

Karta przedmiotu

Nazwa i kod przedmiotu	Wstęp do programowania (Ćw. laboratoryjne), PG_00153241						
Kierunek studiów	Modelowanie matematyczne i analiza danych (O)						
Data rozpoczęcia studiów	październik 2025 r.	Rok akademicki realizacji przedmiotu			2025/2026		
Poziom kształcenia	I stopnia - licencjackie	Grupa zajęć			Grupa zajęć obowiązkowych z zakresu kierunku studiów		
Forma studiów	stacjonarne	Sposób realizacji			na uczelni		
Rok studiów	1	Język wykładowy			polski		
Semestr studiów	1	Liczba punktów ECTS			5.0		
Profil kształcenia	ogólnoakademicki	Forma zaliczenia			zaliczenie		
Jednostka prowadząca	Rektor -> Wydział Matematyki, Fizyki i Informatyki -> Instytut Matematyki						
Imię i nazwisko wykładowcy (wykładowców)	Odpowiedzialny za przedmiot	dr Iwona Krzyżanowska					
	Prowadzący zajęcia z przedmiotu						
Formy zajęć	Forma zajęć	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium	RAZEM
	Liczba godzin zajęć	0.0	0.0	60.0	0.0	0.0	60
	W tym liczba godzin zajęć na odległość: 0.0						
Aktywność studenta i liczba godzin pracy	Aktywność studenta	Udział w zajęciach dydaktycznych, objętych planem studiów		Udział w konsultacjach		Praca własna studenta	RAZEM
	Liczba godzin pracy studenta	60		5.0		60.0	125
Cel przedmiotu	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.						

Efekty uczenia się przedmiotu	Efekt kierunkowy	Efekt z przedmiotu	Sposób weryfikacji i oceny efektu
	[MMiADL3_U11] potrafi ułożyć i analizować algorytm zgodny ze specyfikacją i zapisać go w wybranym języku programowania	Student potrafi sformułować algorytm dla prostego problemu np. matematycznego czy analizy danych oraz potrafi zaimplementować ten algorytm w języku Python dobierając odpowiednie typy danych i instrukcje sterujące. Student potrafi zaprojektować strukturę programu wyodrębniając podprogramy (funkcje).	[SU5] realizacja zadania problemowego
	[MMiADL3_W09] zna i rozumie podstawy technik obliczeniowych i programowania, wspomagających pracę matematyka i rozumie ich ograniczenia	Student zna i rozumie podstawowe typy zmiennych, wyrażenia, podstawowe instrukcje sterujące sposoby definiowania funkcji oraz podstawowe operacje wejścia-wyjścia języka Python. Student rozumie ich ograniczenia.	[SW5] realizacja zadania problemowego
	[MMiADL3_U12] potrafi skompilować, uruchomić i testować napisany samodzielnie program komputerowy	Student potrafi skompilować, uruchomić i testować napisany samodzielnie program komputerowy napisany z języku wysokiego rzędu w zakresie treści programowych.	[SU5] realizacja zadania problemowego [SU8] obserwacja samodzielnej lub zespołowej pracy studenta
	[MMiADL3_U13] potrafi wykorzystywać programy komputerowe w zakresie analizy danych	Student potrafi sformułować algorytm dla prostego problemu np. matematycznego czy analizy danych oraz potrafi zaimplementować ten algorytm w języku Python dobierając odpowiednie typy danych i instrukcje sterujące. Student potrafi zaprojektować strukturę programu wyodrębniając podprogramy (funkcje).	[SU5] realizacja zadania problemowego
	[MMiADL3_U09] 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	Student potrafi sformułować algorytm dla prostego problemu np. matematycznego czy analizy danych oraz potrafi zaimplementować ten algorytm w języku Python dobierając odpowiednie typy danych i instrukcje sterujące. Student potrafi zaprojektować strukturę programu wyodrębniając podprogramy (funkcje).	[SU5] realizacja zadania problemowego
	[MMiADL3_U10] potrafi rozpoznać problemy, w tym zagadnienia praktyczne, które można rozwiązać algorytmicznie; potrafi dokonać specyfikacji takiego problemu	Student potrafi sformułować algorytm dla prostego problemu np. matematycznego czy analizy danych oraz potrafi zaimplementować ten algorytm w języku Python dobierając odpowiednie typy danych i instrukcje sterujące. Student potrafi zaprojektować strukturę programu wyodrębniając podprogramy (funkcje).	[SU5] realizacja zadania problemowego
	[MMiADL3_K03] jest gotów do pracy zespołowej; rozumie konieczność systematycznej pracy nad wszelkimi projektami, które mają długofalowy charakter	Student jest gotów do współpracy w zespole oraz rozumie potrzebę systematycznego zaangażowania w realizację projektów związanych z treściami przedmiotu o charakterze długoterminowym.	[SK8] obserwacja samodzielnej lub zespołowej pracy studenta

Treści przedmiotu	1. Przygotowanie środowiska. 2. Podstawowe typy zmiennych (listy, słowniki, krotki, zbiory). 3. Tworzenie prostych skryptów. 4. Instrukcja warunkowa if. 5. Liczby pseudolosowe. 6. Pętle for, while. 7. Praca z plikiem, zapis, odczyt, modyfikacja. 8. Testowanie oraz obsługa błędów. 9. Pisanie własnych funkcji. 10. Graficzny interfejs użytkownika GUI, np. moduł tkinter, graphics.py 11. Wprowadzenie do klas.		
Wymagania wstępne i dodatkowe			
Sposoby i kryteria oceniania osiąganych efektów uczenia się	Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa oceny końcowej
	kolokwia i/lub projekt	51.0%	100.0%
	obserwacja postawy studenta	51.0%	0.0%
Zalecana lista lektur	Podstawowa lista lektur	1. https://docs.python.org/3/ 2. Lutz M., Python. Wprowadzenie.	
	Uzupełniająca lista lektur	- 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	
	Adresy eZasobów		
Przykładowe zagadnienia/ przykładowe pytania/ realizowane zadania	brak		
Praktyki zawodowe w ramach przedmiotu	Nie dotyczy		

Dokument wygenerowany elektronicznie. Nie wymaga pieczęci ani podpisu.