

WYDZIAŁ ELEKTRONIKI**KARTA PRZEDMIOTU**

Nazwa przedmiotu w języku polskim: **Programowanie obiektowe**
 Nazwa przedmiotu w języku angielskim: **Object Oriented Programming**
 Kierunek studiów (jeśli dotyczy): **Electronic and Computer Engineering**
 Specjalność (jeśli dotyczy):
 Poziom i forma studiów: **I stopień/ stacjonarna**
 Rodzaj przedmiotu: **obowiązkowy**
 Kod przedmiotu: **ECEA17004**
 Grupa kursów: **TAK**

	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium
Liczba godzin zajęć zorganizowanych w Uczelni (ZZU)	30		30		
Liczba godzin całkowitego nakładu pracy studenta (CNPS)	90		90		
Forma zaliczenia	Zaliczenie na ocenę		Zaliczenie na ocenę		
Dla grupy kursów zaznaczyć kurs końcowy (X)	X				
Liczba punktów ECTS	5				
w tym liczba punktów odpowiadająca zajęciom o charakterze praktycznym (P)	-		2		
w tym liczba punktów ECTS odpowiadająca zajęciom wymagającym bezpośredniego udziału nauczycieli lub innych osób prowadzących zajęcia (BU)	1		2		

WYMAGANIA WSTĘPNE W ZAKRESIE WIEDZY, UMIEJĘTNOŚCI I KOMPETENCJI SPOŁECZNYCH

1. K1ECE_W07, K1ECE_U07

CELE PRZEDMIOTU

C1. Zna podstawy inżynierii i metodologii programowania obiektowego
 C2. Umie samodzielnie tworzyć programy zorientowane obiektowo

PRZEDMIOTOWE EFEKTY UCZENIA SIĘ

Z zakresu wiedzy:

PEU_W01	Zna filozofię podejścia obiektowego
PEU_W02	Zna podejście obiektowe jako sposób pojmowania otaczającej rzeczywistości
PEU_W03	Zna podstawy zunifikowanego języka modelowania (UML)
PEU_W04	Zna podstawy inżynierii i metodologii programowania obiektowego
PEU_W05	Zna podstawowe narzędzia obiektowo zorientowanego języka programowania na przykładzie języka C++

Z zakresu umiejętności:

PEU_U01	Potrafi samodzielnie formułować i używać technologii budowy programów obliczeniowych zorientowanych obiektowo
PEU_U02	Potrafi wykonywać i tworzyć fragmenty kodu pozwalające na aktywowanie konstruktorów i destruktorów zarówno w klasach bazowych jak i pochodnych
PEU_U03	Potrafi wykonywać i tworzyć fragmenty kodu zawierające samodzielnie opracowane funkcje wirtualne i operatory przeciążone

TREŚCI PROGRAMOWE

Forma zajęć – wykład		Liczba godzin
Wy1	Wprowadzenie. Omówienie idei podejścia obiektowego	2
Wy2	Prezentacja typowych zastosowań podejścia obiektowego (np. zarządzanie projektami) i najnowszych języków programowania obiektowego	2
Wy3	Obiektowy język programowania C++. Główne koncepcje języka C++. Konstruktory i destruktory.	2
Wy4	Gadżety języka C++. Argumenty domniemane, referencje, deklaratory złożone, modyfikatory, etc. Konstruktor kopiujący i operator przypisania.	2
Wy5	Porównanie obiektowo zorientowanych języków programowania: C++, C# i Java. Platforma programistyczna .NET.	2
Wy6	Obiektowy język programowania Java. Główne koncepcje języka Java, pakiety i implementacje.	2
Wy7	Obiektowy język programowania C#. Główne koncepcje języka C#, interfejsy i odświeżanie.	2
Wy8	Paradygmaty podejścia obiektowego. Hermetyzacja i dziedziczenie. Funkcje wirtualne i klasy abstrakcyjne.	2
Wy9	Budowanie prostej klasy. Hermetyzacja klasy. Pola i funkcje statyczne i niestyczne. Przykład przeciążenia operatora jako metody i operatora jako funkcji globalnej. Przeciążanie operatorów w C++ i C#	2
Wy10	Dziedziczenie i klasy pochodne. Dziedziczenie wielobazowe w C++ i interfejsy w C# i w Javie.	2
Wy11	Język C#. Klasy, wyrażenia i operatory.	2
Wy12	Dziedziczenie, interfejsy, iteratory, obsługa wyjątków, procesy i wątki.	2
Wy13	Elementy zunifikowanego języka modelowania (UML) – diagramy klas, przykłady, przypadki użycia.	4
Wy14	Repetitorium	2
	Suma godzin	30

Forma zajęć – laboratorium

Forma zajęć – laboratorium		Liczba godzin
La1,2	Zapoznanie się ze środowiskiem programistycznym. Realizacja prostego programu z użyciem podejścia strukturalnego	4

La3-6	Realizacja wskazanego przez prowadzącego prostego programu w C++ z wykorzystaniem filozofii podejścia obiektowego	8
La7-9	Indywidualny program w języku C++ uzgodniony z prowadzącym	6
La10-12	Realizacja wskazanego przez prowadzącego prostego programu w C# lub w języku Java	6
La13-15	Indywidualny program w języku C# lub Java uzgodniony z prowadzącym	6
	Suma godzin	30

STOSOWANE NARZĘDZIA DYDAKTYCZNE	
N1. Rzutnik, tablica	
N2. Stanowisko komputerowe, środowisko programistyczne IDE, MS Visual Studio, pakiet aplikacji biurowych	

OCENA OSIĄGNIĘCIA PRZEDMIOTOWYCH EFEKTÓW UCZENIA SIĘ

Oceny (F – formująca (w trakcie semestru), P – podsumowująca (na koniec semestru))	Numer efektu uczenia się	Sposób oceny osiągnięcia efektu uczenia się
F1	PEU_W01 – PEU_W05	Kolokwium zaliczeniowe
F2	PEU_U01 – PEU_U03	Prezentacja aplikacji
$P = 0.6 * F1 + 0.4 * F2$ (pod warunkiem zaliczenia wszystkich form)		

LITERATURA PODSTAWOWA I UZUPEŁNIAJĄCA
<u>LITERATURA PODSTAWOWA:</u> [1] Stroustrup B., The C++ programming language, NJ, Addison-Wesley, 2013. [2] Sahay S., Object oriented programming with C++, 2nd edition, New Delhi : Oxford University Press, 2012. [3] Eckel, B., Thinking in Java, Upper Saddle River: Prentice Hall, 2006 [4] Hejlsberg A., Torgersen M., Wiltamuth S., Golde P., The C# Programming Language (3rd Edition), Microsoft .NET Development Series [5] Malik. D. S., Introduction to C++ programming, Boston, MA: Course Technology, Cengage Learning, 2009. [6] Actual documentation for C++, C#, Java <u>LITERATURA UZUPEŁNIAJĄCA:</u> [1] Kubik T., Kruczkiewicz Z., UML and service description languages: information systems modelling, Wrocław University of Technology, PRINTPAP, 2011. [2] Martin J., Odell J.J., Podstawy metod obiektowych, WNT, 1997
OPIEKUN PRZEDMIOTU (IMIE, NAZWISKO, ADRES E-MAIL)
Dr inż. Marcin Markowski, marcin.markowski@pwr.edu.pl