



IV. Opis programu studiów

3. KARTA PRZEDMIOTU

Kod przedmiotu	E-I2G-2009-s1
Nazwa przedmiotu	Algorytmy grafiki komputerowej
Nazwa przedmiotu w języku angielskim	Algorithms of computer graphics
Obowiązuje od roku akademickiego	2019/20

USYTUOWANIE MODUŁU W SYSTEMIE STUDIÓW

Kierunek studiów	
Poziom kształcenia	
Profil studiów	
Forma i tryb prowadzenia studiów	
Zakres	
Jednostka prowadząca przedmiot	
Koordynator przedmiotu	Grzegorz Łukawski
Zatwierdził	Dziekan Wydziału Elektrotechniki Automatyki i Informatyki Dr hab. inż. Antoni Różowicz, prof. PŚk

OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU

Przynależność do grupy/bloku przedmiotów	
Status przedmiotu	
Język prowadzenia zajęć	
Usytuowanie modułu w planie studiów - semestr	
Wymagania wstępne	Algorytmy i struktury danych, Podstawy grafiki komputerowej
Egzamin (TAK/NIE)	
Liczba punktów ECTS	

Forma prowadzenia zajęć	wykład	ćwiczenia	laboratorium	projekt	Inne
Liczba godzin w semestrze	30	0	16	16	0

EFEKTY UCZENIA SIĘ

Kategoria	Symbol efektu	Efekty kształcenia	Odniesienie do efektów kierunkowych
Wiedza	W01	Znajomość algorytmów i struktur danych stosowanych w silnikach gier.	INF2_W03 INF2_W06
	W02	Znajomość drzew klasyfikacyjnych i logiki rozmytej pod kątem zastosowania w grach komputerowych.	INF2_W07
	W03	Znajomość zaawansowanych metod wizualizacji w grafice 3D.	INF2_W03 INF2_W06
	W04	Znajomość algorytmów optymalizacji w grafice komputerowej.	INF2_W03 INF2_W06
Umiejętności	U01	Umiejętność stworzenia podstaw własnego silnika gier z wykorzystaniem poznanych algorytmów.	INF2_U16 INF2_U19 INF2_U20
	U02	Umiejętność wyboru i zastosowania zaawansowanych metod wizualizacji i optymalizacji dla wybranych problemów związanych z grafiką 3D.	INF2_U01 INF2_U06
Kompetencje społeczne	K01	Student umie podzielić problem programistyczny na elementy i współpracować w grupie przy jego implementacji.	INF2_K01 INF2_K02

TREŚCI PROGRAMOWE

Forma zajęć*	Treści programowe
wykład	1-4. Algorytmy i struktury danych stosowane w silnikach gier.
	5-9. Zaawansowane metody wizualizacji w grafice komputerowej 3D.
	10-13. Metody optymalizacji jakości/szybkości stosowane w grafice komputerowej.
	14-15. Zastosowanie drzew klasyfikacyjnych i logiki rozmytej w grach komputerowych.
laboratorium	1-4. Algorytmy i struktury danych w silnikach gier.
	5-8. Problemy optymalizacji w grafice komputerowej.
projekt	1-8. Zadanie projektowe polegające na przygotowaniu aplikacji wykorzystującej zaawansowane metody wizualizacji i optymalizacji w grafice 3D.

*) zostawić tylko realizowane formy zajęć

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	

A.

FORMA I WARUNKI ZALICZENIA

Forma zajęć*	Forma zaliczenia	Warunki zaliczenia
wykład		Uzyskanie co najmniej 50% punktów z kolokwium końcowego.
laboratorium		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		Uzyskanie co najmniej 50% punktów za wykonane zadanie projektowe oraz sprawozdanie.

*) zostawić tylko realizowane formy zajęć

NAKŁAD PRACY STUDENTA

Bilans punktów ECTS							
Lp.	Rodzaj aktywności	Obciążenie studenta					Jednostka
1.	Udział w zajęciach zgodnie z planem studiów	W	C	L	P	S	h
		30		16	16		
3.	Inne (konsultacje, egzamin)*	2		2	2		h
4.	Razem przy bezpośrednim udziale nauczyciela akademickiego	68					h
5.	Liczba punktów ECTS, którą student uzyskuje przy bezpośrednim udziale nauczyciela akademickiego	2,72					ECTS
6.	Liczba godzin samodzielnej pracy studenta	32					h
7.	Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy	1,28					ECTS
8.	Nakład pracy związany z zajęciami o charakterze praktycznym	32					h
9.	Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym	1,88					ECTS
10.	Sumaryczne godzinowe obciążenie pracą studenta	100					h
11.	Punkty ECTS za moduł <i>1 punkt ECTS=25 godzin obciążenia studenta</i>	4					

* wszelkie formy weryfikacji efektów, w tym egzaminy oraz nie więcej niż 2 godziny konsultacji dla każdej formy zajęć

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.