



KARTA PRZEDMIOTU

Kod przedmiotu	studia stacjonarne:	E-I2G-2009-s1
	studia niestacjonarne:	E-1I22G-1004-s1
Nazwa przedmiotu	Algorytmy grafiki komputerowej	
Nazwa przedmiotu w języku angielskim	Algorithms of computer graphics	
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	Grafika komputerowa
Jednostka prowadząca przedmiot	Katedra Systemów Informatycznych
Koordinator przedmiotu	Dr inż. Grzegorz Łukawski
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	Obowiązkowy	
Język prowadzenia zajęć	Polski	
Usytuowanie w planie studiów - semestr	studia stacjonarne	Semestr I
	studia niestacjonarne	Semestr I
Wymagania wstępne	Algorytmy i struktury danych, Podstawy grafiki komputerowej	
Egzamin (TAK/NIE)	Nie	
Liczba punktów ECTS	4	

Forma prowadzenia zajęć		wykład	ćwiczenia	laboratorium	projekt	Inne
Liczba godzin w semestrze	studia stacjonarne:	30		15	20	
	studia niestacjonarne:	18		9	12	

EFEKTY UCZENIA SIĘ

Kategoria	Symbol efektu	Efekty uczenia się	Odniesienie do efektów kierunkowych
Wiedza	W01	Student ma znajomość metod programowania grafiki 3D z pomocą interfejsu OpenGL oraz programów shaderowych.	INF2_W03, INF2_W04
	W02	Student ma znajomość drzew klasyfikacyjnych i podstaw logiki rozmytej pod kątem zastosowania w grach komputerowych.	INF2_W05
	W03	Student ma znajomość zaawansowanych metod wizualizacji w grafice 3D.	INF2_W03, INF2_W04
	W04	Student ma znajomość algorytmów optymalizacji w grafice komputerowej.	INF2_W03, INF2_W04, INF2_W08
Umiejętności	U01	Student posiada umiejętność tworzenia aplikacji i gier 3D z wykorzystaniem poznanych algorytmów oraz interfejsu OpenGL.	INF2_U01, INF2_U10, INF2_U11
	U02	Student posiada umiejętność wyboru i zastosowania zaawansowanych metod wizualizacji i optymalizacji dla wybranych problemów związanych z grafiką 3D.	INF2_U01
Kompetencje społeczne	K01	Student umie podzielić problem programistyczny na elementy i współpracować w grupie przy jego implementacji.	INF2_K01 INF2_K04

TREŚCI PROGRAMOWE

Forma zajęć	Treści programowe
wykład	<ol style="list-style-type: none"> 1. Podstawy programowania grafiki 3D z pomocą interfejsu OpenGL oraz programów shaderowych. 2. Zaawansowane metody wizualizacji w grafice komputerowej 3D. 3. Metody optymalizacji jakości/szybkości stosowane w grafice komputerowej. 4. Zastosowania drzew klasyfikacyjnych i logiki rozmytej w grach komputerowych.
laboratorium	<ol style="list-style-type: none"> 1. Programowanie grafiki 3D z pomocą interfejsu OpenGL. 2. Problemy optymalizacji w grafice komputerowej.
projekt	Zadanie projektowe polegające na przygotowaniu aplikacji wykorzystującej zaawansowane metody wizualizacji i optymalizacji w grafice 3D.

METODY WERYFIKACJI EFEKTÓW UCZENIA SIĘ

Symbol efektu	Metody sprawdzania efektów uczenia się					
	Egzamin ustny	Egzamin pisemny	Kolokwium	Projekt	Sprawozdanie	Inne
W01			X			
W02			X			
W03			X			
W04			X			
U01			X			X
U02			X			X
K01				X	X	

FORMA I WARUNKI ZALICZENIA

Forma zajęć	Forma zaliczenia	Warunki zaliczenia
wykład	zaliczenie z oceną	Uzyskanie co najmniej 50% punktów z kolokwium końcowego.
laboratorium	zaliczenie z oceną	Uzyskanie co najmniej 50% punktów za wykonane zadania laboratoryjne oraz z kolokwium końcowego, kolokwium w trakcie zajęć lub kolokwium końcowego i kolokwium w trakcie zajęć.
projekt	zaliczenie z oceną	Uzyskanie co najmniej 50% punktów za wykonane zadanie projektowe oraz sprawozdanie

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		15	20		18		9	12		h
2.	Inne (konsultacje, egzamin)	2		2	2		2		2	2		h
3.	Razem przy bezpośrednim udziale nauczyciela akademickiego	71					45					h
4.	Liczba punktów ECTS, którą student uzyskuje przy bezpośrednim udziale nauczyciela akademickiego	2,84					1,8					ECTS
5.	Liczba godzin samodzielnej pracy studenta	29					55					h
6.	Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy	1,16					2,2					ECTS
7.	Nakład pracy związany z zajęciami o charakterze praktycznym	35					21					h
8.	Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym	1,40					0,84					ECTS
9.	Sumaryczne obciążenie pracą studenta	100					100					h
10.	Punkty ECTS za moduł <i>1 punkt ECTS=25 godzin obciążenia studenta</i>	4										ECTS

LITERATURA

1. Jason Gregory: "Game Engine Architecture", A K Peters Ltd., 2009.
2. David M. Bourg, "Fizyka dla programistów gier", Helion 2003.
3. Mark DeLoura, Dante Treglia, Mike Dickheiser: "Perełki programowania gier. Vademecum profesjonalisty", Wydawnictwo Helion.
4. Randi J. Rost: "OpenGL Shading Language (3rd Edition)", Addison-Wesley Professional, 2009.
5. Graham Sellers, Richard S. Wright Jr., Nicholas Haemel: "OpenGL. Księga eksperta. Wydanie VII", Wydawnictwo Helion 2016.