



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Języki programowania [S2MiBP1>JP]

Przedmiot

Kierunek studiów

Mechanika i budowa pojazdów

Studia w zakresie (specjalność)

Pojazdy samochodowe

Poziom studiów

drugiego stopnia

Forma studiów

stacjonarne

Rok/Semestr

1/1

Profil studiów

ogólnoakademicki

Język oferowanego przedmiotu

polski

Wymagalność

obligatoryjny

Liczba godzin

Wykład

15

Laboratorium

15

Inne

0

Ćwiczenia

0

Projekty/seminaria

0

Liczba punktów ECTS

2,00

Koordynatorzy

dr inż. Przemysław Grzymisławski

przemyslaw.grzymislawski@put.poznan.pl

Wykładowcy

Wymagania wstępne

WIEDZA: Student ma elementarną wiedzę w zakresie podstaw informatyki, tj o architekturze komputera, typach zmiennych, ogólną wiedzę o językach wysokiego poziomu używanych w programowaniu komputerów. **UMIEJĘTNOŚCI:** Student umie posługiwać się pojęciami w opisie języków programowania oraz potrafi rozwiązywać konkretne problemy pojawiające się w czasie pisania programów.

KOMPETENCJE SPOŁECZNE: Student potrafi współpracować w grupie, przyjmując w niej różne role, potrafi określić priorytety ważne przy rozwiązywaniu stawianych przed nim zadań oraz wykazuje samodzielność w rozwiązywaniu problemów, zdobywaniu i doskonaleniu nabytej wiedzy i umiejętności.

Cel przedmiotu

Celem przedmiotu jest przekazanie studentom informacji z zakresu wybranych języków programowania (Python, C++), definicji oraz pojęć. Studenci uzyskują wiedzę i umiejętności w zakresie tworzenia programów komputerowych.

Przedmiotowe efekty uczenia się

Wiedza:

Ma poszerzoną wiedzę w zakresie informatyki, dotyczącą programowania komputerów oraz programów

do obliczeń inżynierskich w zakresie symulacji komputerowej układów fizycznych.
Zna współczesne metody inżynierskiej grafiki komputerowej i teoretyczne podstawy obliczeń inżynierskich metodą elementów skończonych.
Posiada poszerzoną wiedzę o normach dotyczących maszyn roboczych w zakresie metod obliczania i badania maszyn, bezpieczeństwa, w tym bezpieczeństwa ruchu drogowego, ochrony środowiska a także interface'u mechanicznego i elektrycznego.

Umiejętności:

Potrafi posłużyć się popularnym systemem do obliczeń numerycznych do zaprogramowania prostego zadania symulacji systemu o niewielkiej liczbie stopni swobody.
Potrafi napisać prosty program komputerowy z wykorzystaniem nowoczesnych środowisk RAD w znany sobie języku do obliczeń optymalizacyjnych konstrukcji z wykorzystaniem przyswojonych elementarnych metod numerycznych.
Potrafi prowadzić debatę.

Kompetencje społeczne:

Jest gotów do krytycznej oceny posiadanej wiedzy i odbieranych treści.
Jest gotów do myślenia i działania w sposób przedsiębiorczy.
Jest gotów do odpowiedzialnego pełnienia ról zawodowych z uwzględnieniem zmieniających się potrzeb społecznych, w tym:
– rozwijania dorobku zawodu,
– podtrzymywania etosu zawodu,
– przestrzegania i rozwijania zasad etyki zawodowej oraz działania na rzecz przestrzegania tych zasad.

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Wiedza nabyta w ramach wykładu jest weryfikowana przez 45-minutowe kolokwia realizowane na ostatnim wykładzie. Kolokwium składa się z 6-10 pytań różnie punktowanych. Próg zaliczeniowy: 50% punktów. Zagadnienia zaliczeniowe, na podstawie których opracowywane są pytania zostaną przesłane studentom drogą mailową z wykorzystaniem systemu uczelnianej poczty elektronicznej.
Umiejętności nabyte w ramach zajęć laboratoryjnych weryfikowane są na podstawie napisanych przez studenta 4 krótkich programów w języku Python. Próg zaliczeniowy to 3 poprawnie działające programy.

Treści programowe

Budowa programów komputerowych. Porównanie konstrukcji języków C++ Python. Omówienie deklaracji stałych, zmiennych oraz typów zmiennych. Operatory arytmetyczne. Funkcje – wartość funkcji i parametry, przekazywanie argumentów przez wartość oraz referencję. Wyrażenia – przypisanie, porównywanie danych, priorytety i łączność. Rozgałęzienia i pętle. Tablice i struktury. Biblioteki standardowe języka Python. Podstawowe pojęcia związane z obliczeniami numerycznymi: iteracja, interpolacja, aproksymacja, ekstrapolacja, całkowanie numeryczne, rozwiązywanie równań różniczkowych zwyczajnych. Algorytmy: obliczania pierwiastka kwadratowego, znajdowania miejsc zerowych funkcji - metoda Newtona, siecznych i bisekcji, całkowania numerycznego z ekstrapolacją Richardsona, rozwiązywania równań różniczkowych zwyczajnych metodą Eulera oraz punktu środkowego. Procedury dla tych algorytmów w języku Python.

Tematyka zajęć

brak

Metody dydaktyczne

1. Wykład: prezentacja multimedialna, ilustrowana przykładami podawanymi na tablicy.
2. Laboratorium: przykłady podawane na tablicy oraz wykonanie zadań podanych przez prowadzącego.

Literatura

Podstawowa

1. Michael Dawson, Python dla każdego - podstawy programowania, Wydawnictwo Helion, Wydanie III.

2. <https://docs.python.org/3/>

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	50	2,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	30	1,00
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu)	20	1,00