

1.	Nazwa kierunku	informatyka
2.	Cykl rozpoczęcia	2015/2016 (semestr zimowy)
3.	Poziom kształcenia	studia pierwszego stopnia (inżynierskie)
4.	Profil kształcenia	ogólnoakademicki
5.	Forma prowadzenia studiów	niestacjonarna

Moduł kształcenia: Programowanie równoległe

Kod modułu: 08- IO1N-13-PR

1. Liczba punktów ECTS: 4

2. Zakładane efekty kształcenia modułu			
kod	opis	efekty kształcenia kierunku	stopień realizacji (skala 1-5)
ZP-K_9	Potrafi pracować w zespole dwuosobowym i dokonuje właściwego podziału pracy	K_1_A_I_K03	1
ZP-U_6	Potrafi dokonać dekompozycji problemu obliczeniowego na potrzeby obliczeń równoległych	K_1_A_I_U13	1
ZP-U_7	Potrafi zaimplementować program współbieżny i zastosować podstawowe mechanizmy współbieżności	K_1_A_I_U13 K_1_A_I_U14 K_1_A_I_U15 K_1_A_I_U16 K_1_A_I_U17	1 1 1 1 1
ZP-U_8	Potrafi zaimplementować program równoległy dla komputerów ze współdzieloną pamięcią, w tym również z użyciem GPU	K_1_A_I_U13 K_1_A_I_U14 K_1_A_I_U15 K_1_A_I_U16 K_1_A_I_U17	1 1 1 1 1
ZP-W_1	Ma podstawową wiedzę na temat architektury współczesnych komputerów i jej znaczenia dla programowania równoległego	K_1_A_I_K01 K_1_A_I_K06 K_1_A_I_W06	1 1 1
ZP-W_2	Ma podstawową wiedzę na temat modeli obliczeń równoległych	K_1_A_I_W09	1
ZP-W_3	Ma wiedzę na temat sposobów oceny efektywności algorytmów równoległych	K_1_A_I_W04	1

		K_1_A_I_W09	1
ZP-W_4	Ma wiedzę na temat podstawowych algorytmów równoległych i ich zastosowań	K_1_A_I_W09	1
ZP-W_5	Ma wiedzę na temat architektury GPU i ich zastosowań w obliczeniach ogólnego przeznaczenia	K_1_A_I_W09	1

3. Opis modułu	
Opis	Celem zajęć jest przygotowanie studentów do projektowania i implementacji poprawnych i efektywnych algorytmów równoległych. W ramach zajęć studenci uzyskują wiedzę i umiejętności dotyczące metod i narzędzi przydatnych w programowaniu komputerów równoległych, w tym również procesorów graficznych (GPU).
Wymagania wstępne	Podstawowa znajomość programowania w językach C oraz Java. Znajomość obsługi zintegrowanych środowisk programistycznych i podstawowych narzędzi programistycznych wspomagających tworzenie i testowanie oprogramowania.

4. Sposoby weryfikacji efektów kształcenia modułu			
kod	nazwa (typ)	opis	efekty kształcenia modułu
ZP_w_1	Prace kontrolne	Co najmniej jedno kolokwium sprawdzające wiedzę z tematyki poruszanej na wykładzie oraz realizowanej w ramach laboratoriów.	ZP-U_6, ZP-U_7, ZP-U_8, ZP-W_1, ZP-W_2, ZP-W_3, ZP-W_4, ZP-W_5
ZP_w_2	Projekt programistyczny	Realizacja projektu / projektów programistycznych pozwalających na praktyczną weryfikację nabytej wiedzy i umiejętności dotyczących programowania równoległego	ZP-K_9, ZP-U_6, ZP-U_7, ZP-W_5
ZP_w_3	Zaliczenie wykładu	Test złożony z pytań wielokrotnego wyboru oraz zadań otwartych dotyczących tematów poruszanych na wykładach oraz laboratoriach.	ZP-U_6, ZP-U_7, ZP-U_8, ZP-W_1, ZP-W_2, ZP-W_3, ZP-W_4, ZP-W_5

5. Rodzaje prowadzonych zajęć						
kod	rodzaj prowadzonych zajęć			praca własna studenta		sposoby weryfikacji efektów kształcenia
	nazwa	opis (z uwzględnieniem metod dydaktycznych)	liczba godzin	opis	liczba godzin	
ZP_fs_1	wykład	Podanie treści kształcenia w formie werbalnej z wykorzystaniem wizualizacji treści. Omówienie ważnych kwestii teoretycznych i praktycznych dotyczących programowania współbieżnego.	20	Zapoznanie się z tematyką prezentowaną podczas wykładów oraz przygotowanie się do laboratoriów związanych z wykładami.	20	ZP_w_1
ZP_fs_2	laboratorium	Szczegółowe przygotowanie studentów do tworzenia aplikacji współbieżnych w nowoczesnych językach programowania. Prezentacja i omówienie narzędzi wspierających realizację oprogramowania współbieżnego.	20	Rozwiązywanie zadań praktycznych z poszczególnych tematów wraz z analizą rozwiązań już istniejących. Zrealizowanie projektu programistycznego z zastosowaniem prezentowanych na wykładach metod.	60	ZP_w_2, ZP_w_3