



## Karta przedmiotu Programowanie

### 1. Informacje podstawowe

<b>Kierunek studiów</b> inżynieria w medycynie	<b>Cykl kształcenia (nabór)</b> 2024/25	
<b>Specjalność</b> -	<b>Kod przedmiotu</b> 03IMEN.DI1C.2983.24	
<b>Jednostka zarządzająca kierunkiem studiów</b> Wydział Inżynierii Mechanicznej	<b>Języki wykładowe</b> polski	
<b>Poziom studiów</b> drugiego stopnia (mgr inż.)	<b>Obligatoryjność</b> Obowiązkowy	
<b>Profil studiów</b> Profil ogólnoakademicki	<b>Blok zajęciowy</b> Przedmioty kierunkowe	
<b>Forma studiów</b> studia niestacjonarne		
<b>Wymagania wstępne</b>	Podstawowa wiedza z zakresu programowania	
<b>Przedmioty wprowadzające</b>	Brak	
<b>Koordinator</b>	Sandra Śmigiel	
<b>Okres</b> Semestr 1	<b>Forma i godziny zajęć</b> <ul style="list-style-type: none"><li>Wykład: 18, Zaliczenie na ocenę</li><li>Ćwiczenia laboratoryjne: 18, Zaliczenie na ocenę</li><li>Ćwiczenia projektowe: 18, Zaliczenie na ocenę</li></ul>	<b>Liczba punktów ECTS</b> 6

### 2. Efekty uczenia się dla przedmiotu

Kod	Opis efektów uczenia się	Odniesienie do kierunkowych efektów uczenia się	Odniesienie do charakterystyk PRK
<b>Wiedza:</b>			

Kod	Opis efektów uczenia się	Odniesienie do kierunkowych efektów uczenia się	Odniesienie do charakterystyk PRK
W1	Zna i rozumie języki programowania i rozumie znaczenie oraz wykorzystanie ich w zastosowaniach medycznych.	IME_O2_K_W03	P7S_WG P7S_WG_inż
W2	Posiada wiedzę z zakresu wytwarzania oprogramowania na potrzeby urządzeń medycznych.	IME_O2_K_W06	P7S_WG P7S_WG_inż
W3	Zna i rozumie zasady implementacji oprogramowania, oraz rozwiązywania problemów inżynierskich w medycynie.	IME_O2_K_W07	P7S_WG P7S_WG_inż
W4	Zna i rozumie algorytmy mające zastosowanie w diagnostyce i terapii medycznej.	IME_O2_K_W10	P7S_WG P7S_WG_inż
<b>Umiejętności:</b>			
U1	Potrafi podnosić własne kompetencje zawodowe (w tym ma umiejętność samokształcenia się), w zakresie wytwarzania oprogramowania na potrzeby medycyny.	IME_O2_K_U02	P7S_UK P7S_UU
U2	Potrafi ocenić przydatność i wskazać możliwość wykorzystania nowych rozwiązań i osiągnięć w zakresie wytwarzania oprogramowania.	IME_O2_K_U06	P7S_UW P7S_UW_inż
<b>Kompetencje społeczne:</b>			
K1	Jest gotów do pogłębiania wiedzy technicznej, a także ma świadomość istotnej roli inżyniera medycznego obecnie oraz w przyszłości; ponadto rozumie techniczne i pozatechniczne aspekty i skutki działalności inżynierskiej.	IME_O2_K_K01	P7S_KK P7S_KO

### 3. Treści programowe

Lp.	Treści programowe	Formy zajęć	Efekty uczenia się dla przedmiotu
1.	<ul style="list-style-type: none"> <li>• Podstawowe typy danych w Pythonie (np. liczby, łańcuchy znaków, listy)</li> <li>• Operatory matematyczne i logiczne</li> <li>• Instrukcje warunkowe</li> <li>• Pętle (np. for, while)</li> <li>• Funkcje i struktury danych</li> <li>• Sortowanie i filtracja danych</li> <li>• Generatory</li> <li>• Programowanie obiektowe</li> <li>• Klasy i obiekty</li> <li>• Dziedziczenie i polimorfizm</li> <li>• Enkapsulacja i interfejsy</li> <li>• Przeciążanie metod i operatorów</li> <li>• Dekoratory</li> <li>• Wyjątki i obsługa błędów</li> <li>• Wstęp do programowania funkcyjnego, wyrażenia lambda</li> <li>• Programowanie sieciowe</li> <li>• Programowanie asynchroniczne</li> <li>• Moduły do przetwarzania danych</li> </ul>	Wykład	W1, W2, W3, W4, K1
2.	Ćwiczenia laboratoryjne wykonywane indywidualnie adekwatne do treści przedstawionych na wykładzie, wykonywane w języku Python.	Ćwiczenia laboratoryjne	W1, W4, U1, U2, K1

Lp.	Treści programowe	Formy zajęć	Efekty uczenia się dla przedmiotu
3.	Projekt indywidualny lub zespołowy polegający na napisaniu aplikacji realizującej zadanie wyznaczone przez prowadzącego.	Ćwiczenia projektowe	U1, U2, K1

#### 4. Metody prowadzenia zajęć, weryfikacji efektów uczenia się i warunki zaliczenia

Forma zajęć			
Wykład	<b>Metody prowadzenia zajęć:</b>		
	Wykład, Dyskusja		
	<b>Metody (sposoby) weryfikacji:</b>		<b>Udział:</b>
	Zaliczenie pisemne		70%
	Aktywność		30%
	<b>Warunki zaliczenia przedmiotu:</b>		
Uzyskanie minimum 51% punktów na zaliczeniu. Punkty za aktywność są doliczane do punktów z egzaminu.			
Ćwiczenia laboratoryjne	<b>Metody prowadzenia zajęć:</b>		
	Ćwiczenia laboratoryjne		
	<b>Metody (sposoby) weryfikacji:</b>		<b>Udział:</b>
	Sprawozdanie		100%
	<b>Warunki zaliczenia przedmiotu:</b>		
Oddanie prawidłowo wykonanych sprawozdań z przeprowadzonych ćwiczeń.			
Ćwiczenia projektowe	<b>Metody prowadzenia zajęć:</b>		
	Projekt		
	<b>Metody (sposoby) weryfikacji:</b>		<b>Udział:</b>
	Projekt		100%
	<b>Warunki zaliczenia przedmiotu:</b>		
Oddanie prawidłowo wykonanego projektu wraz z dokumentacją.			

Efekt uczenia się dla przedmiotu	<b>Metody (sposoby) weryfikacji</b>			
	Zaliczenie pisemne	Aktywność	Sprawozdanie	Projekt
W1	x	x	x	
W2	x	x		
W3	x	x		
W4	x	x	x	

U1			x	x
U2			x	x
K1	x	x	x	x

## 5. Literatura

### Literatura podstawowa

1. Dawson M., 2014, Python dla każdego : podstawy programowania, Helion, Gliwice.
2. Mark Summerfield, 2010, Python 3 : kompletne wprowadzenie do programowania, Helion, Gliwice
3. Mark Lutz, 2014, Python : leksykon kieszonkowy, Helion, Gliwice

### Literatura uzupełniająca

1. Coldwind G., 2016, Zrozumieć programowanie, Wydawnictwo Naukowe PWN, Warszawa
2. Wes McKinney, 2023, Python w analizie danych. Przetwarzanie danych za pomocą pakietów pandas i NumPy oraz środowiska Jupyter. Wydanie III, Helion, Gliwice
3. Mark Lutz, 2022, Python. Wprowadzenie. Wydanie V, Helion, Gliwice

## 6. Nakład pracy studenta - bilans godzin i punktów ECTS

Aktywność studenta		Obciążenie studenta Liczba godzin
Zajęcia prowadzone z bezpośrednim udziałem nauczyciela akademickiego lub innych osób prowadzących zajęcia	Wykład	18
	Ćwiczenia laboratoryjne	18
	Ćwiczenia projektowe	18
Praca własna studenta	Konsultacje	15
	Przygotowanie do zajęć	30
	Studiowanie literatury	10
	Przygotowanie do zaliczenia	16
	Przygotowanie sprawozdania	5
	Przygotowanie projektu	20
<b>Łączny nakład pracy studenta</b>		<b>150</b>
<b>Liczba punktów ECTS</b>		<b>6</b>

\* Godzina (dydaktyczna) oznacza 45 minut