


KAPITAŁ LUDZKI
 NARODOWA STRATEGIA SPÓJNOŚCI

 Projekt współfinansowany przez
 Unię Europejską w ramach
 Europejskiego Funduszu
 Społecznego

UNIA EUROPEJSKA
 EUROPEJSKI
 FUNDUSZ SPOŁECZNY


Nazwa przedmiotu		Kod ECTS													
Podstawy programowania		11.1.0719													
Nazwa jednostki prowadzącej przedmiot															
Instytut Matematyki															
Studia															
wydział	kierunek	poziom	pierwszego stopnia												
Wydział Matematyki, Fizyki i Informatyki	Matematyka	forma	stacjonarne												
		moduł	matematyka nauczycielska, matematyka ogólna												
		specjalnościowy	wszystkie												
specjalizacja															
Nazwisko osoby prowadzącej (osób prowadzących)															
dr Iwona Krzyżanowska															
Formy zajęć, sposób ich realizacji i przypisana im liczba godzin		Liczba punktów ECTS													
Formy zajęć		5													
Ćw. laboratoryjne		Udział w zajęciach dydaktycznych, objętych planem studiów: 60h													
Sposób realizacji zajęć		Udział w konsultacjach: 4h													
zajęcia w sali dydaktycznej		Praca własna studenta: 61h													
Liczba godzin		RAZEM: 125h													
Ćw. laboratoryjne: 60 godz.															
Termin realizacji przedmiotu															
2023/2024 letni															
Status przedmiotu		Język wykładowy													
obowiązkowy		polski													
Metody dydaktyczne		Forma i sposób zaliczenia oraz podstawowe kryteria oceny lub wymagania egzaminacyjne													
<ul style="list-style-type: none"> - Praca w grupach - Rozwiązywanie zadań programistycznych - ćwiczenia laboratoryjne 		Sposób zaliczenia													
		Zaliczenie na ocenę													
		Formy zaliczenia													
		<ul style="list-style-type: none"> - ustalenie oceny zaliczeniowej na podstawie ocen cząstkowych otrzymywanych w trakcie trwania semestru - Projekty programistyczne w grupach 2-3 osobowych. - kolokwium 													
		Podstawowe kryteria oceny													
		<table border="1"> <thead> <tr> <th>Sposób oceniania (składowe)</th> <th>Próg zaliczeniowy</th> <th>Składowa oceny końcowej</th> </tr> </thead> <tbody> <tr> <td>kolokwia</td> <td>0,00%</td> <td>66,(6)%</td> </tr> <tr> <td>projekt</td> <td>0,00%</td> <td>33,(3)%</td> </tr> <tr> <td>obserwacja postawy studenta</td> <td>100%</td> <td>0%</td> </tr> </tbody> </table>		Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa oceny końcowej	kolokwia	0,00%	66,(6)%	projekt	0,00%	33,(3)%	obserwacja postawy studenta	100%	0%
Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa oceny końcowej													
kolokwia	0,00%	66,(6)%													
projekt	0,00%	33,(3)%													
obserwacja postawy studenta	100%	0%													
Sposób weryfikacji założonych efektów uczenia się															

zakładany efekt kształcenia	Kolokwia	Projekt	Obserwacja postawy studenta
	Wiedza		
M_W10	+	+	
M_W12			+
	Umiejętności		
M_U09	+		
M_U10	+	+	
M_U11	+	+	
M_U12	+	+	
M_U13	+	+	+
	Kompetencje		
M_K01			+
M_K03		+	
M_K04			+
M_K09		+	+

Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi**A. Wymagania formalne**

Brak

B. Wymagania wstępne

Brak

Cele kształcenia

Nabycie przez studentów umiejętności algorytmizacji prostych problemów i implementacji ich w języku programowania wysokiego poziomu ogólnego przeznaczenia np. Python w oparciu o podstawowe elementy tego języka. Wykształcenie umiejętności kontroli i weryfikacji własnego i cudzego rozumowania i kodu źródłowego. Przygotowanie do pogłębionych poszukiwań intelektualnych, literaturowych i technicznych wobec problemów informatycznych, dla których proste narzędzia nie wystarczają.

Treści programowe

- Przygotowanie środowiska.
- Python interaktywnie, interface IDLE, podstawowe typy zmiennych (listy, słowniki, krotki, zbiory).
- Tworzenie prostych skryptów.
- Instrukcja warunkowa if.
- Pętle for, while.
- Liczby pseudolosowe.
- Praca z plikiem, zapis, odczyt i modyfikacja.
- Testowanie oraz obsługa błędów.
- Pisanie własnych funkcji.
- Graficzny interfejs użytkownika GUI, np. moduł tkinter, graphics.py.
- Wprowadzenie do klas.

Wykaz literatury**A. Literatura wymagana do ostatecznego zaliczenia zajęć (zdania egzaminu):****A.1. wykorzystywana podczas zajęć**

- <https://docs.python.org/3/>

A.2. studiowana samodzielnie przez studenta

- <https://docs.python.org/3/>
- Lutz M., Python. Wprowadzenie.

B. Literatura uzupełniająca

- Heinold B., A Practical Introduction to Python Programming
- McKinney W., Python w analizie danych. Przetwarzanie danych za pomocą pakietów Pandas i NumPy oraz środowiska IPython

Kierunkowe efekty uczenia się

M_W10
zna i rozumie podstawy technik obliczeniowych i programowania, wspomagających pracę matematyka i

Wiedza

Student zna i rozumie podstawowe typy zmiennych, wyrażenia, podstawowe instrukcje sterujące sposoby definiowania funkcji oraz podstawowe operacje wejścia-wyjścia poznanego języka, zna i rozumie ich ograniczenia (M_W10).

<p>rozumie ich ograniczenia M_W12 zna i rozumie zasady bezpieczeństwa i higieny pracy M_U09 potrafi zaplanować sposób rozwiązania określonego problemu oraz sporządzić poprawny zapis tego rozwiązania, podając ścisłe i precyzyjne uzasadnienia poprawności swoich rozumowań M_U10 potrafi wykorzystywać poznany pakiet oprogramowania lub poznany język programowania do rozwiązywania wybranych zagadnień z poznanych dziedzin, w szczególności z analizy matematycznej, algebry liniowej oraz statystyki M_U11 potrafi rozpoznać problemy, w tym zagadnienia praktyczne, które można rozwiązać algorytmicznie; potrafi dokonać specyfikacji takiego problemu M_U12 potrafi ułożyć i analizować algorytm zgodny ze specyfikacją i zapisać go w wybranym języku programowania M_U13 potrafi skompilować, uruchomić i testować napisany samodzielnie program komputerowy M_K01 jest gotów do uznania ograniczenia własnej wiedzy i jest gotów do dalszego kształcenia M_K03 jest gotów do pracy zespołowej; rozumie konieczność systematycznej pracy nad wszelkimi projektami, które mają długofalowy charakter M_K04 jest gotów do zrozumienia i docenienia znaczenia uczciwości intelektualnej w działaniach własnych i innych osób; etycznego postępowania M_K09 jest gotów do krytycznej oceny argumentów, znajdowania luk w rozumowaniach i konstruktywnej krytyki w stosunku do rozumowań innych osób</p>	<p>Student zna i rozumie zasady bezpieczeństwa i higieny pracy w laboratorium komputerowym (M_W12).</p> <p>Umiejętności</p> <p>Student potrafi skompilować, uruchomić i testować napisany samodzielnie program komputerowy. (M_U13) Student potrafi sformułować algorytm dla prostego problemu np. matematycznego, uzasadnić jego poprawność oraz potrafi zaimplementować ten algorytm w poznanym języku dobierając odpowiednie typy danych i instrukcje sterujące; potrafi zaprojektować strukturę programu wyodrębniając podprogramy (funkcje). (M_U09-M_U12)</p> <p>Kompetencje społeczne (postawy)</p> <p>Student zna ograniczenia własnej wiedzy i jest gotów do dalszego kształcenia. (M_K01) Student jest gotów do pracy zespołowej; rozumie konieczność systematycznej pracy nad wszelkimi projektami, które mają długofalowy charakter. (M_K03) Student jest gotów do zrozumienia i docenienia znaczenia uczciwości intelektualnej w działaniach własnych i innych osób; etycznego postępowania. (M_K04) Student jest gotów do krytycznej oceny argumentów, znajdowania luk w rozumowaniach i konstruktywnej krytyki w stosunku do rozumowań innych osób. (M_K09)</p>
<p>Kontakt</p> <p>iwona.krzyzanowska@ug.edu.pl</p>	