

Politechnika Białostocka										
Kierunek studiów	Informatyka							Poziom i forma studiów	drugiego stopnia stacjonarne	
Specjalność / Ścieżka dyplomowania	Biometria i przetwarzanie sygnałów							Profil kształcenia	ogólnoakademicki	
Nazwa przedmiotu	Programowanie akceleratorów graficznych							Kod przedmiotu	INF2PAG	
								Rodzaj przedmiotu	obieralny	
Forma zajęć i liczba godzin	W	Ć	L	P	Ps	T	S	Semestr	2	
	15				15			Punkty ECTS	2	
Przedmioty wprowadzające										
Cele przedmiotu	Celem przedmiotu jest zapoznanie studentów z zaawansowanymi technikami akceleracji sprzętowej w oparciu o procesory graficzne (GPU) oraz rozszerzenia multimedialne SIMD (MMX/SSE) i wielordzeniowość jednostek ogólnego przeznaczenia (CPU). W szczególności student zostanie zaznajomiony z technologiami: OpenMP, Intel threading building blocks (TBB), językiem cieniowania (ang. shader) GLSL oraz z dedykowanymi środowiskami obliczeń ogólnych GPGPU: Cuda, OpenCL.									
Treści programowe	Wykład i pracownia specjalistyczna: Obliczenia w grafice i multimediach, wektoryzacja kodu, obliczenia współbieżne. Rozszerzenia multimedialne (MMX/SSE) procesorów ogólnego przeznaczenia, wprowadzenie do architektury 64bit. Wykorzystanie wstawek assemblera oraz funkcji wbudowanych (ang. Intrinsics), optymalizacja kodu. Programowanie w oparciu o procesory wielordzeniowe – technologie OpenMP, TBB. Architektura akceleratorów graficznych, unifikacja jednostek obliczeniowych GPU. Języki cieniowania (ang. shaders), technologia GLSL (OpenGL Shading Language). Tworzenie zaawansowanych efektów graficznych z wykorzystaniem GLSL. Wprowadzenie do obliczeń ogólnych z wykorzystaniem jednostek GPU: shadery obliczeniowe GLSL/Vulkan oraz technologie Cuda i OpenCL.									
Metody dydaktyczne	programowanie z użyciem komputera, wykład informacyjny, ćwiczenia przedmiotowe, wykład problemowy,									
Forma zaliczenia	Wykład: zaliczenie pisemne. Pracownia specjalistyczna: wykonanie zadań projektowych, obrona projektu.									
Symbol efektu uczenia się	Zakładane efekty uczenia się								Odniesienie do kierunkowych efektów uczenia się	
EU1	posiada wiedzę o aktualnych trendach i narzędziach w zakresie akceleracji sprzętowej z wykorzystaniem procesorów GPU oraz wielordzeniowych jednostek ogólnego przeznaczenia								K_W06	
EU2	ma rozszerzoną i pogłębioną wiedzę z zakresu grafiki komputerowej								K_W01	
EU3	potrafi wykorzystywać poznane technologie do przetwarzania obrazów								K_U06	
EU4	ma podstawową wiedzę o aktualnych trendach w zakresie obliczeń z wykorzystaniem jednostek GPU, potrafi określić kierunki dalszego uczenia się i zrealizować proces samokształcenia.								K_U05	
Symbol efektu uczenia się	Sposób weryfikacji efektu uczenia się								Forma zajęć na której zachodzi weryfikacja	
EU1	zaliczenie pisemne								W	
EU2	zaliczenie pisemne								W	
EU3	ocena zadań projektowych na pracowni specjalistycznej								Ps	
EU4	ocena zadań projektowych na pracowni specjalistycznej								Ps	
Bilans nakładu pracy studenta (w godzinach)								Liczba godz.		
Wyliczenie										
	1 - Udział w wykładach - 15x1								15	
	2 - Udział w pracowni specjalistycznej - 15x1								15	
	3 - Udział w konsultacjach -								5	
	4 - Opracowanie projektów i prac domowych -								10	
	5 - Przygotowanie do zaliczenia -								8	
RAZEM:								53		
Wskaźniki ilościowe								GODZINY	ECTS	
Nakład pracy studenta związany z zajęciami wymagającymi bezpośredniego udziału nauczyciela								35 (1)+(2)+(3)	1,3	
Nakład pracy studenta związany z zajęciami o charakterze praktycznym								25 (2)+(4)	0,9	
Literatura podstawowa	1. K. Sobiesiak, P. Sydow: Zaawansowane programowanie w GLSL, PWN, 2015. 2. J. Kessenich (ed.): The OpenGL Shading Language v 4.2, The Khronos Group, 2011 (dok. On-line). 3. A. Munshi (ed.): The OpenCL Specification, Version: 1.2, Khronos OpenCL Working Group, 2011 (dok. On-line).									
Literatura uzupełniająca	1. NVIDIA CUDA C Programming Guide, Version 4.2 (dok. on-line). 2. Y. Magda: Visual C++ .NET Optimization with Assembly Code, A-LIST Publishing, 2004. 3. Wen-mei W. Hwu (ed.): GPU Computing Gems Emerald Edition, Morgan Kaufmann, 2011. 4. D.B. Kirk,Wen-mei W. Hwu: Programming Massively Parallel Processors: A Hands-on Approach, Morgan Kaufmann, 2011. 5. OpenMP Application Program Interface, OpenMP Architecture Review Board, 2011 (dok. On-line). 6. Intel(R) Threading Building Blocks Reference Manual, Intel Corp., 2012 (dok. on-line).									
Jednostka realizująca	Katedra Mediów Cyfrowych i Grafiki Komputerowej								Data opracowania programu	
Program opracował(a)	dr inż. Adam Borowicz								5 kwietnia 2019	