



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Języki programowania [S1IBio1>JP_1]

Przedmiot

Kierunek studiów

Inżynieria biomedyczna

Rok/Semestr

1/2

Studia w zakresie (specjalność)

–

Profil studiów

ogólnoakademicki

Poziom studiów

pierwszego stopnia

Język oferowanego przedmiotu

polski

Forma studiów

stacjonarne

Wymagalność

obligatoryjny

Liczba godzin

Wykład

15

Laboratorium

30

Inne (np. online)

0

Ćwiczenia

0

Projekty/seminaria

0

Liczba punktów ECTS

4,00

Koordynatorzy

dr hab. inż. Maciej Tabaszewski
maciej.tabaszewski@put.poznan.pl

Wykładowcy

Wymagania wstępne

Wiedza podstawowa dotycząca logiki matematycznej i informatyki

Cel przedmiotu

Wykład ma za zadanie przedstawienie podstaw wiedzy teoretycznej dotyczącej języków programowania, charakterystyki wybranych języków (Python, C, C++), oraz programowania proceduralnego i obiektowego. Omawiane są również na przykładach konstrukcje programów w tych językach. Laboratoria kładą nacisk na stronę praktyczną umiejętności pisania programów, poprzez rozwiązywanie krótkich łatwych do szybkiego zrozumienia problemów z zakresu statystyki, modelowania szeregów czasowych, analizy sygnałów, podstawowych metod numerycznych itp.

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: Egzamin na podstawie testu, 20- 25 pytań zamkniętych, próg zaliczenia - 50% maksimum punktów

Laboratorium: Zaliczenie na podstawie testów przed ćwiczeniami oraz na podstawie dwóch sprawdzianów praktycznych - 50% maksimum punktów

Treści programowe

Podstawowe zasady i konstrukcje w programowaniu wysokopoziomowym. Zagadnienia związane z programowaniem strukturalnym, obiektowym. Tworzenie programów w językach Python, C, C++ .
Trenowanie umiejętności w oparciu o tworzenie prostych programów rozwiązujących konkretne zagadnienia.

Tematyka zajęć

Wykład:

Ogólne zasady konstruowania programów. Kompilatory i interpretry. Programowanie w językach wysokiego poziomu, przegląd i podział języków. Charakterystyka wybranych języków. Wizualne języki programowania. Programowanie funkcyjne. Programowanie obiektowe. Rekurencja.

Podstawy programowania w językach Python, C/C++

Język Python - zmienne, typy danych, operatory, funkcje wbudowane, funkcje wejścia i wyjścia, referencje, pętle, instrukcja warunkowa, łańcuchy, listy, krotki, słowniki, zbiory, funkcje (definiowane, sposoby przekazywania argumentów), wyrażenia listowe, funkcje lambda, domknięcia, zapis i odczyt z plików tekstowych i binarnych. Moduły. Tworzenie własnych klas, przeciążanie operatorów.

Obsługa wyjątków. Funkcje o zmiennej liczbie argumentów. Tworzenie wykresów - biblioteka matplotlib, podstawowe obliczenia z wykorzystaniem modułu numpy

Język C/C++

biblioteki, proces kompilacji, łączenia, preprocesor, zmienne, deklaracje i definicje zmiennych, tablice, wskaźniki, arytmetyka wskaźników, operatory, pętle, instrukcja warunkowa, instrukcja wyboru, struktury, tworzenie funkcji, funkcje wejścia i wyjścia, zapis i odczyt z pliku tekstowego i binarnego, funkcje o zmiennej liczbie argumentów, dynamiczny przydział pamięci.

C++

Referencje, programowanie zorientowane obiektowo. Pojęcia klas, obiektów, hermetyzacji, dziedziczenia, polimorfizmu, abstrakcji. Przeciążanie operatorów, strumienie, wyjątki, przestrzenie nazw, biblioteka STL

Laboratorium:

Tworzenie programów związanych z tematyką: prostych modeli ekonomicznych, modelowania szeregów czasowych, statystyki (w tym regresji liniowej i analizy korelacji), analizy sygnałów cyfrowych, wybranych prostych metod numerycznych.

Podstawy programowania w języku Python - funkcje wbudowane, operacje wejścia - wyjścia, proste obliczenia, symulacje, podstawy sterowania kodem, łańcuchy i listy, definiowanie i wywoływanie funkcji (ćwiczenia 1-4).

Słowniki i zbiory, zapis do pliku i odczyt z pliku, sortowanie, filtrowanie, poszukiwanie elementów (ćwiczenie 5,6).

Import danych z programu MATLAB oraz dowolnych plików binarnych o znanej strukturze (ćwiczenie 7).

Wykorzystanie matplotlib do ilustracji graficznej wyników oraz moduł random. (ćwiczenie 8)

Podstawy programowania obiektowego, tworzenie własnych klas i przeciążanie operatorów (ćwiczenia 9,10).

Wykorzystanie numpy do obliczeń na macierzach (ćwiczenie 11).

Praktyczne sprawdzenie umiejętności (ćwiczenie 12)

Podstawy programowania w C/C++ - kompilacja programu, zmienne i typy zmiennych, operacje wejścia wyjścia, tworzenie funkcji, instrukcje sterujące kodem, tablice i wskaźniki, podstawy programowania obiektowego w tym przeciążanie operatorów, tworzenie szablonów funkcji (ćwiczenia 13,14)

Praktyczne sprawdzenie umiejętności (ćwiczenie 15) .

Metody dydaktyczne

Wykład: prezentacje multimedialne dotyczące teorii i przykładów praktycznych

Ćwiczenia laboratoryjne: ćwiczenia praktyczne, pisanie krótkich programów, wspólnie i samodzielnie

Literatura

Podstawowa:

M. Dawson, Python dla każdego, podstawy programowania, Wydanie III, Helion

A. Bell, Python, uczymy się programowania, Helion

M. Lutz, Python, wprowadzenie, Helion

J. Liberty, S.Rao, B.L. Jones, C++ dla każdego, Wydanie II, Helion

B. Stroustrup, Programowanie, Teoria i praktyka z wykorzystaniem C++, Helion

Uzupełniająca:

W. McKinney, Python w analizie danych, Wydanie II, Helion,

R. Sedgewick, Algorytmy w C++, Wydawnictwo RM, Warszawa

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	100	4,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	47	2,00
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu)	53	2,00