



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Programowanie systemów sterowania w mechatronice

Przedmiot

Kierunek studiów

Mechatronika

Studia w zakresie (specjalność)

Konstrukcje i sterowanie urządzeń mechatronicznych

Poziom studiów

drugiego stopnia

Forma studiów

stacjonarne

Rok/semestr

2/3

Profil studiów

ogólnoakademicki

Język oferowanego przedmiotu

polski

Wymagalność

obieralny

Liczba godzin

Wykład

Laboratoria

Inne (np. online)

30

Ćwiczenia

Projekty/seminaria

Liczba punktów ECTS

2

Wykładowcy

Odpowiedzialny za przedmiot/wykładowca:

mgr inż. Tymoteusz Lindner

Odpowiedzialny za przedmiot/wykładowca:

email: tymoteusz.lindner@put.poznan.pl

Wydział Inżynierii Mechanicznej

ul. Piotrowo 3, 60-965 Poznań

Wymagania wstępne

Podstawowa znajomość matematyki, informatyki, języków programowania. Znajomość obsługi komputera, systemu operacyjnego Windows i Linux oraz programowania w C++, C# i Python, budowanie algorytmów z wykorzystaniem elementów języka C++, C# i Python do sterowania urządzeniami mechatronicznymi. Świadomość potrzeby poszerzania wiedzy i umiejętności. Umiejętność przestrzegania zasad obowiązujących na zajęciach laboratoryjnych.

Cel przedmiotu

Poznanie podstaw programowania obiektowego, nabycie umiejętności posługiwania się klasami i strukturami. Poznanie podstaw sterowania robotami mobilnymi i ramionami robotów. Możliwość sterowania robotami w środowisku symulacyjnym. Poznanie i projektowanie oprogramowania do sterowania robotami. Autonomiczne sterowanie robotami.



Przedmiotowe efekty uczenia się

Wiedza

Ma poszerzoną wiedzę z zakresu sterowania robotami, w tym robotami mobilnymi i ramionami robotów.

Ma podstawową wiedzę w zakresie programowania autonomicznych systemów sterowania robotami mobilnymi.

Ma szeroką wiedzę z zakresu programowania w językach: C++, C# i Python. Ma również podstawową wiedzę na temat budowy, działania i programowania systemów sterowania.

Ma poszerzoną wiedzę z zakresu mechatroniki z zakresu analizy i projektowania złożonych układów mechatronicznych, teorii systemów oraz zastosowania modelowania i symulacji w projektowaniu mechatronicznym.

Ma poszerzoną wiedzę z zakresu informatyki ze znajomością obsługi systemów w czasie rzeczywistym, programowania z wykorzystaniem algorytmów do przetwarzania sygnałów i sterowania oraz podstaw przetwarzania i analizy obrazów oraz przygotowywania dokumentacji.

Umiejętności

Umiejętność pozyskiwania informacji o mechatronice z Internetu, biblioteki i czytelni oraz z innych źródeł. W szczególności potrafi poprawnie wskazać źródła potrzebnych informacji.

Umiejętność pisania programów w języku C++, C# i Python. Potrafi pisać i używać programy służące do projektowania, analizy, symulacji i sterowania.

Posiada umiejętności wykorzystania narzędzi informatycznych w pozyskiwaniu i integracji informacji, projektowaniu i sterowaniu urządzeniami mechatronicznymi.

Kompetencje społeczne

1. Zrozumienie wymogu uczenia się przez całe życie; umiejętność inspirowania i organizowania procesu uczenia się innych ludzi.
2. Ma świadomość roli robotów autonomicznych we współczesnej gospodarce i ich znaczeniu dla rozwoju społeczeństwa i środowiska.
3. Umiejętność myślenia i działania w sposób kreatywny i przedsiębiorczy.

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Zaliczenie na podstawie sprawozdań z ćwiczeń laboratoryjnych oraz zaliczenia pisemnego składającego się z zadań programistycznych.

Skala ocen:

<51%-60%> punktów - 3.0,



(60%-70%> punktów - 3.5,

(70%-80%> punktów - 4.0,

(80%-90%> punktów - 4.5,

(90%-100%> punktów - 5.0.

Nagradzanie praktycznej wiedzy zdobytej podczas poprzednich zajęć laboratoryjnych.

Praktyczne sprawdzenie umiejętności programowania robotów.

Ocena wiedzy i umiejętności związanych z realizacją zadań indywidualnych i grupowych na laboratorium.

Zdobywanie dodatkowych punktów za aktywność na zajęciach, zwłaszcza za:

- umiejętność pracy w zespole, który praktycznie wykonuje określone zadanie,
- wykonywanie dodatkowych zadań,
- finezję rozwiązań technicznych.

Treści programowe

- Programowanie w języku C++, C# i Python.
- Instalacja i konfiguracja systemu operacyjnego Ubuntu.
- Instalacja i konfiguracja systemu ROS.
- Instalacja i konfiguracja środowiska symulacyjnego.
- Podstawy systemu operacyjnego Linux, Embedded Linux i ROS.
- Projektowanie, budowa i programowanie systemu sterowania robotami mobilnymi i ramionami robotów w środowisku ROS.
- Symulacja systemów sterowania w środowiskach symulacyjnych.
- Budowa oprogramowania do obsługi wybranych elementów systemu sterowania.
- Sterowanie autonomiczne robotami mobilnymi.

Metody dydaktyczne

Indywidualne ćwiczenia praktyczne, wykonywanie eksperymentów, rozwiązywanie zadań, dyskusja, praca w zespole



Literatura

Podstawowa

1. Dokumentacja ROS
2. ROS Robotics Projects, Lentin Joseph
3. Mastering ROS for Robotics Programming, Lentin Joseph, Jonathan Cacace

Uzupełniająca

1. Python. Wprowadzenie, Mark Lutz
2. Opus magnum C++ 11. Programowanie w języku C++, Jerzy Grębosz

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	50	2,0
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	30	1,0
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwίων/egzaminu, wykonanie projektu) ¹	20	1,0

¹ niepotrzebne skreślić lub dopisać inne czynności