

KARTA PRZEDMIOTU (SYLABUS)**Opis przedmiotu**

Kod przedmiotu		Nazwa przedmiotu	PODSTAWY PROGRAMOWANIA	
I/O/1(i)/NST/B1-1			BASICS 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		pierwszy		
Przynależność do grupy zajęć		B1. Grupa zajęć kierunkowych - obowiązkowych		
Status przedmiotu		obowiązkowy		
Formy realizacji zajęć dydaktycznych, wymiar, punkty ECTS		Forma zajęć	Liczba godzin zajęć dydaktycznych	Liczba punktów ECTS
		Wykład	25 [h]	6 ECTS
		Ćwiczenia laboratoryjne	30 [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		6 ECTS
	z dyscypliną	informatyka techniczna i telekomunikacja informatyka		4 ECTS 2 ECTS
Forma nauczania		tradycyjna - zajęcia zorganizowane w Uczelni i/lub zajęcia z wykorzystaniem metod i technik kształcenia na odległość (max. 1 ECTS)		
Wymagania wstępne				
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 możliwie szerokiego zakresu zagadnień podstawowych związanych z programowaniem komputerów, strukturami danych i algorytmami oraz związanych z nimi technik tworzenia programów. Wykłady oraz ćwiczenia o charakterze laboratoryjnym mają na celu opanowanie umiejętności formułowania problemu, tworzenia algorytmów i implementacji w języku C++
Treści programowe:	<p>Wykład</p> <ul style="list-style-type: none"> Języki programowania i ich klasyfikacja. Środowiska programistyczne, kompilacja i interpretacja. Uruchamianie i testowanie programu. Problem poprawności i złożoności programu. [1h] W2 Programowanie proceduralne: struktura programu, funkcje, moduły i biblioteki. [1h] W2 Pojęcia podstawowe: problem, algorytm, program, metody zapisu algorytmów. [2h] W1 Podstawowe konstrukcje języka C/C++: identyfikatory, typy standardowe, typy wyliczeniowe, zmienne i stałe, wyrażenia, funkcje standardowe. Instrukcje warunkowe. [2h] W2, Pojęcie iteracji i rekurencji, pętle programowe [2h] W1 Praca z tablicami jedno i wielowymiarowymi. Algorytmy sortowania. [3h] W2, Funkcje, parametry formalne i aktualne, przekazywanie parametrów do funkcji. Zasięg widoczności zmiennych (lokalny i globalny), przeciążanie funkcji. [4h] W2 Przykłady znanych algorytmów: (np.: algorytm Euklidesa, rozwiązywanie równań kwadratowych itp.). Przykłady wykorzystania poznanych struktur programistycznych do rozwiązywania problemów. [2h] W1 Łańcuch znaków i klasa String [2h] W2 Obsługa plików tekstowych [2h] W2 Złożone typy danych (struktury, unie, pola bitowe) [2h] W2 Wskaźniki i zmienne dynamiczne: [2h] W2 <p>Ćwiczenia laboratoryjne</p> <p>Studenci poznają wybrane środowiska programistyczne dla języków C i C++ [1h] U3</p> <p>Studenci wykonują programy służące do rozwiązywania problemów dotyczących następujących zagadnień:</p> <ul style="list-style-type: none"> rozgałęzienia w programie, instrukcje wyboru oraz pętle programowe, iteracyjne rozwiązywanie problemów numerycznych (algorytmy działające na ciągach liczbowych itp.) [6h] U1 operacje na tablicach, tablicowanie funkcji, działania na znakach i łańcuchach znaków, wyznaczanie min/max n elementowego ciągu liczbowego, sortowania (sortowanie bąbelkowe, łączenie dwóch ciągów uporządkowanych itp.) [6h] U1, U2 podział programu na funkcje, przekazywanie parametrów do funkcji, przeciążanie funkcji [2h] U1 U2 złożone typy danych (łańcuchy znaków i struktury) [4h] U2, U3 operacje na plikach tekstowych [3h] U1, U2 rekurencja (silnia, potęga, itp.) [3h] U1, U2 wskaźniki i zmienne dynamiczne [3h] U2, U3 kolokwium zaliczeniowe [2h]
Metody dydaktyczne (kształcenia):	<p>Metody podające - wykład informacyjny – W1, W2</p> <p>Metody programowane z wykorzystaniem komputera – ćwiczenia laboratoryjne- U1, U2, U3, K1, K2</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:</p> <p>Ocena końcowa z egzaminu pisemnego</p> <p>Ćwiczenia laboratoryjne:</p> <p>Sprawdzian praktyczny</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 komputerów, systemów operacyjnych, struktur danych i algorytmów oraz związanych z nimi technik tworzenia i uruchamiania programów	K_WG06 K_WG14 K_WG08	wykład, ćwiczenia laboratoryjne	egzamin, zaliczenie na ocenę	egzamin, kolokwium
W2	Zna składnię i polecenia języka C oraz elementy składni języka C++ (z wyłączeniem programowania obiektowego).	K_WG06	wykład	egzamin, zaliczenie na ocenę	egzamin, kolokwium
U1	Posiada umiejętność budowy, zapisu oraz implementacji algorytmów w językach programowania C i C++	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 języku wysokiego poziomu	K_UW05 K_UW12	wykład, ćwiczenia laboratoryjne	egzamin, zaliczenie na ocenę	egzamin, kolokwium
U3	Ma umiejętność formułowania problemu w języku C oraz tworzenia i uruchamiania programów w wybranym środowisku programistycznym	K_UW04 K_UW12	wykład, ćwiczenia laboratoryjne	egzamin, zaliczenie na ocenę	egzamin, kolokwium
K1	Jest gotów poszerzać swoją wiedzę i doskonalić umiejętności potrzebne w wykonywaniu zawodu programisty	K_KK02	wykład	egzamin	egzamin
K2	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_UW04 +++; K_UW05 +++; K_UW12 ++; K_KK02 +++; K_KO05 +++					

Literatura podstawowa, literatura uzupełniająca, pomoce naukowe	
Literatura podstawowa: <ol style="list-style-type: none"> Grębosz J.: Symfonia C++, Programowanie w języku C++ orientowane obiektowo, Wydawnictwo Edition 2000. Jakubczyk K.: Turbo Pascal i Borland C++. Przykłady, Helion, Gliwice 2002, Chomicz P.: Programowanie w języku C i C++. PLI 1992. Stroustrup B.: Język C++. Wydawnictwa Naukowo-Techniczne 1995. Kerninghan B. W., Ritchie D. M.: język ANSI C, Wydawnictwo Naukowo Techniczne, Warszawa 1994. Literatura uzupełniająca: <ol style="list-style-type: none"> Sysło M.M.: Piramidy, szyszki i inne konstrukcje algorytmiczne, WSiP, Warszawa 1998. Wirth N.: <i>Algorytmy + struktury danych = programy</i>, WNT, Warszawa 2004. Simon L. D., Drozdek A.: <i>Struktury danych w języku C</i>, WNT, 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 <i>wykładach</i>	X	X	25 [h]
Samodzielne studiowanie tematyki <i>wykładów</i>	X	20 [h]	X
Udział w <i>ćwiczeniach laboratoryjnych</i>	X	X	30 [h]
Samodzielne przygotowanie się do <i>ćwiczeń</i>	X	30 [h]	X
Udział w konsultacjach	5 [h]	X	X
Przygotowanie do <i>egzaminu</i>	X	30 [h]	X
Udział w <i>egzaminie</i>	5 [h]	X	X
Sumaryczne obciążenie pracą studenta	10 [h] 0,5 ECTS	80[h] 3,5 ECTS	55 [h] 2 ECTS
Punkty ECTS za przedmiot	6 ECTS		

Informacje dodatkowe, uwagi