

Sylabus przedmiotu / modułu kształcenia		
Nazwa przedmiotu/modułu kształcenia:		Wybrane zastosowania języka Python
Nazwa w języku angielskim:		Selected applications of the Python language
Język wykładowy:	polski	
Kierunek studiów, dla którego przedmiot jest oferowany:		informatyka
Jednostka realizująca:		Wydział Nauk Ścisłych i Przyrodniczych
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):		Fakultatywny
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):		Pierwszego stopnia
Rok studiów:	drugi	
Semestr:	trzeci	
Liczba punktów ECTS:	3	
Imię i nazwisko koordynatora przedmiotu:		Dr Piotr Świtalski
Imię i nazwisko prowadzących zajęcia:		Dr Piotr Świtalski
Założenia i cele przedmiotu:		Celem kursu jest przekazanie studentom wiedzy na temat wybranych zastosowań i bibliotek języka Python oraz zdobycie praktycznych umiejętności ich wykorzystania w implementowanych systemach
Symbol efektu	Efekt uczenia się: WIEDZA	Symbol efektu kierunkowego
W_01	Zna i rozumie podstawowe konstrukcje i typy danych wykorzystywane w programowaniu skrypcowym, proceduralnym, obiektowym i funkcyjnym wspierane przez język Python.	K_W06
W_02	Zna i rozumie koncepcję lambda-wyrażeń, częściowej aplikacji argumentów funkcji, funkcji wyższego rzędu oraz standardowe funkcje wyższego rzędu wspierane przez język Python, m.in. map, filter, reduce, zip.	K_W06
W_03	Zna wybrane biblioteki języka Python, w szczególności te związane z obliczeniami naukowymi i analizą danych, m.in. NumPy, SciPy, Matplotlib	K_W06
Symbol efektu	Efekt uczenia się: UMIEJĘTNOŚCI	Symbol efektu kierunkowego
U_01	Potrafi sprawnie korzystać ze środowiska PyCharm (lub alternatywnego) w zakresie tworzenia aplikacji w języku Python. Potrafi posługiwać się notatnikami Jupyter w wybranym środowisku chmurowym (np. Google Colab)	K_U11

U_02	Umie dobrać i zastosować odpowiednie konstrukcje języka Python do praktycznego rozwiązania problemu. Potrafi zaprojektować i zaimplementować program w języku Python wg zadanych wymagań	K_U22
U_03	Potrafi przeanalizować kod źródłowy napisany w języku Python, a także znaleźć i naprawić typowe błędy	K_U22
Symbol efektu	Efekt uczenia się: KOMPETENCJE SPOŁECZNE	Symbol efektu kierunkowego
K_01	jest gotów do krytycznej oceny posiadanej wiedzy oraz jest gotów do uznawania znaczenia wiedzy w rozwiązywaniu problemów poznawczych i praktycznych z zakresu informatyki	K_K01
Forma i typy zajęć:	Studia stacjonarne: wykłady (21 godz.), ćwiczenia laboratoryjne (24 godz.) Studia niestacjonarne: wykłady (15 godz.), ćwiczenia laboratoryjne (15 godz.)	
Wymagania wstępne i dodatkowe:		
<p>Warunkiem uczestnictwa w zajęciach jest wcześniejsze uzyskanie zaliczenia z następujących przedmiotów:</p> <ul style="list-style-type: none"> • Matematyka dyskretna • Analiza matematyczna • Podstawy programowania <p>lub znajomość literatury obowiązującej w tym przedmiocie. Student musi mieć opanowane podstawy logiki matematycznej, kombinatoryki i statystyki.</p> <p>Ponadto wymagana jest znajomość podstaw strukturalnego lub obiektowego języka programowania, a w tym umiejętność definiowania typów, posługiwanie się instrukcjami iteracyjnymi, podprogramami i rekurencją.</p>		
Treści modułu kształcenia:		
<ol style="list-style-type: none"> 1. Wprowadzenie do języka Python. Rys historyczny, wersje, charakterystyka. Python 2 i Python 3. Zastosowania. Mocne i słabe strony języka Python. 2. Środowisko programistyczne Pycharm, podstawowa składnia i konstrukcje. Funkcje. Skrypty. Zapoznanie z wybranymi narzędziami, np. anaconda, Jupyter Notebook, Google Colab. 3. Proste i złożone typy danych. Listy, krotki, zbiory, słowniki. Wyrażenia logiczne. Iteratory, generatory. 4. Podstawowe koncepcje i techniki przetwarzania funkcyjnego. Wyrażenia lambda, częściowa aplikacja argumentów funkcji, funkcje wyższego rzędu. 5. Podstawy programowania obiektowego w języku Python. Dziedziczenie, metody specjalne, dekoratory, wyjątki. 6. Aplikacje webowe w języku Python: biblioteki flask i Django. 7. Wstęp do obliczeń naukowych i symbolicznych, analizy danych i uczenia maszynowego. Biblioteki: NumPy, Matplotlib, pandas, SciPy, z3py, scikit-learn 		
Literatura podstawowa:		
<ol style="list-style-type: none"> 1. R. Johansson, Matematyczny Python. Obliczenia naukowe i analiza danych z użyciem NumPy, SciPy i Matplotlib. Helion 2021. 2. M. Gorelick, I. Ozsvald. Wysoko wydajny Python: efektywne programowanie w praktyce. Helion 2021. 3. E. Matthes. Python. Instrukcje dla programisty. Helion 2016. 		
4. Literatura dodatkowa:		
<ol style="list-style-type: none"> 5. J. Grus. Data science od podstaw : analiza danych w Pythonie. Helion 2018. 6. S. Raschka, V. Mirjalili. Python. Uczenie maszynowe. Wydanie 2. Helion 2019. 		

7. Peter Farrell. Matematyczne przygody z Pythonem: ilustrowany podręcznik do nauki matematyki przez programowanie. Wydanie 1. Wydawnictwo Naukowe PWN 2019.
8. A. Niewiadomski, P. Switalski, T. Sidoruk, W. Penczek. SMT-Solvers in Action: Encoding and Solving Selected Problems in NP and EXPTIME. Sci. Ann. Comp. Sci. 28(2): 269-288, 2018, doi:10.7561/SACS.2018.2.269

Planowane formy/działania/metody dydaktyczne:

Wykład tradycyjny wspomagany technikami multimedialnymi. Zajęcia laboratoryjne – zajęcia praktyczne z wykorzystaniem języka Python i wybranego środowiska programistycznego (PyCharm lub alternatywne). Praca z notatnikami Jupyter w środowisku Google Colab, lub alternatywnym (np. Azure)

Sposoby weryfikacji efektów uczenia się osiągniętych przez studenta:

Efekty W_01 – W_03 będą sprawdzane na egzaminie pisemnym. Na egzaminie pisemnym pytania będą dotyczyły wybranych zagadnień teoretycznych oraz praktycznych związanych z zastosowaniami języka Python. Przykładowe pytania i zadania:

- Wymień wbudowane w język Python typy kolekcji i scharakteryzuj ich własności.
- Co to jest funkcja wyższego rzędu? Podaj definicję i przykład w języku Python.
- Na czym polega dziedziczenie? Podaj definicję i przykład w języku Python.
- Porównaj wbudowane w Python listy z typem numpy.ndarray. Podaj przykłady obydwu struktur.

Efekty U_01, U_02 i U_03 sprawdzane będą sukcesywnie i oceniane po każdym laboratorium, również poprzez sprawdzanie prac domowych i kolokwiów. Przykładowe zadania:

- Zdefiniuj funkcję w języku Python, która przyjmuje jako argumenty współczynniki równania kwadratowego (w postaci ogólnej) i wypisuje na ekran jego rozwiązania, lub komunikat o ich braku
- Zdefiniuj klasę Ułamek i zaimplementuj operatory arytmetyczne: dodawanie, odejmowanie, mnożenie i dzielenie ułamków zwykłych jako odpowiednie metody specjalne

Efekt K_01 będzie weryfikowany, w oparciu o posiadaną wiedzę i umiejętności w czasie zajęć laboratoryjnych, a także będzie sprawdzany na egzaminie.

Forma i warunki zaliczenia:

Moduł kończy się egzaminem. Do egzaminu mogą przystąpić osoby, które uzyskały zaliczenie laboratorium. Na zaliczenie laboratorium składają się oceny cząstkowe uzyskane na regularnych zajęciach z nauczycielem akademickim. Na tej formie zajęć student może maksymalnie uzyskać 60 pkt. Zajęcia laboratoryjne będą zaliczone w wypadku uzyskania powyżej 30 punktów.

Poprawa laboratorium: najpóźniej w ciągu 3 tygodni od danych zajęć. Poprawa nie jest możliwa po zakończeniu semestru.

Egzamin jest egzaminem pisemnym. Można na nim uzyskać do 40 pkt. Egzamin będzie zaliczony w przypadku uzyskania powyżej 20 pkt.

Ocena końcowa z przedmiotu, w zależności od sumy uzyskanych punktów (maksymalnie 100 pkt) jest następująca (w nawiasach ocena wg skali ECTS):

- 0 – 50 pkt: niedostateczna (F),
- 51 – 60 pkt: dostateczna (E),
- 61 – 70 pkt: dostateczna plus (D),
- 71 – 80 pkt: dobra (C),
- 81 – 90 pkt: dobra plus (B),
- 91 – 100 pkt: bardzo dobra (A).

Bilans punktów ECTS:

Studia stacjonarne

Aktywność

Obciążenie studenta

Udział w wykładach	21 godzin
Udział w ćwiczeniach laboratoryjnych	24 godziny
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	12 godzin
Udział w konsultacjach godz. z przedmiotu	2 godziny
Studia indywidualne z tematów realizowanych na wykładach	8 godzin
Przygotowanie się do egzaminu i obecność na egzaminie	10 godzin
Sumaryczne obciążenie pracą studenta	77 godzin
Punkty ECTS za przedmiot	3
Studia niestacjonarne	
Aktywność	Obciążenie studenta
Udział w wykładach	15 godzin
Udział w ćwiczeniach laboratoryjnych	15 godzin
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	15 godzin
Udział w konsultacjach godz. z przedmiotu	2 godziny
Studia indywidualne z tematów realizowanych na wykładach	15 godzin
Przygotowanie się do egzaminu i obecność na egzaminie	15 godzin
Sumaryczne obciążenie pracą studenta	77 godzin
Punkty ECTS za przedmiot	4