

KARTA PRZEDMIOTU

Cykl kształcenia od roku akademickiego: 2023/2024

I. Dane podstawowe

Nazwa przedmiotu	Programowanie w języku Python
Nazwa przedmiotu w języku angielskim	Python language programming
Kierunek studiów	Informatyka
Poziom studiów (I, II, jednolite magisterskie)	I
Forma studiów (stacjonarne, niestacjonarne)	stacjonarne
Dyscyplina	Informatyka
Język wykładowy	polski

Koordinator przedmiotu	Dr Waldemar Suszyński
------------------------	-----------------------

Forma zajęć (<i>katalog zamknięty ze słownika</i>)	Liczba godzin	semestr	Punkty ECTS
wykład	15	VI	3
konwersatorium			
ćwiczenia			
laboratorium	15	VI	
warsztaty			
seminarium			
proseminarium			
lektorat			
praktyki			
zajęcia terenowe			
pracownia dyplomowa			
translatorium			
wizyta studyjna			

Wymagania wstępne	Podstawy algorytmiki i programowania Programowanie obiektowe
-------------------	---

II. Cele kształcenia dla przedmiotu

C1. Zapoznanie z językiem programowania Python
C2. Przedstawienie technik programowania w języku Python

III. Efekty uczenia się dla przedmiotu wraz z odniesieniem do efektów kierunkowych

Symbol	Opis efektu przedmiotowego	Odniesienie do efektu kierunkowego
WIEDZA		
W_01	Student zna składnię i semantykę języka Python.	K_W01, K_W06
W_02	Student zna techniki programowania w języku Python.	K_W01, K_W06
UMIEJĘTNOŚCI		
U_01	Student potrafi przeanalizować skrypty napisane w języku	K_U04

	Python.	
U_02	Student potrafi zaprojektować skrypty napisane w języku Python.	K_U08, K_U11, K_U17
U_03	Student potrafi zastosować techniki programowania obiektowego w języku Python.	K_U08, K_U11, K_U17
KOMPETENCJE SPOŁECZNE		
K_01	Student jest świadom posiadanej wiedzy i umiejętności. Rozumie potrzebę uczenia się przez całe życie.	K_K01, K_K02, K_K05, K_K06
K_02	Student potrafi tworzyć efektywne projekty z wykorzystaniem języka Python.	K_K01, K_K02, K_K05, K_K06

IV. Opis przedmiotu / treści programowe

W ramach tego kursu studenci zdobywają umiejętności projektowania, rozwijania i ulepszania programów komputerowych przy użyciu języka Python. Zajęcia prowadzone są na platformie *skillsforall.com* i przygotowują do egzaminów certyfikacyjnych: *PCEP: Certified Entry-Level Python Programmer* i *PCAP - Certified Associate in Python Programming*. Dodatkowo studenci po zaliczeniu wymaganych testów i wykonaniu projektów mogą otrzymać cyfrowe potwierdzenie zdobytych umiejętności (Digital Badge)

Treści programowe:

1. Podział języków programowania, wersje języka Python
2. Typy danych, zmienne, operatory i podstawowe operacje we/wy
3. Wartości logiczne, wykonanie warunkowe, pętle, listy i przetwarzanie list. Operacje logiczne i bitowe.
4. Funkcje, krotki, słowniki, wyjątki i przetwarzanie danych
5. Moduły, pakiety i PIP
6. Znaki, ciągi znaków i wyjątki
7. Programowanie obiektowe
8. Praca z plikami i narzędziami

V. Metody realizacji i weryfikacji efektów uczenia się

Symbol efektu	Metody dydaktyczne (lista wyboru)	Metody weryfikacji (lista wyboru)	Sposoby dokumentacji (lista wyboru)
WIEDZA			
W_01	Wykład konwencjonalny	Egzamin	Protokół
W_02	Wykład konwencjonalny	Egzamin	Protokół
UMIEJĘTNOŚCI			
U_01	Ćwiczenia laboratoryjne design thinking	Kolokwium	Protokół
U_02	Ćwiczenia laboratoryjne design thinking	Kolokwium	Protokół
U_03	Ćwiczenia laboratoryjne design thinking	Kolokwium	Protokół
KOMPETENCJE SPOŁECZNE			
K_01	Ćwiczenia laboratoryjne design thinking	Kolokwium	Protokół
K_02	Ćwiczenia laboratoryjne design thinking	Kolokwium	Protokół

VI. Kryteria oceny, wagi...

Wykład (egzamin): zaliczenie testów finałowych na platformie *skillsforall.com*

Laboratorium (zaliczenie na ocenę): zaliczenie testów dla poszczególnych modułów, wykonanie projektów

W obu przypadkach:

91% - 100% bardzo dobry,

81% - 90% dobry z plusem,

71% - 80% dobry,

61% - 70% dostateczny z plusem,

50% - 60% dostateczny,

poniżej 50% niedostateczny.

Szczegółowe zasady oceniania są podawane studentom z każdą edycją przedmiotu.

VI. Obciążenie pracą studenta

Forma aktywności studenta	Liczba godzin
Liczba godzin kontaktowych z nauczycielem	Wykład 15 Laboratorium 15 Konsultacje 15
Liczba godzin indywidualnej pracy studenta	30

VII. Literatura

Literatura podstawowa
<ol style="list-style-type: none"> 1. Materiały dydaktyczne na stronie <i>skillsforall.com</i>, <i>netacad.com</i> 2. D. Kopec, „Klasyczne problemy informatyki w Pythonie”, PWN, Warszawa, 2020. 2. E. Matthes, „Python: instrukcje dla programisty”, Helion, Gliwice, 2020. 3. https://www.python.org/ 4. https://pl.wikibooks.org/wiki/Zanurkuj_w_Pythonie (ebook)
Literatura uzupełniająca
<ol style="list-style-type: none"> 1. M. Jaworski, T. Ziadé, „Profesjonalne programowanie w Pythonie : poziom ekspert”, Helion, Gliwice, 2017. 2. H. M. Deitel, P. J. Deitel, J. P. Liperi, B. A. Wiedermann, „Python. How to programm.”, Prentice Hall, New Jersey, 2002.