



KARTA PRZEDMIOTU

Kod przedmiotu	studia stacjonarne:	E-ID2S-25-s2, E-ID2S-16-s3, E-ID2G-13-s2, E-ID2G-15-s3
	studia niestacjonarne:	E-2IZ2S-1004-s3 E-2IZ2G-1009-s3
Nazwa przedmiotu	Programowanie imperatywne, obiektowe i deklaratywne	
Nazwa przedmiotu w języku angielskim	Imperative, Object and Declarative programming	
Obowiązuje od roku akademickiego	2022/23	

USYTUOWANIE PRZEDMIOTU W SYSTEMIE STUDIÓW

Kierunek studiów	Informatyka
Poziom kształcenia	II stopień
Profil studiów	Ogólnoakademicki
Forma i tryb prowadzenia studiów	Studia stacjonarne i niestacjonarne
Zakres	Systemy informacyjne, Grafika komputerowa
Jednostka prowadząca przedmiot	Katedra Informatyki Stosowanej
Koordinator przedmiotu	Prof. dr hab. inż. Volodymyr Ovsyak
Zatwierdził	Dziekan Wydziału Elektrotechniki, Automatyki i Informatyki dr hab. inż. Roman Deniziak, prof. PŚk

OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU

Przynależność do grupy/bloku przedmiotów	Przedmiot specjalnościowy	
Status przedmiotu	Wybieralny	
Język prowadzenia zajęć	Polski	
Usytuowanie w planie studiów - semestr	studia stacjonarne	Semestr II lub Semestr III
	studia niestacjonarne	Semestr II lub Semestr III
Wymagania wstępne	Brak	
Egzamin (TAK/NIE)	Nie	
Liczba punktów ECTS	3	

Forma prowadzenia zajęć		wykład	ćwiczenia	laboratorium	projekt	Inne
Liczba godzin w semestrze	studia stacjonarne:	30			30	
	studia niestacjonarne:	18			18	

EFEKTY UCZENIA SIĘ

Kategoria	Symbol efektu	Efekty uczenia się	Odniesienie do efektów kierunkowych
Wiedza	W01	Student ma podstawową wiedzę języka imperatywnego paradygmatu programowania mikroprocesorów.	INF2_W01
	W02	Student ma podstawową wiedzę języka obiektowego programowania C# i języka markowania XAML Windows Presentation Foundation.	INF2_W02
Umiejętności	U01	Student potrafi stworzyć oprogramowanie systemu informatycznego stosując język imperatywnego paradygmatu programowania.	INF2_U01
	U02	Student potrafi stworzyć oprogramowanie systemu informatycznego wykorzystaniem języka obiektowego programowania C# platformy Microsoft Visual Studio .NET.	INF2_U10
	U03	Student potrafi zaprogramować interfejs użytkownika systemu informatycznego i podstawowe elementy grafiki komputerowej wykorzystaniem języka markowania XAML.	INF2_U01
Kompetencje społeczne	K01	Student zna podstawy składni i semantyki języków Asemblera mikroprocesorów ARM, C# i XAML platformy Microsoft Visual Studio .NET.	INF2_K04

TREŚCI PROGRAMOWE

Forma zajęć	Treści programowe
wykład	<ol style="list-style-type: none"> Paradygmaty programowania. Paradygmaty imperatywnego, obiektowego i deklaratywnego programowania. Model architektury mikroprocesorów ARM. Rejestry, ALU, dekodery instrukcji, magistrale i pamięć. Model programowania. Typy danych. Tryby funkcjonowania mikroprocesora. Rejestry. Rejestry stanu programu. Tabela wektorów. Składnia kodu systemu informatycznego w języku Asemblera. Struktura modułów języka Asemblera. Nazwy rejestrów. Dyrektywy. Makra. Operacje. Literały. Stałe, pole literałów i operacje. Ładowanie stałych do rejestrów. Ładowanie adresów do rejestrów. Operacje logiczne i arytmetyczne. Pętle, skoki i tablice. Pętla while. Pętla for. Pętla do ... while. Znaczniki. Wykonanie warunkowe. Tablice. Składnia języka C#. Identyfikatory i słowa kluczowe. Literały, znaki interpunkcyjne. Komentarze. System typów danych. Typy predefiniowane. Typy własne. Składowe statyczne i składowe instancje. Konwersje typów. Typy wartościowe i typy referencyjne. Zmienne i parametry. Operatory. Metody. Klasy. Operatory logiczne i arytmetyczne. Metody statyczne i dynamiczne, wirtualne i abstrakcyjne. Parametry metod. Klasy częściowy i statyczne. Definicja XAML. Elementy i atrybuty. Przestrzenie nazw. Elementy właściwości. Konwertery typów. Rozszerzenie znaczników. Elementy potomne obiektów. Łączenie XAML z kodem obiektowym. Słowa kluczowe XAML.
projekt	Przedmiotem projektu jest stworzenie kodu systemu informatycznego zadanych graficzno-analitycznych wyrazów algebry algorytmów w oparciu na wykorzystanie nowoczesnego języka obiektowego programowania C#, języka markowania i środowiska Windows Presentation Foundation.

METODY WERYFIKACJI EFEKTÓW UCZENIA SIĘ

Symbol efektu	Metody sprawdzania efektów uczenia się					
	Egzamin ustny	Egzamin pisemny	Kolokwium	Projekt	Sprawozdanie	Inne
W01				X	X	X
W02				X	X	X
U01				X	X	X
U02				X	X	X
U03				X	X	X
K01				X	X	X

FORMA I WARUNKI ZALICZENIA

Forma zajęć	Forma zaliczenia	Warunki zaliczenia
wykład	zaliczenie z oceną	Uzyskanie co najmniej 50% punktów odpowiadających wiedzy i umiejętnościom teoretycznym.
projekt	zaliczenie z oceną	Sprawozdanie i kod działającego systemu.

NAKŁAD PRACY STUDENTA

Bilans punktów ECTS														
Lp.	Rodzaj aktywności	Obciążenie studenta										Jednostka		
		studia stacjonarne					studia niestacjonarne							
		W	C	L	P	S	W	C	L	P	S			
1.	Udział w zajęciach zgodnie z planem studiów	30			30		18			18				h
2.	Inne (konsultacje, egzamin)	2			2		2			2				h
3.	Razem przy bezpośrednim udziale nauczyciela akademickiego	64					40					h		
4.	Liczba punktów ECTS, którą student uzyskuje przy bezpośrednim udziale nauczyciela akademickiego	2,56					1,6					ECTS		
5.	Liczba godzin samodzielnej pracy studenta	11					35					h		
6.	Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy	0,44					1,4					ECTS		
7.	Nakład pracy związany z zajęciami o charakterze praktycznym	30					18					h		
8.	Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym	1,20					0,72					ECTS		
9.	Sumaryczne obciążenie pracą studenta	75					75					h		
10.	Punkty ECTS za moduł <i>1 punkt ECTS=25 godzin obciążenia studenta</i>	3										ECTS		

LITERATURA

1. J.Albachi, B. Albachani. C# 6.0. Leksykon kieszonkowy. – Gliwice: Wydawnictwo “Helion”, 2017.
2. A.Nathan. WPF 4.5. Księga eksperta. – Gliwice: Wydawnictwo “Helion”, 2015.
3. J.Matulewski. NVVM i XAML w Visual Studio 2015. – Gliwice: Wydawnictwo “Helion”, 2016.
4. R.Osherowe. Testy jednostkowe. Świat niezawodowych aplikacji. – Gliwice: Wydawnictwo “Helion”, 2014.
5. J.Matulewski. NVVM i XAML w Visual Studio 2015. – Gliwice: Wydawnictwo “Helion”, 2016.
6. W. Hohl. Asembler dla mikroprocesorów ARM. – Gliwice: Wydawnictwo “Helion”, 2014.
7. A.Kempa. Wprowadzenie do WPF. Tworzenie aplikacji w WPF przy użyciu XAML i C#. – Gliwice: Wydawnictwo “Helion”, 2017.