

1.	Nazwa kierunku	informatyka
2.	Wydział	Wydział Nauk Ścisłych i Technicznych
3.	Cykl rozpoczęcia	2019/2020 (semestr letni)
4.	Poziom kształcenia	studia drugiego stopnia
5.	Profil kształcenia	ogólnoakademicki
6.	Forma prowadzenia studiów	stacjonarna

Moduł kształcenia: Programowanie współbieżne

Kod modułu: 08-IN-IJO-S2-PW

1. Liczba punktów ECTS: 3

2. Zakładane efekty uczenia się modułu			
kod	opis	efekty uczenia się kierunku	stopień realizacji (skala 1-5)
PW_1	Zna i potrafi zastosować mechanizmy takie jak wątek, zmienna i instrukcja atomowa, semafor, monitor w implementacji programów współbieżnych.	K_U12 K_W06	2 2
PW_2	Potrafi dokonać dekompozycji rozpatrywanego problemu obliczeniowego na składowe umożliwiające opracowanie algorytmu równoległego.	K_U12 K_W06	2 2
PW_3	Ma wiedzę na temat modeli obliczeń równoległych.	K_U12 K_W06	1 3
PW_4	Potrafi ocenić efektywność algorytmów równoległych za pomocą podstawowych miar, takich jak przyspieszenie, koszt i efektywność. Potrafi ocenić możliwe do uzyskania przyspieszenie na podstawie prawa Amdahla oraz Gustavsona.	K_W07 K_W09	1 1
PW_5	Ma wiedzę na temat architektury programowalnych układów graficznych (GPU) i narzędzi programistycznych umożliwiającą tworzenie dedykowanego oprogramowania.	K_K01 K_W04 K_W07 K_W15	2 2 2 1
PW_6	Ma wiedzę na temat wzorców komunikacji i typowych operacji zbiorczych w programach równoległych ze szczególnym uwzględnieniem GPU. Zna algorytmy równoległe takie jak redukcja, suma prefiksowa, histogram, wybrane algorytmy sortowania.	K_U12 K_W06	1 3
PW_7	Potrafi rozwiązać problem wzajemnego wykluczania w środowisku rozproszonym, a także zaimplementować algorytm uzgadniania rozproszonego.	K_U12 K_W13	1 1
PW_8	Potrafi pracować w zespole dwuosobowym i dokonuje właściwego podziału pracy	K_K01	1

		K_U02	1
--	--	-------	---

3. Opis modułu

Opis	Celem zajęć jest przygotowanie studentów do projektowania i implementacji poprawnych i efektywnych algorytmów współbieżnych oraz rozproszonych. W trakcie zajęć studenci uzyskują wiedzę na temat typowych problemów związanych z projektowaniem poprawnych i efektywnych programów współbieżnych, a także sposobów ich rozwiązywania
Wymagania wstępne	

4. Sposoby weryfikacji efektów uczenia się modułu

kod	nazwa (typ)	opis	efekty uczenia się modułu
PW_w_1	Prace kontrolne	Co najmniej jedno kolokwium sprawdzające wiedzę z tematyki poruszanej na wykładzie oraz realizowanej w ramach laboratoriów.	PW_1, PW_2, PW_3, PW_4, PW_5, PW_6, PW_7
PW_w_2	Projekt programistyczny	Realizacja projektu / projektów programistycznych pozwalających na praktyczną weryfikację nabytej wiedzy i umiejętności dotyczących programowania współbieżnego, ze szczególnym uwzględnieniem programowania układów graficznych oraz programowania komputerów z pamięcią rozproszoną.	PW_5, PW_6, PW_7, PW_8
PW_w_3	Egzamin	Test złożony z pytań wielokrotnego wyboru oraz zadań otwartych dotyczących tematów poruszanych na wykładach oraz laboratoriach.	PW_1, PW_2, PW_3, PW_4, PW_5, PW_6, PW_7

5. Rodzaje prowadzonych zajęć

kod	rodzaj prowadzonych zajęć			praca własna studenta		sposoby weryfikacji efektów uczenia się
	nazwa	opis (z uwzględnieniem metod dydaktycznych)	liczba godzin	opis	liczba godzin	
PW_fs_1	wykład	Podanie treści kształcenia w formie werbalnej z wykorzystaniem wizualizacji treści. Omówienie ważnych kwestii teoretycznych i praktycznych dotyczących programowania współbieżnego, ze szczególnym uwzględnieniem programowania równoległego.	15	Zapoznanie się z tematyką prezentowaną podczas wykładów, w tym lektura literatury obowiązkowej oraz uzupełniającej.	15	PW_w_1
PW_fs_2	laboratorium	Szczegółowe przygotowanie studentów do tworzenia aplikacji współbieżnych w nowoczesnych językach programowania. Prezentacja i omówienie narzędzi wspierających realizację oprogramowania współbieżnego.	30	Rozwiązywanie zadań praktycznych z poszczególnych tematów wraz z analizą rozwiązań już istniejących. Zrealizowanie projektu programistycznego z zastosowaniem prezentowanych na wykładach metod.	30	PW_w_2, PW_w_3