



Karta przedmiotu

Nazwa i kod przedmiotu	Concurrent and distributed processing, PG_00061802										
Kierunek studiów	Inżynieria danych										
Data rozpoczęcia studiów	październik 2024 r.	Rok akademicki realizacji przedmiotu		2026/2027							
Poziom kształcenia	I stopnia - inżynierskie	Grupa zajęć		Grupa zajęć fakultatywnych Grupa zajęć powiązanych z prowadzonymi badaniami naukowymi w dziedzinie nauki związanej z kierunkiem - profil ogólnoakademycki							
Forma studiów	stacjonarne	Sposób realizacji		na uczelni							
Rok studiów	3	Język wykładowy		polski							
Semestr studiów	6	Liczba punktów ECTS		5.0							
Profil kształcenia	ogółnoakademicki	Forma zaliczenia		zaliczenie							
Jednostka prowadząca	Wydział Elektroniki, Telekomunikacji i Informatyki -> Katedra Architektury Systemów Komputerowych										
Imię i nazwisko wykładowcy (wykładowców)	Odpowiedzialny za przedmiot	dr inż. Mariusz Matuszek									
	Prowadzący zajęcia z przedmiotu	dr inż. Tadeusz Matuszek									
Formy zajęć i metody nauczania	Forma zajęć	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium	RAZEM				
	Liczba godzin zajęć	15.0	0.0	30.0	15.0	0.0	60				
	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	60		10.0	55.0		125				
Cel przedmiotu	Nauczenie podstaw oraz zasad przetwarzania rozproszonego i równoległego w sieciowych systemach komputerowych.										
Efekty uczenia się przedmiotu	[K6_W06] klasyfikuje pozyskiwane informacje, oceniac ich przydatność do rozwiązania sformułowanych problemów  [K6_W01] identyfikuje uwarunkowania procesów zachodzących w analizowanych systemach i dobiera metody ich rozwiązania, wykorzystując zgromadzoną wiedzę i uwzględniając wzajemne relacje między analizowanymi zjawiskami  [K6_U04] formułuje logiczne rozwiązania złożonych lub nieuszkuryzowanych problemów	Efekt kierunkowy	Efekt z przedmiotu		Sposób weryfikacji i oceny efektu						
		[K6_W06] klasyfikuje pozyskiwane informacje, oceniac ich przydatność do rozwiązania sformułowanych problemów	Student rozróżnia podstawowe wzorce projektowe w programowaniu spółbieżnym i potrafi je opisać		[SW1] Ocena wiedzy faktograficznej						
		[K6_W01] identyfikuje uwarunkowania procesów zachodzących w analizowanych systemach i dobiera metody ich rozwiązania, wykorzystując zgromadzoną wiedzę i uwzględniając wzajemne relacje między analizowanymi zjawiskami	Student potrafi podać genezę zjawiska wy ciągu i zastosować odpowiednie metody ochrony przed wystąpieniem tego zjawiska.		[SW3] Ocena wiedzy zawartej w opracowaniu tekstowym i projektowym						
		[K6_U04] formułuje logiczne rozwiązania złożonych lub nieuszkuryzowanych problemów	Student wytwarza nietrywialną aplikację rozproszoną realizującą współbieżne przetwarzanie danych i ochronę synchronizowanego globalnego stanu.		[SU4] Ocena umiejętności korzystania z metod i narzędzi [SU1] Ocena realizacji zadania						

Treści przedmiotu	<ol style="list-style-type: none"> <li>1. Zapoznanie z przedmiotem. Przedstawienie zasad zaliczeń</li> <li>2. Abstrakcja programowania rozproszonego i współbieżnego</li> <li>3. Przetwarzanie rozproszone i równolegle w przykładach</li> <li>4. Sekcja krytyczna wprowadzenie</li> <li>5. Przegląd klasycznych problemów: producent konsument, czytelnicy pisarze, pięciu filozofów</li> <li>6. Semafor szczegółowa klasyfikacja i omówienie</li> <li>7. Procedury współbieżne i procedury wielowejściowe</li> <li>8. Rozwiązania typowych problemów z zastosowaniem semaforów</li> <li>9. Semafor binarne i uogólnione w systemie Unix</li> <li>10. Programowanie wielowątkowe</li> <li>11. Synchronizacja dostępu i wykonania dla wątków i procesów</li> <li>12. Biblioteki funkcji wielowątkowych w systemie Unix</li> <li>13. Monitory wprowadzenie i omówienie mechanizmu</li> <li>14. Wykorzystanie monitorów w rozwiązywaniu typowych problemów przykłady</li> <li>15. Zmienne warunkowe w systemie Unix, praktyczna implementacja procedur monitorowych</li> <li>16. Porównanie i zestawienie semaforów z mechanizmami monitorowymi - podejście teoretyczne</li> </ol>				
Wymagania wstępne i dodatkowe	Znajomość podstaw programowania w języku C jest pomocna.				
Sposoby i kryteria oceniania osiąganych efektów uczenia się	Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa oceny końcowej		
	Kolokwia	50.0%	40.0%		
	Laboratoria	50.0%	40.0%		
	Projekt semestralny	50.0%	20.0%		
Zalecana lista lektur	<p>Podstawowa lista lektur</p> <p>Uzupełniająca lista lektur</p> <p>Adresy eZasobów</p>	<ol style="list-style-type: none"> <li>1. Ben-Ari M.: Podstawy programowania współbieżnego, Wydawnictwa Naukowo Techniczne, Warszawa.</li> <li>2. Colouris G., Dollimore J., Kindberg G.: Distributed Systems, Concepts and Design, second edition, Addison-Wesley.</li> <li>3. Coulouris G., Dollimore J., Kindberg T.: Systemy rozproszone Podstawy i projektowanie, Wydawnictwa Naukowo Techniczne, Warszawa.</li> <li>4. Hwang K., Briggs F.: Computer Architecture and Parallel Processing, McGraw - Hill.</li> </ol> <ol style="list-style-type: none"> <li>1. Lister A., Eager R.: Wprowadzenie do systemów operacyjnych, Wydawnictwa Naukowo Techniczne, Warszawa.</li> <li>2. Silberschatz A., Gavlin P.: Podstawy systemów operacyjnych, Wydawnictwa Naukowo Techniczne, Warszawa.</li> <li>3. Stevens R.: Unix Network Programming, Prentice Hall.</li> </ol> <p>Adresy eZasobów</p>			
Przykładowe zagadnienia/ przykładowe pytania/ realizowane zadania					
Praktyki zawodowe w ramach przedmiotu	Nie dotyczy				

Dokument wygenerowany elektronicznie. Nie wymaga pieczęci ani podpisu.