

Karta przedmiotu

Nazwa i kod przedmiotu	Wstęp do programowania, PG_00167091						
Kierunek studiów	Bezpieczeństwo jądrowe i ochrona radiologiczna (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 Grupa zajęć fakultatywnych		
Forma studiów	stacjonarne		Sposób realizacji		na uczelni		
Rok studiów	1		Język wykładowy		polski j. polski		
Semestr studiów	1		Liczba punktów ECTS		5.0		
Profil kształcenia	ogólnoakademicki		Forma zaliczenia		zaliczenie		
Jednostka prowadząca	Rektor -> Wydział Matematyki, Fizyki i Informatyki						
Imię i nazwisko wykładowcy (wykładowców)	Odpowiedzialny za przedmiot		dr Sławomir Werbowy				
	Prowadzący zajęcia z przedmiotu		dr Sławomir Werbowy				
Formy zajęć	Forma zajęć	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium	RAZEM
	Liczba godzin zajęć	15.0	0.0	60.0	0.0	0.0	75
	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	75		0.0		60.0	135
Cel przedmiotu	Wprowadzenie do programowania oraz elementów teorii algorytmów i struktur danych.						

Efekty uczenia się przedmiotu	Efekt kierunkowy	Efekt z przedmiotu	Sposób weryfikacji i oceny efektu
	[BJORL3_U04] potrafi posługiwać się aparatem matematycznym i informatycznym do analizy i rozwiązywania problemów z zakresu ochrony radiologicznej i bezpieczeństwa jądowego	Student potrafi: - precyzyjnie sformułować problem obliczeniowy do dalszej analizy; - zaimplementować sformułowany problem w wybranym języku programowania; - potrafi analizować i interpretować wyniki w celu pogłębienia zrozumienia badanego zagadnienia;	[SU5] implementation of a problem task [SU6] demonstration of practical skills [SU8] observation of student's independent or team work
	[BJORL3_W02] rozumie rolę eksperymentu fizycznego i chemicznego, matematycznych modeli teoretycznych przybliżających rzeczywistość, oraz symulacji komputerowych w metodologii badań naukowych; ma świadomość ograniczeń technologicznych, aparaturowych i metodologicznych w badaniach naukowych	Student: - ma wiedzę na temat programowania strukturalnego i obiektowego w wybranym języku wysokiego poziomu (Python); - zna typy danych, instrukcje sterujące oraz operatory arytmetyczne i logiczne w języku Python; - rozumie zagadnienia programowania obiektowego oraz potrafi je stosować w praktycznych zadaniach; - zna i potrafi wykorzystać pakiety obliczeniowe oraz narzędzia do obliczeń symbolicznych; - zna i potrafi korzystać z pakietów oprogramowania użytkowego do prezentacji wyników i analizy danych.	[SW5] implementation of a problem task
	[BJORL3_U02] posiada umiejętność wykonywania pomiarów podstawowych wielkości stosowanych w fizyce i chemii; potrafi opracować, opisać i przedstawić wyniki prostych eksperymentów i symulacji komputerowych; potrafi wykonywać analizy ilościowe oraz formułować na tej podstawie wnioski jakościowe; potrafi szacować niepewności pomiarowe	Student potrafi: - sformułować prosty algorytm numeryczny rozwiązujący zadany problem; - napisać i uruchomić w wybranym środowisku programistycznym program komputerowy w języku Python, przetwarzający dane liczbowe i tekstowe; - stosować podejście strukturalne i obiektowe w tworzeniu programu gotowego do testowania i dokumentowania; - napisać program odczytujący/zapisujący dane alfanumeryczne do/z pliku; - przetestować program, zidentyfikować i poprawić błędy oraz sporządzić dokumentację opisującą sposób działania programu.	[SU5] implementation of a problem task [SU6] demonstration of practical skills [SU8] observation of student's independent or team work
Treści przedmiotu	1. Wstęp do Pythona i środowiska Spyder. 2. Typy danych, operatory arytmetyczne i logiczne, instrukcje warunkowe i sterujące. 3. Złożone typy danych: listy, krotki, słowniki. 4. Programowanie proceduralne: funkcje, moduły, podstawowy pracy z pakietami. 5. Odczyt i zapis danych alfanumerycznych do i z pliku. 6. Elementy programowania obiektowego: klasy, obiekty, metody. 7. Podstawy pracy z pakietami obliczeniowymi i wizualizacji danych.		
Wymagania wstępne i dodatkowe			
Sposoby i kryteria oceniania osiągniętych efektów uczenia się	Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa oceny końcowej
	zadania domowe	0.0%	20.0%
	wykonywania zadań programistycznych w trakcie semestru	51.0%	80.0%
Zalecana lista lektur	Podstawowa lista lektur	- A.1. Wykorzystywana podczas zajęć: - Instrukcje i materiały udostępniane przez prowadzącego - A.2. Studiowana samodzielnie przez studenta: - M. Lutz, Python. Wprowadzenie. Helion 2022 - E. Matthes, Python. Instrukcje dla programisty. Helion 2019 - K. Dziedzic, Python w pigułce. Ringier Axel Springer 2021 - M. Kubiak, Python, zadania z Programowania. Helion 2021 - I. Ahmad, 40 algorytmów, które powinien znać każdy programista. Helion 2021	

	Uzupełniająca lista lektur	R. Johansson, Matematyczny Python. Obliczenia naukowe i analiza danych z użyciem NumPy, SciPy i Matplotlib. Helion 2019 2001
	Adresy eZasobów	
Przykładowe zagadnienia/ przykładowe pytania/ realizowane zadania	Napisz program w języku Python, który: 1. Wczyta dane pomiarowe z pliku CSV (np. czas, temperatura, napięcie). 2. Przeprowadzi prostą analizę: obliczy średnie wartości, odchylenia, minimalne i maksymalne wartości dla każdej kolumny. 3. Utworzy wykresy zależności między danymi (np. temperatura vs. napięcie).	
Praktyki zawodowe w ramach przedmiotu	Nie dotyczy	

Dokument wygenerowany elektronicznie. Nie wymaga pieczęci ani podpisu.