułu 24-FZ-ISSP-S1-JsP
6.	Rodzaj przedmiotu/modułu (<i>obowiązkowy lub do wyboru</i>) Obowiązkowy
7.	Kierunek studiów (specjalność/specjalizacja) Informatyka stosowana i systemy pomiarowe
8.	Poziom studiów (<i>I lub II stopień, jednolite studia magisterskie, studia doktoranckie</i>) I stopień
9.	Rok studiów (<i>jeśli obowiązuje</i>) 1
10.	Semestr (<i>zimowy lub letni</i>) Zimowy
11.	Forma zajęć i liczba godzin Metody uczenia się Wykład – 15 godz. Laboratorium komputerowe – 45 godz.
12.	Imię, nazwisko, tytuł/stopień naukowy osoby prowadzącej zajęcia dr Tomasz Golan
13.	Wymagania wstępne w zakresie wiedzy, umiejętności i kompetencji społecznych dla przedmiotu/modułu brak
14.	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.

15.	<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 	
16.	<p>Zakładane efekty uczenia się</p> <p>Zna zasadę działania języków skryptowych (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. 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>Symbole odpowiednich kierunkowych efektów uczenia się:</p> <p>I1_W04, I1_W10, I1_U02, I1_U06, I1_U11, I1_U17, I1_K03, I1_K05</p>
17.	<p>Literatura obowiązkowa i zalecana (<i>źródła, opracowania, podręczniki itp.</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/) • Notatki z wykładu: https://tomaszolan.github.io/wfia_python/ 	

18.	Metody weryfikacji zakładanych efektów uczenia się: <ul style="list-style-type: none"> • Wykład <ul style="list-style-type: none"> ○ Egzamin praktyczny • Laboratorium <ul style="list-style-type: none"> ○ Listy zadań ○ Kartkówki ○ Kolokwia 	
19.	Warunki i forma zaliczenia poszczególnych komponentów przedmiotu/modułu: np. <ul style="list-style-type: none"> • ciągła kontrola obecności i kontroli postępów w zakresie tematyki zajęć <ul style="list-style-type: none"> ○ cotygodniowe listy zadań ○ cotygodniowe kartkówki ○ Kolokwia • Egzamin praktyczny 	
20.	20. Nakład pracy studenta/doktoranta	
	forma działań studenta/doktoranta	liczba godzin na realizację działań
	Zajęcia (wg planu studiów) z prowadzącym: - wykład: - konwersatorium: - laboratorium: - inne:	15 0 45 0
	Praca własna studenta, doktoranta (w tym udział w pracach grupowych) np.: - przygotowanie do zajęć: - opracowanie wyników: - czytanie wskazanej literatury: -przygotowanie prac/wystąpień/projektów: - napisanie raportu z zajęć: - przygotowanie do sprawdzianów i egzaminu:	30 10 30 0 0 20
	Łączna liczba godzin	150
	Liczba punktów ECTS	5