

SYLABUS ZAJ /GRUPY ZAJ

Dane ogólne:

Jednostka organizacyjna:	Katedra Informatyki				
Kierunek studiów:	Informatyka				
Specjalno /Specjalizacja:					
Nazwa zaj / grupy zaj :	Grafika 3D i programowanie kart graficznych II				
Course / group of courses:	3D Graphics and Graphics Cards Programming II				
Forma studiów:	stacjonarne				
Nazwa katalogu:	WP-IN-I-20/21Z				
Nazwa bloku zaj :					
Kod zaj /grupy zaj :	105993	Kod Erasmus:			
Punkty ECTS:	1	Rodzaj zaj :		fakultatywny	
Rok studiów:	2	Semestr:		4	
Rok	Semestr	Forma zaj	Liczba godzin	Forma zaliczenia	ECTS
2	4	P	15	Zaliczenie z ocen	1
Razem			15		1
Koordynator:	J drzej Byrski				
Prowadz cy zaj cia:					
J zyk wykładowy:	semestr: 4 - j zyk polski				

Obja nienia:

Rodzaj zaj : obowi zkowe, do wyboru.

Forma prowadzenia zaj : W - wykład, - wiczenia audytoryjne, L - lektorat, S – seminarium/ zaj cia seminaryjne, P - wiczenia praktyczne (w tym zaj cia wf), M - wiczenia specjalistyczne (medyczne/ kliniczne), LO – wiczenia laboratoryjne, LI - laboratorium informatyczne, ZTI - zaj cia z technologii informacyjnych, P – wiczenia projektowe, ZT – zaj cia terenowe, T - wiczenia terenowe na obozach programowych, SK - samokształcenie (i inne), PR - praktyka zawodowa

Dane merytoryczne

Wymagania wst pne:
Znajomo teoretyczna oraz praktyczna j zyka C oraz C++, znajomo teoretyczna OpenGL oraz CUDA, zaliczenie pozytywnie kursów: Programowanie w C, Programowanie w C++ oraz Grafika 3D i programowanie kart graficznych.

Szczegółowe efekty uczenia si

Lp.	Student, który zaliczył zaj cia zna i rozumie/potrafi/jest gotowy do:	Kod efektu dla kierunku studiów	Sposób weryfikacji efektu uczenia si
1	Umie tworzy zrównoleglony algorytm rozwijajacy dany problem z wykorzystaniem koprocatora GPU.	IN1_U01, IN1_U03, IN1_U07	wypowied ustna
2	Umie programowa renderowanie scen trójwymiarowych przy uyciu biblioteki OpenGL..	IN1_U01, IN1_U07, IN1_U12	wypowied ustna
3	Umie wykorzysta cechy programowalnych potoków przetwarzania grafiki przy tworzeniu oprogramowania dla akceleratorów graficznych.	IN1_U01, IN1_U07, IN1_U12	wypowied ustna
4	Potrafi tworzy zaawansowane oprogramowanie z wykorzystaniem platformy CUDA.	IN1_U01, IN1_U11, IN1_U13, IN1_U03, IN1_U07	wypowied ustna
5	Potrafi samodzielnie rozwija zło one problemy wystujace podczas wytwarzania systemów opartych o trójwymiarowe biblioteki graficzne.	IN1_U01, IN1_U11, IN1_U13, IN1_U03, IN1_U07	wypowied ustna

6	Potrafi wykorzystać zaawansowane i niskopoziomowe funkcje platformy CUDA.	IN1_U01, IN1_U12	wypowiedź ustna
7	Umie optymalizować programy korzystające z platformy CUDA.	IN1_U01, IN1_U12	obserwacja wykonania zadania, wypowiedź ustna
8	Potrafi tworzyć oprogramowanie wykorzystujące bibliotekę OpenGL w procesach wytwarzania grafiki.	IN1_U01, IN1_U13, IN1_U03, IN1_U12	wypowiedź ustna
9	Potrafi wykonać złożony projekt bazujący na technologii OpenGL lub CUDA wraz z jego optymalizacją oraz opracowaniem dokumentacji.	IN1_U11	obserwacja wykonania zadania, wypowiedź ustna
10	jest gotów pracować w zespole programistycznym, komunikować się w obrębie grupy. Ma świadomość odpowiedzialności wobec grupy i z wywiązaniem w terminie swojej części zadania.	IN1_K01	wypowiedź ustna
Stosowane metody osiągnięcia zakładanych efektów uczenia się (metody dydaktyczne)			
metody praktyczne (1. Projekt (metoda projektów) - wykonanie oraz prezentacja (platforma MS Teams) projektu, 2. Objaśnienie (wyjaśnienie, omówienie) - odpowiedź podczas oddawania projektu na platformie MS Teams.)			
Kryteria oceny i weryfikacji efektów uczenia się			
umiejętności: obserwacja wykonania zadania (Obserwacja wykonania zadania podczas pracy zdalnej na platformie MS Teams) ocena wypowiedzi ustnej (Ocena wypowiedzi podczas pracy zdalnej na platformie MS Teams)			
kompetencje społeczne: ocena wypowiedzi ustnej (Ocena wypowiedzi podczas pracy zdalnej na platformie MS Teams)			
Warunki zaliczenia			
zaliczenie ćwiczeń projektowych z oceną na podstawie oddanego projektu jego prezentacji oraz odpowiedzi ustnej (na platformie MS Teams), oceny wystawiane są zgodnie z aktualnym regulaminem studiów PWSZ w Tarnowie.			
Treści programowe (opis skrócony)			
Zrealizowany grupowy projekt pozwoli zapoznać się z: 1. Architekturą oraz programowaniem CUDA. 2. Podstawami programowania biblioteki OpenGL.			
Content of the study programme (short version)			
Completing a group project will allow participants to familiarize themselves with: 1. Architecture and CUDA programming. 2. Basics of OpenGL programming.			
Treści programowe			
			Liczba godzin
Semestr: 4			
Forma zajęć: wiczenia projektowe			
W ramach zajęć student nauczy się programować złożone zagadnienia związane z renderowaniem trójwymiarowym przy użyciu biblioteki OpenGL, szybkim renderowaniem potoków wielokątów w sprzętowych akceleratorach 3D wykorzystując układy odwzorowania przestrzeni 3D, rysowanie figur prostych (prymitywów) i wyświetlanie siatek wielokątów, reprezentacje brył trójwymiarowych, tworzenie materiałów graficznych dla OpenGL oraz wykorzystanie OpenGL. Zapozna się oraz utrwali w ramach realizowanego projektu konwencje nazewnictwa funkcji i stałych bibliotecznych w OpenGL, typy zmiennych, obsługę błędów, podstawowe czynności konfiguracyjne, tworzenie aplikacji OpenGL, rysowanie w OpenGL, definiowanie obiektów rysowanych w OpenGL, definiowanie wierzchołków w układzie współrzędnych, definiowanie parametrów wyświetlania figur, przegląd funkcji bibliotecznych rysujących standardowe figury OpenGL oraz przekształcenia w przestrzeni trójwymiarowej OpenGL. W ramach projektu po wcześniejszym programowaniu procesorów graficznych zapozna się oraz utrwali umiejętność programowania obliczeń w grafice komputerowej wykorzystując technologię CUDA, pozna praktycznie podstawowe oraz zaawansowane programowanie aplikacji korzystających z technologii CUDA, wraz z zagadnieniem optymalizacji. Pozna technologie programowania aplikacji z bezpośrednim wykorzystaniem GPU, technologie tworzenia kerneli w praktyce. Uruchomienie kerneli w aplikacjach bazujących na CUDA. Profile wymiany danych pomiędzy CPU a GPU, rozszerzenia biblioteki CUDA. Praktycznie będzie stosował w ramach realizowanego			15

projektu j zyk nVidia CG, jego składni , dost pne funkcje i profile wymiany danych, importowanie, kompilacj i uruchomienie programów. Po zrealizowaniu projektu b dzie znał platform CUDA (Compute Unified Device Architecture) wraz z jej konfigurowaniem, wersje CUDA API i ich funkcjonalno oraz programowanie niskopoziomowe z u yciem CUDA.	15
Literatura	
Podstawowa	
Jason Sanders, Edward Kandrot, CUDA w przykładach. Wprowadzenie do ogólnego programowania procesorów GPU.	
Richard S. Wright jr, Graham Sellers, OpenGL. Ksi ga eksperta,	
Uzupełniaj ca	
Bharatkumar Sharma, Jack Han, CUDA CookbookEffectiverecipes for parallelprogramming on GPU	
Janusz Ganczarski, OpenGL. Podstawy programowania grafiki 3D.	
Kevin Hawkins, DaveAstle, OpenGL. Programowanie gier.	
Muhammad MobeenaMovania, OpenGL. Receptury dla programisty	
Nicholas Wilt, The Cuda Handbook: A Comprehensive Guide to GPU Programming,	
Tolga Soyata, GPU Parallel Program Development Using CUDA	

Dane jako ciowe

Przyporzkowanie zaj /grup zaj do dyscypliny naukowej/artystycznej		informatyka techniczna i telekomunikacja	
Sposób okre lenia liczby punktów ECTS			
Forma nakładu pracy studenta (udział w zaj ciach, aktywno , przygotowanie sprawozdania, itp.)		Obci enia studenta [w godz.]	
Udział w zaj ciach		15	
Konsultacje z prowadz cym		0	
Udział w egzaminie		0	
Bezpo redni kontakt z nauczycielem - inne		0	
Przygotowanie do laboratorium, wicze , zaj		5	
Przygotowanie do kolokwiiów i egzaminu		5	
Indywidualna praca własna studenta z literatur , wykładami itp.		0	
Inne		0	
Sumaryczne obci enie prac studenta		25	
Liczba punktów ECTS			
Liczba punktów ECTS		1	
Zaj cia wymagaj ce bezpo redniego udziału nauczyciela akademickiego		L. godzin	ECTS
		15	0,6
Zaj cia o charakterze praktycznym		L. godzin	ECTS
		25	1,0

1 godz = 45 minut; 1 punkt ECTS = 25-30 godzin

W sekcji 'Liczba punktów ECTS' suma punktów ECTS zaj wymagaj cych bezpo redniego udziału nauczyciela akademickiego i o charakterze praktycznym mo e si ró ni od ł cznej liczby punktów ECTS dla zaj /grup zaj .