



Karta przedmiotu

Nazwa i kod przedmiotu	Programowanie równoległe, PG_00068272						
Kierunek studiów	Automatyka, cybernetyka i robotyka						
Data rozpoczęcia studiów	październik 2026 r.	Rok akademicki realizacji przedmiotu			2027/2028		
Poziom kształcenia	I stopnia - inżynierskie	Grupa zajęć			Grupa zajęć fakultatywnych Grupa zajęć powiązanych z prowadzonymi badaniami naukowymi w dziedzinie nauki związanej z kierunkiem - profil ogólnoakademicki		
Forma studiów	stacjonarne	Sposób realizacji			na uczelni		
Rok studiów	2	Język wykładowy			polski		
Semestr studiów	4	Liczba punktów ECTS			3.0		
Profil kształcenia	ogólnoakademicki	Forma zaliczenia			zaliczenie		
Jednostka prowadząca	Wydziały Politechniki Gdańskiej -> Wydział Elektroniki, Telekomunikacji i Informatyki -> Katedra Systemów Decyzyjnych i Robotyki						
Imię i nazwisko wykładowcy (wykładowców)	Odpowiedzialny za przedmiot	dr hab. inż. Tomasz Stefański					
	Prowadzący zajęcia z przedmiotu	dr hab. inż. Tomasz Stefański					
Formy zajęć	Forma zajęć	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium	RAZEM
	Liczba godzin zajęć	15.0	0.0	15.0	0.0	0.0	30
	W tym liczba godzin zajęć na odległość: 0.0						
Aktywność studenta i liczba godzin pracy	Aktywność studenta	Udział w zajęciach dydaktycznych, objętych planem studiów		Udział w konsultacjach		Praca własna studenta	RAZEM
	Liczba godzin pracy studenta	30		5.0		40.0	75
Cel przedmiotu	Zapoznanie studentów z programowaniem równoległym systemów wieloprocesorowych tj. kart graficznych, jednostek centralnych i ich klastrów, ze szczególnym naciskiem na programowanie w środowisku CUDA.						

Efekty uczenia się przedmiotu	Efekt kierunkowy	Efekt z przedmiotu	Sposób weryfikacji i oceny efektu
	[K6_U04] potrafi wykorzystywać posiadaną wiedzę z zakresu metod i technik programowania oraz dobrać i zastosować właściwe metody i narzędzia programistyczne w tworzeniu oprogramowania komputerów albo programowania urządzeń lub sterowników wykorzystujących mikroprocesory albo elementy lub układy programowalne, charakterystycznych dla danego kierunku studiów	Student potrafi pisać programy inżynierskie w środowisku CUDA.	[SU4] Ocena umiejętności korzystania z metod i narzędzi [SU1] Ocena realizacji zadania
	[K6_U12] potrafi analizować działanie elementów, układów i systemów związanych z kierunkiem studiów oraz mierzyć ich parametry i badać charakterystyki techniczne, a także planować i przeprowadzać eksperymenty związane z kierunkiem studiów, w tym pomiary i symulacje komputerowe, oraz interpretować uzyskane wyniki i wyciągać wnioski	Student potrafi analizować działanie programów równoległych napisanych w językach C/C++ stosujących systemy wieloprocessorowe.	[SU3] Ocena umiejętności wykorzystania wiedzy uzyskanej w ramach przedmiotu [SU4] Ocena umiejętności korzystania z metod i narzędzi
[K6_W04] zna i rozumie w zaawansowanym stopniu zasady, metody i techniki programowania oraz zasady tworzenia oprogramowania komputerów albo programowania urządzeń lub sterowników wykorzystujących mikroprocesory albo elementy lub układy programowalne, specyficznych dla kierunku studiów, a także organizację pracy systemów wykorzystujących komputery lub te urządzenia	Student rozumie zasady programowania równoległego i potrafi pisać programy stosujące wielordzeniowe procesory i karty graficzne.	[SW1] Ocena wiedzy faktograficznej	
Treści przedmiotu	<p>Treści przedmiotu - wykład</p> <ol style="list-style-type: none"> 1. Wprowadzenie do programowania równoległego. 2. Architektury systemów wieloprocessorowych, tj. kart graficznych, jednostek centralnych i ich klastrów, oraz standardy ich programowania. 3. Standardy programowania OpenMP i MPI. 4. Wprowadzenie do środowiska CUDA programowania kart graficznych. 5. Omówienie kompilatora (nvcc), debugera (cuda-gdb) oraz profilera w środowisku CUDA. 6. Model programowania równoległego w środowisku CUDA: jądra obliczeniowe, wątki, bloki wątków, hierarchia pamięci. 7. Interfejs programowania CUDA C. 8. Optymalizacja kodu w środowisku CUDA. 9. Przegląd bibliotek programistycznych dostępnych w ramach środowiska CUDA. 10. Wprowadzenie do standardu OpenCL. Heterogeniczne systemy wieloprocessorowe. Zastosowanie standardu OpenCL do programowania macierzy bramek FPGA. 11. Model programowania równoległego w standardzie OpenCL: jądra obliczeniowe, jednostki robocze, grupy robocze, hierarchia pamięci, sekwencje operacji. 		
Wymagania wstępne i dodatkowe	Student posiada podstawową wiedzę z programowania.		
Sposoby i kryteria oceniania osiągniętych efektów uczenia się	Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa ocena końcowej
	projekt	40.0%	50.0%
	laboratorium	40.0%	30.0%
	wykład	40.0%	20.0%
Zalecana lista lektur	Podstawowa lista lektur	<ul style="list-style-type: none"> - CUDA C Programming Guide - CUDA Runtime API - CUDA C Best Practices Guide 	
	Uzupełniająca lista lektur	<ul style="list-style-type: none"> - CUDA Compiler Driver NVCC - CUDA Visual Profiler - CUDA-gdb debugger - CUDA-memcheck memory checker 	
	Adresy eZasobów		
Przykładowe zagadnienia/ przykładowe pytania/ realizowane zadania	<ol style="list-style-type: none"> 1. Model programowania równoległego CUDA. 2. Rodzaje pamięci na karcie graficznej. 3. Zastosowanie pamięci współdzielonej. 4. Strumienie w środowisku CUDA. 5. Zdarzenia w środowisku CUDA. 		
Zajęcia praktyczne w ramach przedmiotu	Nie dotyczy		

Dokument wygenerowany elektronicznie. Nie wymaga pieczęci ani podpisu.