



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Wirtualne prototypowanie w automatyzacji procesów [N2AiR1-ISA>WPwAP]

Przedmiot

Kierunek studiów

Automatyka i robotyka

Rok/Semestr

1/1

Studia w zakresie (specjalność)

Inteligentne systemy automatyki

Profil studiów

ogólnoakademicki

Poziom studiów

drugiego stopnia

Język oferowanego przedmiotu

polski

Forma studiów

niestacjonarne

Wymagalność

obligatoryjny

Liczba godzin

Wykład

20

Laboratorium

20

Inne

0

Ćwiczenia

0

Projekty/seminaria

0

Liczba punktów ECTS

4,00

Koordynatorzy

dr hab. inż. Konrad Urbański

konrad.urbanski@put.poznan.pl

Wykładowcy

Wymagania wstępne

Student rozpoczynający ten przedmiot powinien posiadać wiedzę z automatyki i robotyki odpowiadającą 6 poziomowi Polskiej Ramy Kwalifikacji, w szczególności wiedzę z zakresu podstaw automatyki, teorii liniowych systemów dynamicznych oraz umiejętność programowania. Powinien również rozumieć konieczność poszerzania swoich kompetencji i być gotowy do podjęcia współpracy w zespole.

Cel przedmiotu

Zapoznanie studentów z różnymi środowiskami programistycznymi służącymi do modelowania i symulacji obiektów dynamicznych w kontekście szybkiego prototypowania systemów sterowania. Przedstawienie podstawowych funkcji i możliwości wybranych środowisk programowania. Prezentacja sposobów użycia we własnych programach różnych metod modelowania obiektów. Zapoznanie z wybranymi sposobami optymalizacji czasu symulacji oraz metod doboru parametrów.

Przedmiotowe efekty uczenia się

Wiedza

ma uporządkowaną i pogłębioną wiedzę z zakresu modelowania oraz identyfikacji systemów; ma wiedzę o trendach rozwojowych i najistotniejszych nowych osiągnięciach z zakresu automatyki i robotyki i

pokrewnych dyscyplin naukowych

Umiejętności

potrafi przeprowadzić symulację i analizę działania złożonych układów automatyki i robotyki oraz zaplanować i przeprowadzić weryfikację eksperymentalną; potrafi krytycznie ocenić i dobrać odpowiednie metody i narzędzia do rozwiązania zadania z zakresu automatyki i robotyki; potrafi wykorzystać narzędzia nowatorskie i niekonwencjonalne z zakresu automatyki i robotyki;

Kompetencje społeczne

rozumie potrzebę i zna możliwości ciągłego doskonalenia się – podnoszenia kompetencji zawodowych, osobistych i społecznych, potrafi inspirować i organizować proces uczenia się innych osób;

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Wykład: kolokwium na koniec semestru

Laboratorium: bieżące sprawdzanie umiejętności w ramach realizowanych zadań z zakresu tworzenia modeli obiektów i struktur sterowania

Treści programowe

Środowiska symulacji i modelowania układów mechatronicznych

Narzędzia sterowania, analizy i wizualizacji z użyciem ROS (Robot Operating System).

Tematyka zajęć

Funkcje obliczeniowe i analizy wyników środowiska Matlab oraz wybranych modułów języka Python

Narzędzia optymalizacyjne w Matlabie i w wybranych modułach języka Python

Wykorzystanie specjalizowanych bibliotek w środowisku programowania i symulacji

Realizacja obliczeń sztucznych sieci neuronowych w różnych środowiskach, przenoszenie struktur SSN, sieci płytkie i głębokie

Podstawowe operacje oraz paczki języka Python w modelowaniu i symulacji

Izolowane środowisko virtualenv/virtualenvwrapper dla języka Python

Wprowadzenie do środowiska konteneryzacji Docker

Wykorzystanie symulatora Gazebo do modelowania układów mechatronicznych

Wykorzystanie ROS (Robot Operating System) do komunikacji i sterowania układami mechatronicznymi.

Metody dydaktyczne

wykład:

-wykład z prezentacją multimedialną (w tym: rysunki, zdjęcia, animacje, dźwięk, filmy) uzupełniany przykładami podawanymi na tablicy

-wykład prowadzony w sposób interaktywny z formułowaniem pytań do grupy studentów

-przedstawianie nowego tematu poprzedzone przypomnieniem treści powiązanych, znanych studentom z innych przedmiotów

laboratoria:

-praca w zespołach

-eksperymenty obliczeniowe

Literatura

Podstawowa:

1. Internetowe tutoriale dla aktualnej wersji języka Python 3.x
2. Dokumentacja (internet) wybranych modułów języka Python dla wersji 3.x
3. Dokumentacja (internet) GazeboSim dla określonej wersji
4. Dokumentacja (internet) ROS dla określonej wersji
5. Dokumentacja (internet) Docker
6. Internetowe tutoriale i baza wiedzy firmowane przez MathWorks®.

Uzupełniająca:

1. Modelowanie matematyczne systemów, J. Gutenbaum, Wyd. 3 rozsz. i popr. Warszawa: Exit 2003
2. Modelowanie i symulacja układów i procesów dynamicznych, Stanisław Osowski, Warszawa 2007

3. Ćwiczenia z automatyki w Matlabie i Simulinku, Jerzy Brzózka, Wydawnictwo EDU-MIKOM, Warszawa 1997
4. MATLAB The Language of Technical Computing, The Math Works, Inc., (wydanie od 2008r.)
5. Automatyzacja nudnych zadań z Pythonem, A. Sweigart, wydanie jak najnowsze
6. Python: wprowadzenie, M. Lutz, Helion, wydanie jak najnowsze
7. Python dla każdego. Podstawy programowania, M. Dawson, wydanie jak najnowsze.

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	100	4,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	40	1,50
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu)	60	2,50