



**POLITECHNIKA
BYDGOSKA**

Wydział Inżynierii Mechanicznej

Karta przedmiotu
Wybrane elementy z zakresu programowania

1. Informacje podstawowe

Kierunek studiów mechanika i budowa maszyn Specjalność: konstrukcja dronów Jednostka zarządzająca kierunkiem studiów Wydział Inżynierii Mechanicznej Poziom studiów drugiego stopnia (mgr inż.) Profil studiów Profil ogólnoakademicki Forma studiów studia niestacjonarne		Cykl kształcenia (nabór) 2024/25 Kod przedmiotu 03MBMKDN.DICD.3270.24 Języki wykładowe polski Obligatoryjność Obligatoryjny specjalnościowy Blok zajęciowy Przedmioty specjalnościowe	
Wymagania wstępne			
Przedmioty wprowadzające			
Koordinator		Krzysztof Nowicki	
Okres Semestr 3		Forma zaliczenia Zaliczenie na ocenę Forma prowadzenia i godziny zajęć Ćwiczenia laboratoryjne: 10	Liczba punktów ECTS 2.0
Okres Semestr 4		Forma zaliczenia Zaliczenie na ocenę Forma prowadzenia i godziny zajęć Ćwiczenia laboratoryjne: 10	Liczba punktów ECTS 1.0

2. Efekty uczenia się dla przedmiotu

Kod	Opis efektów uczenia się	Odniesienie do kierunkowych efektów uczenia się	Odniesienie do charakterystyk PRK
Wiedza:			
W1	Posiada rozszerzoną wiedzę z zakresu matematyki i fizyki przydatną w procesie programowania dronów.	MBM_O2_K_W01	P7S_WG P7S_WG_inż
Umiejętności:			
U1	Posiada umiejętności przygotowywania prostych planów badawczych i analizy wyników w zakresie testowania oprogramowania dronów. Ma umiejętność posługiwania się oprogramowaniem narzędziowym do programowania dronów.	MBM_O2_K_U04, MBM_O2_K_U05	P7S_UW, P7S_UW_inż, P7S_UW P7S_UW_inż
Kompetencje społeczne:			
K1	Potrafi odpowiednio określić priorytety służące realizacji określonego przez siebie lub innych zadania.	MBM_O2_K_K02	P7S_KO

3. Treści programowe

Lp.	Treści programowe	Formy zajęć	Efekty uczenia się dla przedmiotu
1.	Temat: Paradygmaty przetwarzania informacji. Cel: Przedstawienie szeregowego i równoległego paradygmatu przetwarzania informacji oraz ich związków z architekturą maszyn cyfrowych. Temat: Paradygmaty programowania. Cel: Przedstawienie najpopularniejszych paradygmatów programowania: imperatywnego, obiektowego, funkcyjnego oraz w logice. Temat: Semantyka zmiennych. Cel: Przedstawienie koncepcji zmiennej jako abstrakcji komórek pamięci. Zasady tworzenia zmiennych i związane z tym problemy implementacyjne. Temat: Typy danych. Cel: Przedstawienie koncepcji typów danych (zmiennych) jako narzędzia do wykrywania błędów oraz optymalizacji działania programów komputerowych wraz z omówieniem problemów implementacyjnych. Temat: Typy abstrakcyjne. Cel: Prezentacja koncepcji typów abstrakcyjnych jako metody tworzenia własnych typów zmiennych oraz sposobu modularyzacji programu. Temat: Programowania strukturalne – podprogramy. Cel: Podprogramy jako abstrakcja procesu i metoda na modularyzację programu. Temat: Programowanie obiektowe. Cel: Klasa i obiekt jako abstrakcja danych i metoda na modularyzację programu bliską ludzkiemu sposobowi opisu świata materialnego. Temat: Programowanie funkcyjne i w logice. Cel: Prezentacja alternatywnych paradygmatów programowania jako narzędzia do rozwiązywania problemów niszowych.	Ćwiczenia laboratoryjne	W1, K1

Lp.	Treści programowe	Formy zajęć	Efekty uczenia się dla przedmiotu
2.	Temat: Wprowadzenie do języka Python. Cel: Zapoznanie się praktyczne z podstawowymi cechami języka programowanie. Opanowanie zasad posługiwania się środowiskiem programistycznym. Temat: Wartości liczbowe i operacje na nich. Cel: Opanowanie zasad zapisu podstawowych liczbowych typów zmiennych: liczb stałopozycyjnych, zmiennopozycyjnych oraz zespolonych. Analiza podstawowych cech tych liczb oraz operacji na nich. Temat: Ciągi znaków i operacje na nich. Cel: Opanowanie zasad posługiwania się ciągami znaków w języku Python. Temat: Złożone struktury danych. Cel: Złożone struktury danych – tworzenie dostępne operacje oraz obszar zastosowanie. Praktyczne porównanie możliwości poszczególnych typów złożonych. Temat: Podstawowe struktury języków programowania. Cel: Wprowadzenie zasad działania podstawowych struktur języków programowania: wyrażenia matematyczne, wyrażenia logiczne, podstawianie, instrukcje warunkowe i pętle. Temat: Funkcje i procedury Cel: Wprowadzenie do programowania strukturalnego – tworzenie oraz wywoływanie procedur i funkcji. Zasady tworzenia aplikacji strukturalnych. Temat: Klasy, obiekty, metody. Cel: Wprowadzenie do programowania obiektowego – tworzenie klas, obiektów, metod, zmiennych obiektowych i klasowych. Zasady tworzenia aplikacji obiektowych. Temat: Obsługa wyjątków. Cel: Opanowanie zasad obsługi błędów w języku Python Temat: Obsługa plików. Cel: Opanowanie zasad obsługi plików i urządzeń blokowych w języku Python. Temat: Biblioteka standardowa. Cel: Prezentacji i praktyczne wykorzystanie możliwości biblioteki standardowej języka Python. Temat: Biblioteki zewnętrzne. Cel: Prezentacji i praktyczne wykorzystanie możliwości bibliotek zewnętrznych języka Python. Temat: Modularyzacja programu. Cel: Podział aplikacji na moduły jako narzędzie do organizacji projektu programistycznego.	Ćwiczenia laboratoryjne	U1, K1

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

Semestr 3

Forma zajęć	
-------------	--

Ćwiczenia laboratoryjne	Metody prowadzenia zajęć:	
	Ćwiczenia laboratoryjne	
	Metody (sposoby) weryfikacji:	Udział:
	Sprawozdanie	100%
	Warunki zaliczenia przedmiotu:	
	Sprawozdania z ćwiczeń laboratoryjnych ocenione co najmniej na 3.0	

Semestr 4

Forma zajęć		
Ćwiczenia laboratoryjne	Metody prowadzenia zajęć:	
	Ćwiczenia laboratoryjne	
	Metody (sposoby) weryfikacji:	Udział:
	Sprawozdanie	100%
	Warunki zaliczenia przedmiotu:	
	Sprawozdania z ćwiczeń laboratoryjnych ocenione co najmniej na 3.0	

Efekt uczenia się dla przedmiotu	Metody (sposoby) weryfikacji
	Sprawozdanie
W1	x
U1	x
K1	x

5. Literatura

Literatura podstawowa

1. Mark Lutz M., Learning Python, O'Reilly Media, 2013

Literatura uzupełniająca

1. Nagar S., Introduction to Python for Engineers and Scientists: Open Source Solutions for Numerical Computation, Apress, 2017

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	Ćwiczenia laboratoryjne	20
Praca własna studenta	Przygotowanie do zajęć	20
	Konsultacje	15
	Inne (przygotowanie do egzaminu)	25
Łączny nakład pracy studenta		80
Liczba punktów ECTS		3

* Godzina (dydaktyczna) oznacza 45 minut