

KARTA PRZEDMIOTU (SYLABUS)

Opis przedmiotu

Kod przedmiotu		Nazwa przedmiotu	PROGRAMOWANIE OBIEKTOWE	
I/O/1(i)/NST/B1-15			Object-oriented programming	
Język wykładowy		język polski		
Rok akademicki		2020/2021		
Kierunek		Informatyka		
w zakresie				
Poziom studiów		studia pierwszego stopnia		
Profil studiów		ogólnoakademicki		
Forma studiów		studia niestacjonarne		
Semestr / semestry		drugi		
Przynależność do grupy zajęć		B 1. Grupa zajęć kierunkowych - obowiązkowych		
Status przedmiotu		obowiązkowy		
Formy realizacji zajęć dydaktycznych, wymiar, punkty ECTS		Forma zajęć	Liczba godzin zajęć dydaktycznych	Liczba punktów ECTS
		Wykład	20 [h]	5 ECTS
		Ćwiczenia laboratoryjne	20 [h]	
Powiązanie przedmiotu	z profilem studiów	związany z prowadzoną działalnością naukową w dyscyplinach, do których przyporządkowany jest kierunek studiów		3 ECTS
	z uprawnieniami	służy do zdobywania przez studenta kompetencji inżynierskich		5 ECTS
	z dyscypliną	informatyka techniczna i telekomunikacja informatyka		3 ECTS 2 ECTS
Forma nauczania		tradycyjna - zajęcia zorganizowane w Uczelni i/lub zajęcia z wykorzystaniem metod i technik kształcenia na odległość (max. 0,8 ECTS)		
Wymagania wstępne		Znajomość Podstaw programowania		
Jednostka prowadząca		Katedra Informatyki		
Koordynator		dr Artur Bartoszewski		
Adres strony internetowej pjo		www.wteii.uniwersytetradom.pl		
Adres e-mail, telefon koordynatora		artur.bartoszewski@uthrad.pl		

EFEKTY UCZENIA SIĘ, TREŚCI PROGRAMOWE, REALIZACJA ZAJĘĆ DYDAKTYCZNYCH, WERYFIKACJA EFEKTÓW UCZENIA SIĘ

Cel kształcenia:	Poznanie i opanowanie narzędzi i metod wytwarzania oprogramowania użytecznych w wykonywaniu zawodu informatyka, a w szczególności programowania obiektowego w języku C++,
Treści programowe:	<p>Wykład</p> <ul style="list-style-type: none"> • Dynamiczne alokowanie struktur (struct) [1h] W1, W2 • Podstawowe pojęcia programowania obiektowego, klasy i składniki klas; Dostęp; Enkapsulacja; klasa, a obiekt; [2h] W1, W2 • Konstruktor i destruktor klasy; Wskaźnik this, Przeciążanie metod [2h] • Przekazywanie obiektów do funkcji [2h] W1, W2 • Składniki statyczne klas [1h] W1, W2 • Dziedziczenie [3h] W1, W2 • Funkcje zaprzyjaźnione [1h] W1, W2 • Konstruktor kopiujący; Płytkie i głębokie kopiowanie obiektów [2h] • Przeciążanie operatorów [2h] W1, W2 • Polimorfizm i klasy wirtualne [2h] W1, W2 • Dynamiczne struktury danych; Listy jednokierunkowe [1h] W1, W2 • Dynamiczne struktury danych; Listy dwukierunkowe [1h] W1, W2 <p>Ćwiczenia laboratoryjne</p> <ul style="list-style-type: none"> • Dynamiczne alokowanie struktur (struct), tablica wskaźników do struktur [1h] U1, U2 • Pojęcia programowania obiektowego, klasy i składniki klas; Dostęp; Enkapsulacja [2h] U1, U2 • Konstruktor i destruktor; Wskaźnik this, Przeciążanie metod, Zadanie praktyczne (np.: Generator liczb pseudolosowych wzorowany na dostępnym w C#) [2h] • Przekazywanie obiektów do funkcji [1h] U1 • Składniki statyczne klas, Funkcje zaprzyjaźnione; Przykład praktyczny (np. klasa która zapisująca do pliku tekstowego informacje o czasie utworzenia i usunięcia obiektów) [2h] U1 • Przykład praktyczny (np. klasa realizująca kolejkę LIFO) [2h] U1 • Dziedziczenie [2h] U1 • Konstruktor kopiujący; Płytkie i głębokie kopiowanie obiektów [2h] U1 • Przeciążanie operatorów, Przykład praktyczny (np. klasa umożliwiająca wykonywanie działań na wektorach) [2h] U1 U2 • Polimorfizm i klasy wirtualne [2h] U1 • Dynamiczne struktury danych; Listy jednokierunkowe; Listy dwukierunkowe [1h] U2 • Kolokwium zaliczeniowe [1h]
Metody dydaktyczne (kształcenia):	<p>Metody podające - wykład informacyjny – W1, W2</p> <p>Metody programowane z wykorzystaniem komputera – ćwiczenia laboratoryjne- U1, U2, U3, K1</p> <p>Wszystkie zastosowane metody umożliwiają rozpoznawanie i zaspokajanie indywidualnych potrzeb studentów, w tym studentów niepełnosprawnych oraz indywidualizację toku studiów.</p>
Rygor zaliczenia, kryteria oceny osiągniętych efektów uczenia się, sposób obliczania oceny końcowej:	<p>Warunkiem zaliczenia przedmiotu jest osiągnięcie wszystkich wymaganych efektów kształcenia określonych dla danego przedmiotu.</p> <p>Wykład: Ocena końcowa z egzaminu</p> <p>Ćwiczenia laboratoryjne: Kolokwium praktyczne (przy komputerze)</p>

Efekty uczenia się dla przedmiotu w odniesieniu do efektów kierunkowych i formy zajęć				Metody weryfikacji efektów uczenia się	
Numer efektu uczenia się	Opis efektów uczenia się dla przedmiotu (PEU) Student, który zaliczył przedmiot (W) zna i rozumie/ (U) potrafi /(K) jest gotów do:	Kierunkowy efekt uczenia się (KEU)	Forma zajęć	Forma weryfikacji (zaliczeń)	Metody sprawdzania i oceny
W1	Ma podstawową wiedzę na temat programowania obiektowego paradygmatu programowania	K_WG06 K_WG14 K_WG08	wykład, ćwiczenia laboratoryjne	egzamin, zaliczenie na ocenę	egzamin, kolokwium
W2	Zna składnię i polecenia języka C++	K_WG06	wykład	egzamin, zaliczenie na ocenę	egzamin, kolokwium
U1	Posiada umiejętność projektowania i implementacji programów komputerowych z wykorzystaniem paradygmatu obiektowego	K_WG06 K_WG14 K_WG08	wykład	egzamin, zaliczenie na ocenę	egzamin, kolokwium

U2	Umie dobrać właściwe struktury danych oraz konstrukcje algorytmiczne do rozwiązywanego problemu oraz umie zaprogramować algorytm w formie obiektowej	K_UW05 K_UW12	wykład, ćwiczenia laboratoryjne	egzamin, zaliczenie na ocenę	egzamin, kolokwium
K1	Potrafi zaplanować pracę pod kątem zakładanych rezultatów, określić priorytetowe zadania w oparciu o zasady skutecznego działania.	K_KO05	ćwiczenia laboratoryjne	zaliczenie na ocenę	kolokwium
Stopień osiągnięcia kierunkowych efektów uczenia się: K_WG06 +++; K_WG14 ++; K_WG08 +; K_UW05 +++; K_UW12 ++; K_KO05 +++					

Literatura podstawowa, literatura uzupełniająca, pomoce naukowe

Literatura podstawowa:

1. Grębosz J.: Symfonia C++, Programowanie w języku C++ orientowane obiektowo, Wydawnictwo Edition 2000.
2. Eckel B.: Thinking in C++. Edycja polska, Helion, Gliwice 2002.
3. Walczak K.: Nauka programowania obiektowego w języku C++, W & W Warszawa 2002.
4. Chomicz P.: Programowanie w języku C i C++. PLI 1992.
5. Stroustrup B.: Język C++. Wydawnictwa Naukowo-Techniczne 1995.

Literatura uzupełniająca:

1. Grębosz J.: Pasja C++, Wydawnictwo Edition 2000.
2. Meyers S.: język C++ bardziej efektywnie, Wydawnictwo Naukowo Techniczne, Warszawa 1996.

Autorskie materiały dydaktyczne zamieszczone na stronie www.bartoszewski.uthrad.pl

Nakład pracy studenta potrzebny do osiągnięcia zakładanych efektów uczenia się – bilans punktów ECTS

Udział w zajęciach, aktywność	Obciążenie studenta [h]		
	Inne godz. kontaktowe (IGK)	Zajęcia bez nauczyciela-praca własna studenta (ZBN)	Zajęcia dydaktyczne
Udział w wykładach	X	X	20 [h]
Samodzielne studiowanie tematyki wykładów	X	20 [h]	X
Udział w ćwiczeniach laboratoryjnych	X	X	20 [h]
Samodzielne przygotowanie się do ćwiczeń	X	250 [h]	X
Udział w konsultacjach	8 [h]	X	X
Przygotowanie do egzaminu	X	30 [h]	X
Udział w egzaminie	2 [h]	X	X
Sumaryczne obciążenie pracą studenta	10 [h]/ 0,4 ECTS	75 [h]/3,0 ECTS	40 [h]/ 1,6 ECTS
Punkty ECTS za przedmiot	5 ECTS		

Informacje dodatkowe, uwagi