



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Języki programowania

Kierunek studiów

Inżynieria biomedyczna

Studia w zakresie (specjalność)

Poziom studiów

pierwszego stopnia

Forma studiów

stacjonarne

Przedmiot

Rok/semestr

2/4

Profil studiów

ogólnoakademicki

Język oferowanego przedmiotu

polski

Wymagalność

obligatoryjny

Liczba godzin

Wykład

15

Ćwiczenia

Laboratoria

15

Projekty/seminaria

Inne (np. online)

Liczba punktów

2

Wykładowcy

Odpowiedzialny za przedmiot/wykładowca:

Dr hab. inż. Maciej TABASZEWSKI

e-mail: Maciej.Tabaszewski@put.poznan.pl

tel. 61 665 23 90

Wydział Inżynierii Mechanicznej

ul. Piotrowo 3, 60-965 Poznań

tel.: 61 665 23 61

Odpowiedzialny za przedmiot/wykładowca:

Wymagania wstępne

Wiedza podstawowa dotycząca logiki matematycznej i informatyki

Cel przedmiotu

Przekazanie wiedzy pozwalającej na samodzielne programowanie proceduralne i obiektowe.

Przedmiotowe efekty uczenia się

Wiedza

Student rozpoznaje i zna cechy programowania proceduralnego, obiektowego, wizualnego

Student zna podstawowe konstrukcje wybranych języków programowania



Student zna pojęcia klas, struktur, obiektów, dziedziczenia, polimorfizmu, hermetyzacji

Umiejętności

Student potrafi tworzyć dedykowane oprogramowanie

Kompetencje społeczne

Student rozumie rolę informatyzacji we współczesnej gospodarce. Potrafi w niej twórczo uczestniczyć

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Wykład: Zaliczenie na podstawie testu, 20 pytań zamkniętych, próg zaliczenia - 50% maksimum punktów

Laboratorium: Zaliczenie na podstawie testów przed każdymi ćwiczeniami, próg zaliczenia - 50% maksimum punktów

Treści programowe

Wykład: Ogólne zasady konstruowania programów. Kompilatory i interpretery. Programowanie w językach niskiego i wysokiego poziomu, przegląd i podział języków. Wizualne języki programowania. Programowanie strukturalne. Podstawy programowania w języku Python. Cechy języka. Zmienne, typy danych (listy, krotki, słowniki, zbiory, liczby zespolone, typ logiczny i inne), referencje, operatory, pętle, instrukcje warunkowe, funkcje wejścia i wyjścia, import bibliotek. Programowanie zorientowane obiektowo. Pojęcia hermetyzacji, klas, obiektów, dziedziczenia, polimorfizmu. Podstawy programowania obiektowego w języku Python. Zapis i odczyt plików, obsługa wyjątków, przestrzenie nazw. Biblioteki przydatne do obliczeń numerycznych.

Laboratorium: Programowanie w języku Python, przykłady : wprowadzanie i wyprowadzanie danych, proste obliczenia, działania na łańcuchach, formatowanie wydruku, operacje na dacie, zastosowanie instrukcji warunkowej, wyrażenia listowe, instrukcji wyboru, pętli, instrukcje break, continue, pass, zapis i odczyt pliku tekstowego, binarnego, tworzenie funkcji. Programowanie obiektowe przykłady: tworzenie klas, dziedziczenie, przeciążanie operatorów, wykorzystanie bibliotek

Metody dydaktyczne

Wykład: prezentacje multimedialne dotyczące teorii i przykładów praktycznych

Ćwiczenia laboratoryjne: ćwiczenia praktyczne, rozwiązywanie zadań

Literatura

Podstawowa

Michael Dawson, Python dla każdego, podstawy programowania, Wydanie III, Helion, Gliwice 2014

Ana Bell, Python, uczy się programowania, Helion, Gliwice 2018



Uzupełniająca

Mark Lutz, Python, wprowadzenie, Helion, Gliwice 2013

Wes McKinney, Python w analizie danych, Wydanie II, Helion, Gliwice 2018

Douwe Osinga, Deep Learning, Receptury, Helion, Gliwice 2019

Robert C. Martin, Czysty kod, podręcznik dobrego programisty, Helion, Gliwice 2014

Aurelien Geron, Uczenie maszynowe z użyciem Scikit-Learn i TensorFlow, Helion, Gliwice 2018

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	50	2,0
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	35	1,5
Praca własna studenta (Praca własna studenta (studia literaturowe, przygotowanie do wykładu, laboratorium, zaliczenia) ¹	15	0,5

¹ niepotrzebne skreślić lub dopisać inne czynności