

Programowanie obiektowe

Sylabus zajęć

Informacje podstawowe

Kierunek studiów Technologie informatyczne		Cykl dydaktyczny 2024/25
Specjalność -		Kod zajęć 17TINS.32K.00238.24
Jednostka organizacyjna Nadnotecki Instytut UAM w Pile		Języki wykładowe polski
Poziom studiów studia inżynierskie pierwszego stopnia		Obligatoryjność Obowiązkowy
Forma studiów studia stacjonarne		Blok zajęciowy Przedmioty kierunkowe
Profil studiów profil praktyczny		
Koordynator zajęć	Arkadiusz Buchelt	
Prowadzący zajęcia	Arkadiusz Buchelt	
Okres Semestr 2	Forma zajęć / liczba godzin / forma zaliczenia • Laboratorium: 30, Zaliczenie z oceną • Konwersatorium: 30, Egzamin	Liczba punktów ECTS 6

Cele kształcenia dla zajęć

Kod	Cel
C1	Wykład prezentuje podstawowe pojęcia i techniki programowania obiektowego na przykładzie języków C# i JavaScript.

Wymagania wstępne

- Umiejętność programowania w języku strukturalnym takim jak C, Python
- Znajomość struktur danych takich jak tablica, lista, itd.

3. Nie wymagana jest znajomość języka C# i JavaScript, będą one wprowadzane na zajęciach

Efekty uczenia się dla zajęć

Kod	Efekty uczenia się dla zajęć w zakresie	Efekty uczenia się dla kierunku	Metody weryfikacji osiągnięcia efektów uczenia się dla zajęć
Wiedzy - Student/ka:			
W1	Zna podstawowe pojęcia i techniki obiektowego paradygmatu programowania.	TIN_K3_W07_inz	Egzamin pisemny
W2	Zna zasadę enkapsulacji, umie ją zastosować i rozumie jej znaczenie w tworzeniu programowania.	TIN_K3_W07_inz	Egzamin pisemny, Raport
W3	Rozumie pojęcie polimorfizmu i umie wykorzystać techniki programowania polimorficznego w praktyce.	TIN_K3_W07_inz	Egzamin pisemny, Projekt, Raport
W4	Zna mechanizm dziedziczenia, potrafi definiować hierarchie klas.	TIN_K3_W07_inz	Projekt, Raport
Umiejętności - Student/ka:			
U1	Potrafi czytać i analizować kod obiektowy napisany w języku C# i JavaScript	TIN_K3_U21_inz	Egzamin pisemny, Projekt
U2	Potrafi korzystać z klas generycznych i samodzielnie je definiować (C#)	TIN_K3_U21_inz	Egzamin pisemny, Projekt
U3	Zna mechanizmy związane z obsługą wyjątków i potrafi je wykorzystać we własnych programach.	TIN_K3_U21_inz	Egzamin pisemny, Projekt, Raport
U4	Potrafi wykonać niewielki projekt programistyczny w metodyce obiektowej.	TIN_K3_U23_inz	Projekt, Raport

Treści programowe dla zajęć

Lp.	Treści programowe dla zajęć	Efekty uczenia się dla zajęć	Formy zajęć
1.	Wprowadzenie do języka C# i Java Script	W1, U1	Konwersatorium, Laboratorium
2.	1. Obsługa błędów 2. C# Obiekt vs Klasa 3. Statyczność 4. Generyczność 5. Enum 6. Modyfikatory dostępu: publiczny a prywatny	W2, U1, U2, U3	Laboratorium
3.	1. Dziedziczenie a kompozycja 2. Modyfikator: chroniony	W2, U2	Laboratorium
4.	1. Interfejs 2. Funkcja a Akcja 3. Lamba i Linq	W3, W4, U4	Laboratorium
5.	1. Akronim SOLID 2. Wzorce projektowe	U4	Laboratorium

Informacje dodatkowe

Forma zajęć	Metody i formy prowadzenia zajęć
Laboratorium	Uczenie problemowe (Problem-based learning)
Konwersatorium	Wykład konwersatoryjny

Forma zajęć	Warunki zaliczenia zajęć
Laboratorium	W zależności od liczby zdobytych punktów: od 90% - bdb od 80% - db+ od 70% - db od 60% - dst+ od 50% - dst poniżej 50% - ndst
Konwersatorium	W zależności od liczby zdobytych punktów: od 90% - bdb od 80% - db+ od 70% - db od 60% - dst+ od 50% - dst poniżej 50% - ndst

Literatura

Obowiązkowa

1. „Hands-On Object-Oriented Programming with C#” Raihan Taher

Dodatkowa

1. "Analiza i projektowanie obiektowe. Rusz głową!", Brett D. McLaughlin, Gary Pollice, David West

Nakład pracy studenta i punkty ECTS

Rodzaje zajęć studenta	Średnia liczba godzin* przeznaczonych na zrealizowane rodzaje zajęć
Laboratorium	30
Konwersatorium	30
Przygotowanie projektu	50
Przygotowanie do egzaminu	40
Przygotowanie do zajęć	30
Łączny nakład pracy studenta	Liczba godzin 180
Liczba punktów ECTS	ECTS 6

* godzina (lekcyjna) oznacza 45 minut

Efekty uczenia się dla kierunku

Kod	Treść
TIN_K3_U21_inz	Absolwent/ka potrafi projektować oprogramowanie zgodnie z metodyką obiektową
TIN_K3_U23_inz	Absolwent/ka potrafi - zgodnie z zadaną specyfikacją - zaprojektować oraz zrealizować prosty system informatyczny, używając właściwych metod, technik i narzędzi
TIN_K3_W07_inz	Absolwent/ka zna i rozumie paradygmaty programowania i języki programowania (imperatywny, obiektowy, funkcyjny, logiczny, skryptowy, maszyna wirtualna, podstawy translacji, deklaracje i typy, odświeżanie, mechanizmy abstrakcji)