



KARTA MODUŁU / KARTA PRZEDMIOTU

Kod modułu	
Nazwa modułu	Współczesne architektury procesorów graficznych
Nazwa modułu w języku angielskim	Architecture of modern graphics processor units
Obowiązuje od roku akademickiego	2012/13

A. USYTUOWANIE MODUŁU W SYSTEMIE STUDIÓW

Kierunek studiów	Informatyka
Poziom kształcenia	II stopień (I stopień / II stopień)
Profil studiów	Ogólnoakademicki (ogólno akademicki / praktyczny)
Forma i tryb prowadzenia studiów	stacjonarne (stacjonarne / niestacjonarne)
Specjalność	Grafika komputerowa
Jednostka prowadząca moduł	Katedra Informatyki
Koordynator modułu	Roman Stanisław Deniziak
Zatwierdził:	

B. OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU

Przynależność do grupy/bloku przedmiotów	Kierunkowy (podstawowy / kierunkowy / inny HES)
Status modułu	obowiązkowy (obowiązkowy / nieobowiązkowy)
Język prowadzenia zajęć	polski
Usytuowanie modułu w planie studiów - semestr	1
Usytuowanie realizacji przedmiotu w roku akademickim	Semestr letni (semestr zimowy / letni)
Wymagania wstępne	(kody modułów / nazwy modułów)
Egzamin	tak (tak / nie)
Liczba punktów ECTS	5

Forma prowadzenia zajęć	wykład	ćwiczenia	laboratorium	projekt	inne
w semestrze	30		30		



C. EFEKTY KSZTAŁCENIA I METODY SPRAWDZANIA EFEKTÓW KSZTAŁCENIA

Cel modułu	Poznanie architektur współczesnych procesorów graficznych, zasady przetwarzania SPMD, wykorzystanie procesorów graficznych do obliczeń dużej mocy. Nabycie umiejętności programowania równoległego wykorzystującego procesory graficzne zgodne z modelem CUDA. Nabycie umiejętności programowania heterogenicznych architektur równoległych w języku OpenCL. <i>(3-4 linijki)</i>
-------------------	--

Symbol efektu	Efekty kształcenia	Forma prowadzenia zajęć (w/ć/l/p/inne)	odniesienie do efektów kierunkowych	odniesienie do efektów obszarowych
W_01	Znajomość struktury współczesnych procesorów graficznych.	W	K_W08	T1A_W03
W_02	Znajomość zasad programowania równoległego heterogenicznych architektur komputerów opartych o wielordzeniowe procesory uniwersalne i procesory graficzne.	W	K_W03 K_W09	T1A_W04 T1A_W07
W_03	Znajomość zasad przetwarzania SPMD (Single Program Multiple Data) w procesorach graficznych.	W	K_W09	T1A_W03
W_04	Znajomość kierunków rozwoju architektur procesorów graficznych.	W	K_W11	T1A_W05
U_01	Umiejętność projektowania równoległych algorytmów SPMD oraz ich implementacji w języku C(CUDA) i OpenCL.	L	K_U08 K_U11 K_U13 K_U14 K_U17	T1A_U16

Treści kształcenia:

1. Treści kształcenia w zakresie wykładu

Nr wykładu	Treści kształcenia	Odniesienie do efektów kształcenia dla modułu
1	Zasady przetwarzania równoległego w procesorach graficznych.	W_03
2	Model CUDA: zasady przetwarzania, programowanie w języku C.	W_02
3	Język OpenCL: zasady programowania wielowątkowego, zastosowanie do GPGPU.	W_02
4	Struktury współczesnych procesorów graficznych stosowanych w komputerach PC, stacjach graficznych, superkomputerach i systemach wbudowanych.	W_01
5	Kierunki rozwoju architektur procesorów graficznych oraz komputerów	W_04

2. Treści kształcenia w zakresie ćwiczeń

Nr zajęć ćwicz.	Treści kształcenia	Odniesienie do efektów kształcenia dla modułu



3. Treści kształcenia w zakresie zadań laboratoryjnych

Nr zajęć lab.	Treści kształcenia	Odniesienie do efektów kształcenia dla modułu
1	Wstęp do programowania z użyciem GPGPU oraz OpenCL	U_01
2	Interakcja CPU – GPU część 1. Sposoby przesyłania oraz przetwarzania tablic jednowymiarowych z użyciem OpenCL.	U_01
3	Interakcja CPU – GPU część 2. Sposoby przesyłania oraz przetwarzania tablic wielowymiarowych z użyciem OpenCL.	U_01
4	Model pamięci w OpenCL.	U_01
5	Implementacja wybranych algorytmów numerycznych w OpenCL.	U_01
6	Obsługa oraz podstawowe przetwarzanie obrazów w OpenCL.	U_01
7	Implementacja wybranych algorytmów graficznych w OpenCL.	U_01
8	Wstęp do programowania z użycie CUDA.	U_01
9	Interakcja CPU – GPU. Sposoby przesyłania oraz przetwarzania tablic jednowymiarowych i wielowymiarowych z użyciem CUDA.	U_01
10	Model pamięci w CUDA.	U_01
11	Obsługa oraz podstawowe przetwarzanie obrazów w CUDA.	U_01
12, 13	Implementacja prostego silnika fizycznego z użyciem CUDA i OpenCL.	U_01
14, 15	Implementacja prostego ray-tracera z użyciem CUDA i OpenGL.	U_01

4. Charakterystyka zadań projektowych

5. Charakterystyka zadań w ramach innych typów zajęć dydaktycznych

Metody sprawdzania efektów kształcenia

Symbol efektu	Metody sprawdzania efektów kształcenia (sposób sprawdzenia, w tym dla umiejętności – odwołanie do konkretnych zadań projektowych, laboratoryjnych, itp.)
W_01	egzamin
W_02	egzamin
W_03	egzamin
W_04	egzamin
U_01	Zaliczenie ćwiczeń laboratoryjnych.



D. NAKŁAD PRACY STUDENTA

Bilans punktów ECTS		
	Rodzaj aktywności	obciążenie studenta
1	Udział w wykładach	30
2	Udział w ćwiczeniach	
3	Udział w laboratoriach	30
4	Udział w konsultacjach (2-3 razy w semestrze)	3
5	Udział w zajęciach projektowych	
6	Konsultacje projektowe	
7	Udział w egzaminie	2
9	Liczba godzin realizowanych przy bezpośrednim udziale nauczyciela akademickiego	65 (suma)
10	Liczba punktów ECTS, którą student uzyskuje na zajęciach wymagających bezpośredniego udziału nauczyciela akademickiego (1 punkt ECTS=25-30 godzin obciążenia studenta)	3
11	Samodzielne studiowanie tematyki wykładów	15
12	Samodzielne przygotowanie się do ćwiczeń	
13	Samodzielne przygotowanie się do kolokwium	
14	Samodzielne przygotowanie się do laboratoriów	15
15	Wykonanie sprawozdań	15
15	Przygotowanie do kolokwium końcowego z laboratorium	
17	Wykonanie projektu lub dokumentacji	
18	Przygotowanie do egzaminu	15
20	Liczba godzin samodzielnej pracy studenta	60 (suma)
21	Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy (1 punkt ECTS=25-30 godzin obciążenia studenta)	2
22	Sumaryczne obciążenie pracą studenta	125
23	Punkty ECTS za moduł 1 punkt ECTS=25-30 godzin obciążenia studenta	5
24	Nakład pracy związany z zajęciami o charakterze praktycznym Suma godzin związanych z zajęciami praktycznymi	60
25	Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym 1 punkt ECTS=25-30 godzin obciążenia studenta	2

E. LITERATURA

Wykaz literatury	1. Pharr M., Fernando F. "GPU Gems 3", Addison Wesley, 2008. 2. Edward Kandrot, Jason Sanders, "CUDA w przykładach. Wprowadzenie do ogólnego programowania procesorów GPU, Helion, 2012. 3. Aaftab Munshi, The OpenCL Specification, The Khronos Group Inc., 2011. http://www.khronos.org/registry/cl/specs/opencl-1.2.pdf 4. Materiały firmowe Nvidia, AMD.
Witryna WWW modułu/przedmiotu	http://achilles.tu.kielce.pl/Members/sdeniziak/kierunek-informatyka-ii-st/wspo142czesne-procesory-graficzne/