



**POLITECHNIKA
BYDGOSKA**

Wydział Inżynierii Mechanicznej

Karta przedmiotu
Programowanie w mechatronice

1. Informacje podstawowe

Kierunek studiów mechatronika Specjalność: mechatronika przemysłowa 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 03MCHMPRZN.DI6.1528.24 Języki wykładowe polski Obligatoryjność Obligatoryjny specjalnościowy Blok zajęciowy Przedmioty specjalnościowe	
Wymagania wstępne			
Przedmioty wprowadzające		Języki programowania, Algorytmy i struktury danych	
Koordynator		Krzysztof Nowicki	
Okres Semestr 2		Forma zaliczenia Zaliczenie na ocenę Forma prowadzenia i godziny zajęć Wykład: 18 Ćwiczenia audytoryjne: 9	Liczba punktów ECTS 4.0

Okres Semestr 3	Forma zaliczenia Egzamin Forma prowadzenia i godziny zajęć Wykład: 18 Ćwiczenia laboratoryjne: 18 Ćwiczenia projektowe: 9	Liczba punktów ECTS 3.0
---------------------------	--	-----------------------------------

2. Efekty uczenia się dla przedmiotu

Kod	Opis efektów uczenia się	Odniesienie do kierunkowych efektów uczenia się	Odniesienie do charakterystyk PRK
Wiedza:			
W1	Posiada wiedzę na temat możliwości zastosowania systemów do szybkiego programowania systemów mechatronicznych w oparciu o system ROS.	MCH_O2_K_W01	P7S_WG P7S_WK P7S_WG_inż
W2	Posiada wiedzę na temat metod modelowania systemów informatycznych w oparciu o język UML.	MCH_O2_K_W04	P7S_WG P7S_WG_inż
Umiejętności:			
U1	Potrafi samodzielnie opracować oprogramowanie urządzenia mechatronicznego w oparciu o system ROS.	MCH_O2_K_U04	P7S_UW P7S_UK P7S_UO P7S_UW_inż
U2	Potrafi samodzielnie opracować model systemu informatycznego w języku UML.	MCH_O2_K_U05	P7S_UW P7S_UO P7S_UU P7S_UW_inż
Kompetencje społeczne:			
K1	Potrafi samodzielnie skorzystać z dostępnych źródeł informacji w celu szybkiej optymalizacji istniejącego oprogramowania systemu mechatronicznego.	MCH_O2_K_K02	P7S_KK P7S_KR

3. Treści programowe

Lp.	Treści programowe	Formy zajęć	Efekty uczenia się dla przedmiotu
1.	Wprowadzenie do ROS (Robotic Operation System). Elementy systemu ROS - tematy, komunikaty, węzły, nodelets. Środowisko pracy systemu ROS. Środowisko uruchomieniowe. Symulator Gazebo. Struktura pakietów. Biblioteki programistyczne ROS dla C++ oraz Python. Serwer parametrów systemu ROS. Symulator RViz. System transformacji ROS. Przeglądarka sieci ROS - rqt. Modele robotów URDF oraz środowisko symulacji SDF. Usługi i akcje systemu ROS. Sterowanie upływem czasu. Pliki komunikatów. Wprowadzenie do systemu ROS 2.	Wykład	W1, W2, K1
2.	Uruchomienie środowiska programowania robotów przemysłowych MoveIt dla systemu ROS 1 lub 2 wykonanie ćwiczeń związanych z planowaniem ruchów robota w środowisku symulacyjnym.	Ćwiczenia audytoryjne, Ćwiczenia laboratoryjne	U1, U2, K1

Lp.	Treści programowe	Formy zajęć	Efekty uczenia się dla przedmiotu
3.	Podstawy modelowania w języku UML. Diagramy przypadków użycia. Diagramy aktywności. Diagramy sekwencji. Diagramy komunikacji. Przeglądowe diagramy interakcji. Diagramy czasowe. Diagramy maszyny stanowej. Diagramy pakietów. Diagramy komponentów. Diagramy wdrożenia. Diagramy struktur złożonych.	Wykład	W1, W2, K1
4.	Opracowanie projektu własnego systemu informatycznego w oparciu o język UML w systemie Visual Paradigm.	Ćwiczenia projektowe	U1, U2, K1

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

Semestr 2

Forma zajęć		
Wykład	Metody prowadzenia zajęć:	
	Wykład, Case study	
	Metody (sposoby) weryfikacji:	Udział:
	Zaliczenie pisemne	100%
	Warunki zaliczenia przedmiotu:	
	Wymagane uzyskanie 50% punktów z pytań egzaminacyjnych	
Ćwiczenia audytoryjne	Metody prowadzenia zajęć:	
	Case study	
	Metody (sposoby) weryfikacji:	Udział:
	Sprawdzian zdolności metodycznych i umiejętności technicznych	100%
	Warunki zaliczenia przedmiotu:	
	Opracowanie planu funkcjonowanie robota przemysłowego w systemie ROS	

Semestr 3

Forma zajęć		
Wykład	Metody prowadzenia zajęć:	
	Wykład, Case study	
	Metody (sposoby) weryfikacji:	Udział:
	Egzamin pisemny	100%
	Warunki zaliczenia przedmiotu:	
	Uzyskanie 50% punktów z pytań egzaminacyjnych	

Ćwiczenia laboratoryjne	Metody prowadzenia zajęć:	
	Projekt	
	Metody (sposoby) weryfikacji:	Udział:
	Projekt	100%
	Warunki zaliczenia przedmiotu:	
	Wykonanie projektu systemu informatycznego w języku UML	
Ćwiczenia projektowe	Metody prowadzenia zajęć:	
	Projekt	
	Metody (sposoby) weryfikacji:	Udział:
	Projekt	100%
	Warunki zaliczenia przedmiotu:	
	Przygotowanie dokumentacji projektowej - ocena powyżej 2,95.	

Efekt uczenia się dla przedmiotu	Metody (sposoby) weryfikacji			
	Zaliczenie pisemne	Sprawdzian zdolności metodycznych i umiejętności technicznych	Egzamin pisemny	Projekt
W1	x		x	
W2	x		x	
U1		x		x
U2		x		x
K1	x	x	x	x

5. Literatura

Literatura podstawowa

1. Anis Koubaa, Robot Operating System (ROS). The Complete Reference (Volume 1), Springer 2016
2. Anis Koubaa, Robot Operating System (ROS). The Complete Reference (Volume 2), Springer 2017
3. Anis Koubaa, Robot Operating System (ROS). The Complete Reference (Volume 3), Springer 2018
4. G. Booch, J. Rumbaugh, I. Jacobson I: The Unified Modeling Language User Guide (2nd Edition). Addison-Wesley Professional, 2005.
5. Eric J. Naiburg, Robert A. Maksimchuk: UML dla zwykłych śmiertelników. PWN, 2007.

Literatura uzupełniająca

1. Strona projektu ROS: www.ros.org
2. Strona projektu UML: www.uml.org

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	36
	Ćwiczenia audytoryjne	9
	Ćwiczenia laboratoryjne	18
	Ćwiczenia projektowe	9
Praca własna studenta	Studiowanie literatury	30
	Przygotowanie do zaliczenia	15
	Przygotowanie raportu	15
	Konsultacje	15
	Inne (przygotowanie do egzaminu)	20
	Przygotowanie projektu	8
Łączny nakład pracy studenta		175
Liczba punktów ECTS		7

* Godzina (dydaktyczna) oznacza 45 minut