



17S20-24ZJP
ECTS: 3.00
CYKL: 2024L

Sylabus przedmiotu - część A Zaawansowany język Python

TREŚCI MERYTORYCZNE

WYKŁAD

moduły i pakiety: pojęcie modułu, instrukcje import oraz from, ścieżka wyszukiwania modułów, nazwy prywatne modułu, moduły biblioteczne, przestrzenie nazw. Definiowanie klas i programowanie zorientowane obiektowo: różnica między inicjalizatorem a konstruktorem, dekoratory, właściwości, dziedziczenie, przeciążanie metod. Iteratory i generatory. Zaawansowane zagadnienia dotyczące funkcji w języku Python. Listy składane. Klasy i programowanie zorientowane obiektowo. Przykład klasy implementującej zbiór bitowy. Binarne typy sekwencyjne: Klasy bytearray i bytes. Moduł array. Formatowanie (interpolacja) łańcuchów znakowych. Moduły sys, os i os.path. Programowanie GUI w Pythonie. Programy wielowątkowe. Program pylint. Funkcje anonimowe, obiekty iterowalne. Narzędzia programowania funkcyjnego. Moduł itertools. Wyrażenia regularne, asercje, grupowanie, grupy. Przetwarzanie dokumentów XML.

ĆWICZENIA LABORATORYJNE

W ramach ćwiczeń laboratoryjnych studenci mają za zadanie napisanie i uruchamianie szeregu programów ilustrujących kolejne zagadnienia przedstawiane na wykładzie.

CEL KSZTAŁCENIA

Przedstawienie zaawansowanych pojęć i stosowanych rozwiązań występujących w programowaniu obiektowym i funkcyjnym przy wykorzystaniu możliwości języka Python. Nabycie umiejętności właściwego rozwiązywania problemów i poprawnego tworzenia programów, wykorzystując dostępne zasoby języka Python. Rozwinięcie umiejętności analitycznych i praktycznych, niezbędnych w innych dziedzinach informatyki.

OPIS EFEKTÓW UCZENIA SIĘ PRZEDMIOTU W ODNIESIENIU DO OPISU CHARAKTERYSTYK DRUGIEGO STOPNIA EFEKTÓW UCZENIA SIĘ DLA KWALIFIKACJI NA POZIOMACH 6-8 POLSKIEJ RAMY KWALIFIKACJI W ODNIESIENIU DO DYSCYPLIN NAUKOWYCH I EFEKTÓW KIERUNKOWYCH

Symbole efektów dyscyplinowych:

XP/IA_P7S_KK++, XP/IA_P7S_WG +++,
XP/IA_P7S_UW ++, InzA_P7S_WG+,
IT/ITA_P7S_WG+++, IT/ITA_P7S_KR+,
IT/ITA_P7S_KK++, XP/IA_P7S_KR +,
IT/ITA_P7S_UW++

Symbole efektów kierunkowych:

KA7_KR01+, KA7_UW09++, InzA7_WG06+,
KA7_KK01++, KA7_WG07++, KA7_WG08++

EFEKTY UCZENIA SIĘ:

Wiedza:

W1 - zaawansowane konstrukcje języka Python pozwalające na realizację zasady obiektowego paradygmatu programowania

Akty prawne określające efekty uczenia się:

494/2024

Dyscypliny: informatyka,
informatyka techniczna i
telekomunikacja

Status przedmiotu:

Fakultatywny

Grupa przedmiotów: C -
przedmioty

specjalnościowe/związane z
zakresem kształcenia

Kod: ISCED

Kierunek studiów:

Informatyka

Zakres kształcenia: Data
science w praktyce

Profil kształcenia:

Ogólnoakademicki

Forma studiów: Stacjonarne

Poziom studiów: Drugiego
stopnia

Rok/semestr: 1/1

Rodzaj zajęć: Wykład,
Ćwiczenia laboratoryjne

Liczba godzin w

semestrze: Wykład: 15.00,
Ćwiczenia laboratoryjne:
30.00

Język wykładowy: polski

Przedmioty

wprowadzające:

Wymagania

wstępne: Umiejętność
programowania w Pythonie na
poziomie
średniozaawansowanym

Nazwa jednostki org.

realizującej przedmiot:

Instytut Informatyki

**Osoba odpowiedzialna za
realizację**

przedmiotu: dr Agnieszka
Zbrzezny

e-mail:

agnieszka.zbrzezny@matman.
uwm.edu.pl

Uwagi dodatkowe:

W2 – działanie zaawansowanych konstrukcji używanych w języku Python, takich jak wyrażenia lambda, iteratory, generatory, dekoratory funkcji i klas.

Umiejętności:

U1 – definiować własne klasy oraz definiować dla nich operatory przeciążone

U2 – korzystać z wyrażeń lambda, iteratorów oraz generatorów; napisać program z wykorzystaniem graficznego interfejsu użytkownika

Kompetencje społeczne:

K1 – pracy samodzielnej i zespołowej nad rozwiązaniem postawionego problemu

K2 – systematycznego aktualizowania swojej wiedzy i umiejętności związanych z programowaniem i doskonalenia umiejętności zawodowych

K3 – dalszego samodzielnego kształcenia

FORMY I METODY DYDAKTYCZNE:

Wykład(W1;W2;K1;K2;K3;):Wykład tradycyjny (opcjonalnie wzbogacony o prezentację).

Ćwiczenia laboratoryjne(U1;U2;K1;K2;K3;):Ćwiczenia komputerowe-wspólne lub samodzielne pisanie programów i wykonywanie poleceń przygotowanych przez prowadzącego.

FORMA I WARUNKI WERYFIKACJI EFEKTÓW UCZENIA SIĘ:

Wykład (Sprawdzian pisemny) - Sprawdzian w formie testu jednokrotnego lub wielokrotnego wyboru w formie tradycyjnej, lub przy komputerze. - W1, W2, K1, K2, K3

Ćwiczenia laboratoryjne (Kolokwium praktyczne) - Dwa kolokwia komputerowe-rozwiązanie zadań programistycznych - U1, U2, K1, K2, K3

LITERATURA PODSTAWOWA:

1. Bill Lubanovic, *Python. Nowoczesne programowanie w prostych krokach. Wydanie II*, Wyd. Helion, R. 2020

LITERATURA UZUPEŁNIAJĄCA:

1. Micha Gorelick, Ian Ozsvald, *Wysoko wydajny Python. Efektywne programowanie w praktyce. Wydanie II*, Wyd. Helion, R. 2021

Szczegółowy opis przyznanej punktacji ECTS - część B

17S20-24ZJP

ECTS: 3.00

CYKL: 2024L

Zaawansowany język Python

Na przyznaną liczbę punktów ECTS składają się:

1. Godziny kontaktowe z nauczycielem akademickim:

| | |
|-------------------------------------|--------|
| - udział w: Wykład | 15.0 h |
| - udział w: Ćwiczenia laboratoryjne | 30.0 h |
| - konsultacje | 4.0 h |
| OGÓŁEM: 49.0 h | |

2. Samodzielna praca studenta:

| | |
|-------------------------------|---------|
| przygotowanie do laboratoriów | 10.00 h |
| przygotowanie do sprawdzianu | 10.00 h |
| przygotowanie do kolokwium | 15.00 h |

OGÓŁEM: 35.0 h

godziny kontaktowe + samodzielna praca studenta OGÓŁEM: 84.0 h

1 punkt ECTS = 25-30 h pracy przeciętnego studenta,
liczba punktów ECTS = $84.0 \text{ h} : 28.0 \text{ h/ECTS} = 3.00 \text{ ECTS}$

Średnio: **3.0 ECTS**

| | |
|---|-------------------|
| - w tym liczba punktów ECTS za godziny kontaktowe z bezpośrednim udziałem nauczyciela akademickiego | 1.75 punktów ECTS |
| - w tym liczba punktów ECTS za godziny realizowane w formie samodzielnej pracy studenta | 1.25 punktów ECTS |