



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Programowanie II [S1MwT1>Progr2]

Przedmiot

Kierunek studiów

Matematyka w technice

Rok/Semestr

2/4

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 Grzegorz Oleksik

grzegorz.oleksik@put.poznan.pl

Wykładowcy

Wymagania wstępne

Wiedza i umiejętności z kursów Programowania I oraz Technologii Informacyjnych. Umiejętność obsługi komputera, w tym programowania. Umiejętność efektywnego samokształcenia. Znajomość ograniczeń własnej wiedzy oraz zrozumienie potrzeby dalszego kształcenia.

Cel przedmiotu

Zdobycie umiejętności programowania obiektowego.

Przedmiotowe efekty uczenia się

Wiedza:

1. ma rozszerzoną i pogłębioną wiedzę z różnych działów matematyki wyższej oraz szczegółową wiedzę dotyczącą zastosowań metod i narzędzi matematycznych w naukach technicznych,
2. ma uporządkowaną i podbudowaną teoretycznie wiedzę z informatyki, zna co najmniej jeden pakiet oprogramowania lub język programowania.

Umiejętności:

1. potrafi skonstruować algorytm rozwiązania prostego zadania inżynierskiego oraz zaimplementować

- i przetestować go w wybranym środowisku programistycznym,
2. potrafi zgodnie z ogólnymi wymogami i dokumentacją techniczną eksploatować urządzenia, umie stosować zasady bezpieczeństwa i higieny pracy,
 3. potrafi pracować indywidualnie i w zespole; umie oszacować czas potrzebny na realizację zleconego zadania; potrafi opracować i zrealizować harmonogram prac zapewniający dotrzymanie terminu.

Kompetencje społeczne:

1. ma świadomość poziomu swojej wiedzy w odniesieniu do prowadzonych badań w naukach ścisłych i technicznych.

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 kolokwium składające się z różnie punktowanych pytań.

Umiejętności nabyte w ramach zajęć laboratoryjnych weryfikowane są podstawie opracowanych projektów lub kolokwium zaliczeniowego.

Treści programowe

- Obiektowe modelowanie dziedziny
- Wstęp do programowania obiektowego (niezależnie od konkretnego języka programowania)
- Wstęp do programowania obiektowego w języku Python
- Ochrona danych: hermetyzacja i modyfikatory dostępu
- Wstęp do diagramów UMLa
- Praca z obiektem: tworzenie, inicjalizacja, usuwanie
- Zagadnienie dziedziczenia oraz pojęcie polimorfizmu
- Klasy i metody abstrakcyjne
- Graficzny interfejs użytkownika
- Obsługa Wyjątków
- Metody specjalne w klasach:
 - a. `__len__`
 - b. `__str__` i `__repr__`
 - c. `__add__`, `__sub__`, `__mul__`, `__truediv__`
 - d. `__call__`

Metody dydaktyczne

1) wykłady:

- wykład z prezentacją uzupełniany przykładami podawanymi na tablicy,
- wykład prowadzony z formułowaniem pytań do grupy studentów lub do wskazywanych konkretnych studentów,
- uwzględnia się aktywność studentów w czasie zajęć przy wystawianiu oceny końcowej,
- w trakcie wykładu inicjowanie dyskusji,
- teoria przedstawiana w powiązaniu z praktyką,
- teoria przedstawiana w powiązaniu z aktualną wiedzą studentów,
- przedstawianie nowego tematu poprzedzone przypomnieniem treści powiązanych, znanych studentom z innych przedmiotów.

2) laboratorium:

- recenzowanie sprawozdań przez prowadzącego laboratoria i dyskusje nad komentarzami,
- korzystanie z narzędzi umożliwiających studentom wykonanie zadań w domu,
- demonstracje,
- praca w zespołach,
- eksperymenty obliczeniowe.

Literatura

Podstawowa

1. Lutz, M., Python. Wprowadzenie, Helion, najnowsze wydanie.
2. Steven F. Lott, Dusty Phillips: Programowanie zorientowane obiektowo w Pythonie. Helion 2023

Uzupełniająca:

1. Brian, J., Beazley, J., Python. Receptury, Helion, najnowsze wydanie
2. Eric, M., Python. Instrukcje dla Programisty, Helion, najnowsze wydanie
3. http://wazniak.mimuw.edu.pl/index.php?title=Programowanie_obiektowe
4. <https://analitik.edu.pl/programowanie-objektowe-w-python/>

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	100	4,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	45	2,00
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu)	55	2,00