

POLITECHNIKA KRAKOWSKA IM. TADEUSZA KOŚCIUSZKI

KARTA PRZEDMIOTU

obowiązuje studentów rozpoczynających studia w roku akademickim 2022/2023

Wydział Inżynierii Elektrycznej i Komputerowej

Kierunek studiów: Elektrotechnika i Automatyka

Profil: Ogólnoakademicki

Forma studiów: stacjonarne

Kod kierunku: E3

Stopień studiów: II

Specjalności: Automatyka w Przemysle 4.0

1 INFORMACJE O PRZEDMIOCIE

NAZWA PRZEDMIOTU	Programowanie obiektowe
NAZWA PRZEDMIOTU W JĘZYKU ANGIELSKIM	Object oriented programming
KOD PRZEDMIOTU	WIEiK ELEKTRO_OD_2019/2020 oIIS PS2 22/23
KATEGORIA PRZEDMIOTU	Przedmioty specjalnościowe
LICZBA PUNKTÓW ECTS	4.00
SEMESTRY	1

2 RODZAJ ZAJĘĆ, LICZBA GODZIN W PLANIE STUDIÓW

SEMESTR	WYKŁADY	ĆWICZENIA	LABORATORIA	LABORATORIA KOMPUTERO- WE	PROJEKTY	
1	25	0	0	25	0	0

3 CELE PRZEDMIOTU

Cel 1 Zapoznanie studentów z pojęciami dotyczącymi programowania obiektowego.

Cel 2 Nabycie umiejętności programowania obiektowego w języku C++ oraz poznanie fundamentów programowania w języku Python

Cel 3 Zapoznanie studentów z metodami reprezentacji i przetwarzania złożonych danych w programowaniu obiektowym.

4 WYMAGANIA WSTĘPNE W ZAKRESIE WIEDZY, UMIEJĘTNOŚCI I INNYCH KOMPETENCJI

1 Umiejętność programowania strukturalnego.

2 Znajomość języka C++

5 EFEKTY KSZTAŁCENIA

EK1 Wiedza Student zna paradygmaty programowania obiektowego

EK2 Umiejętności Student potrafi projektować programy w języku obiektowym

EK3 Wiedza Student zna podstawowe konstrukcje języka C++ oraz języka Python

EK4 Umiejętności Student potrafi programować aplikacje w językach C++ oraz Python

EK5 Umiejętności Student potrafi przetwarzać złożone dane w technice obiektowej

6 TREŚCI PROGRAMOWE

WYKŁADY		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
W1	Paradygmat programowania. Pojęcia: klasa, obiekt.	2
W2	Cztery kluczowe koncepcje: abstrakcja, enkapsulacja, dziedziczenie, polimorfizm	2
W3	Elementy programowania obiektowego: modyfikatory dostępu, akcesory, kontrola dostępu w praktyce, konstruktor,	3
W4	Elementy programowania obiektowego: dziedziczenie w praktyce, nadpisywanie metod, definiowanie interfejsów, implementowanie interfejsów, klasa abstrakcyjna, metody statyczne	3
W5	Jak zamienić teorię na praktykę: przykład użycia komunikacji, agregacja i kompozycja, zalety użycia interfejsów, spójność	3
W6	Fundamenty programowania w języku Python: zmienne i wartości, nazywanie elementów, liczby int oraz float, operacje arytmetyczne, string w Python, operacje oraz funkcje string, testowanie string	3
W7	Instrukcje warunkowe, pętle w Python, struktury danych, Funkcje w Python	3
W8	Klasy w Python: inicjalizer, metody, atrybuty klasy, dziedziczenie	2
W9	Istotne koncepcje: Wartości, obiekty oraz zmienne, mutowalność danych, identyczność oraz równość, funkcja jako obiekt pierwszej klasy dekoratory, generatory	2

WYKŁADY		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
W10	Ważne techniki: jak używamy modułów, definiowanie własnego modułu, błędy oraz wyjątki, instrukcja raise	2

LABORATORIA KOMPUTEROWE		
LP	TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH	LICZBA GODZIN
K1	Poznanie środowiska programowania w języku C++. Obiektowe modelowanie dziedziny.	2
K2	Podstawowe konstrukcje w języku C++	3
K3	Definiowanie klas w C++. Przeciążanie nazw metod.	2
K4	Tworzenie i inicjalizacja obiektów. Konstruktory.	2
K5	Dziedziczenie i interfejsy.	3
K6	Polimorfizm.	3
K7	Obsługa wyjątków i stosowanie asercji.	2
K8	Fundamenty programowania w języku Python: podstawowe typy, zmienne i wartości, nazywanie elementów, liczby int oraz float, operacje arytmetyczne, string w Python, operacje oraz funkcje string, testowanie string	2
K9	Instrukcje warunkowe, pętle, struktury danych, funkcje w Python	2
K10	Klasy w Python	2
K11	Ważne techniki w Python	2

7 NARZĘDZIA DYDAKTYCZNE

N1 Ćwiczenia laboratoryjne

N2 Konsultacje

N3 Prezentacje multimedialne

N4 Wykłady

N5 Dyskusja

8 OBCIĄŻENIE PRACĄ STUDENTA

FORMA AKTYWNOŚCI	ŚREDNIA LICZBA GODZIN NA ZREALIZOWANIE AKTYWNOŚCI
Godziny kontaktowe z nauczycielem akademickim, w tym:	
Godziny wynikające z planu studiów	50
Konsultacje przedmiotowe	15
Egzaminy i zaliczenia w sesji	5
Godziny bez udziału nauczyciela akademickiego wynikające z nakładu pracy studenta, w tym:	
Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	25
Opracowanie wyników	10
Przygotowanie raportu, projektu, prezentacji, dyskusji	0
przygotowanie się do egzaminu	15
SUMARYCZNA LICZBA GODZIN DLA PRZEDMIOTU WYNIKAJĄCA Z CAŁEGO NAKŁADU PRACY STUDENTA	120
SUMARYCZNA LICZBA PUNKTÓW ECTS DLA PRZEDMIOTU	4.00

9 SPOSOBY OCENY

OCENA FORMUJĄCA

F1 Ćwiczenie praktyczne

F2 Kolokwium zaliczeniowe

OCENA PODSUMOWUJĄCA

P1 Egzamin pisemny

P2 Średnia ważona ocen formujących

WARUNKI ZALICZENIA PRZEDMIOTU

W1 Warunkiem przystąpienia do egzaminu jest zaliczenie ćwiczeń laboratoryjnych

KRYTERIA OCENY

EFEKT KSZTALCENIA 1	
NA OCENĘ 3.0	Student zna pojęcia: klasa, obiekt, metoda, konstruktor.
NA OCENĘ 3.5	Student potrafi omówić paradygmaty programowania obiektowego: abstrakcja, enkapsulacja, dziedziczenie, polimorfizm

NA OCENĘ 4.0	Student potrafi omówić paradygmaty programowania obiektowego: abstrakcja, enkapsulacja, dziedziczenie, polimorfizm. Zna zasady tworzenia klas i metod abstrakcyjnych oraz interfejsów
NA OCENĘ 4.5	Student potrafi stosować paradygmaty programowania obiektowego do modelowania znanych problemów
NA OCENĘ 5.0	Student potrafi stosować paradygmaty programowania obiektowego do modelowania dowolnego problemu.
EFEKT KSZTAŁCENIA 2	
NA OCENĘ 3.0	Student potrafi identyfikować klasy w zadanym problemie.
NA OCENĘ 3.5	Student potrafi tworzyć proste klasy w zadanym problemie.
NA OCENĘ 4.0	Student potrafi tworzyć hierarchie klas w zadanym problemie.
NA OCENĘ 4.5	Student potrafi stosować polimorfizm w projekcie programu obiektowego.
NA OCENĘ 5.0	Student potrafi stosować zasady wielokrotnego wykorzystania kodu, polimorfizm w projekcie programu obiektowego.
EFEKT KSZTAŁCENIA 3	
NA OCENĘ 2.0	Nieznajomość zasad specyfikacji klas języka C++ oraz języka Python
NA OCENĘ 3.0	Student zna podstawowe typy danych i instrukcje w językach C++ oraz Python. Potrafi tworzyć specyfikacje klas i obiekty
NA OCENĘ 3.5	Student potrafi oddzielać interfejs od implementacji, stosować dziedziczenie i polimorfizm.
NA OCENĘ 4.0	Student potrafi oddzielać interfejs od implementacji, stosować dziedziczenie i polimorfizm, klasy i metody abstrakcyjne, statyczne, interfejsy.
NA OCENĘ 4.5	Potrafi obsługiwać wyjątki.
NA OCENĘ 5.0	Student zna mechanizm wyjątków. Potrafi stosować asercje.
EFEKT KSZTAŁCENIA 4	
NA OCENĘ 3.0	Student potrafi utworzyć i przetestować dowolną klasę.
NA OCENĘ 3.5	Student potrafi tworzyć diagramy klas w zadanym problemie.
NA OCENĘ 4.0	Student potrafi utworzyć kilka komunikujących się klas z rozdzieleniem interfejsu od implementacji.
NA OCENĘ 4.5	Student potrafi zaimplementować program wykorzystujący dziedziczenie
NA OCENĘ 5.0	Student potrafi zaimplementować program wykorzystujący dziedziczenie i polimorfizm
EFEKT KSZTAŁCENIA 5	

NA OCENĘ 3.0	Student potrafi tworzyć tablice oraz proste kolekcje obiektów
NA OCENĘ 3.5	Student potrafi tworzyć tablice i wykorzystywać proste kolekcje obiektów
NA OCENĘ 4.0	Student potrafi dobrać odpowiednia kolekcje do zadanego problemu
NA OCENĘ 4.5	Student potrafi tworzyć własne kolekcje.
NA OCENĘ 5.0	Student potrafi tworzyć własne złożone kolekcje.

10 MACIERZ REALIZACJI PRZEDMIOTU

EFEKT KSZTAŁCENIA	ODNIESIENIE DANEGO EFEKTU DO SZCZEGÓŁOWYCH EFEKTÓW ZDEFINIOWANYCH DLA PROGRAMU	CELE PRZEDMIOTU	TREŚCI PROGRAMOWE	NARZĘDZIA DYDAKTYCZNE	SPOSOBY OCENY
EK1	K_U01 K_U03 K_U10	Cel 1	W1 K1 K2	N1 N2 N3 N4 N5	F1 F2 P1 P2
EK2	K_U01 K_U03 K_U10	Cel 2	W2 W3 W4 W5 W6 W7 W8 W9 W10 K2 K3 K4 K5 K6 K7 K8 K9 K10 K11	N1 N2 N3 N4 N5	F1 F2 P1 P2
EK3	K_U01 K_U03 K_U10	Cel 2	W2 W3 W4 W5 W6 W7 W8 W9 W10 K1 K2 K3 K4 K5 K6 K7 K8 K9 K10 K11	N1 N2 N3 N4 N5	F1 F2 P1 P2
EK4	K_U01 K_U03 K_U10	Cel 2 Cel 3	W3 W4 W5 W6 W7 W8 W9 W10 K1 K3 K4 K5 K6 K7 K8 K9 K10 K11	N1 N2 N3 N4 N5	F1 F2 P1 P2
EK5	K_U01 K_U03 K_U10	Cel 3	W8 W9 W10 K9 K10 K11	N1 N2 N3 N4 N5	F1 F2 P1 P2

11 WYKAZ LITERATURY

LITERATURA PODSTAWOWA

- [1] | Jerzy Grębosz — *Opus magnum C++11*, , 2017,
[2] | Eric Matthes — *Python. Instrukcje dla programisty*, , 2021,

12 INFORMACJE O NAUCZYCIELACH AKADEMICKICH

OSOBA ODPOWIEDZIALNA ZA KARTĘ

mgr inż. Grzegorz Nowakowski (kontakt: gnowakowski@pk.edu.pl)

OSOBY PROWADZĄCE PRZEDMIOT

1 mgr inż. Grzegorz Nowakowski (kontakt: gnowakowski@pk.edu.pl)

13 ZATWIERDZENIE KARTY PRZEDMIOTU DO REALIZACJI

(miejscowość, data)

(odpowiedzialny za przedmiot)

(dziekan)

PRZYJMUJĘ DO REALIZACJI (data i podpisy osób prowadzących przedmiot)

.....