



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

Projekt współfinansowany przez
Unię Europejską w ramach
Europejskiego Funduszu
Społecznego

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Nazwa przedmiotu		Kod ECTS	
Programowanie współbieżne		11.3.1543	
Nazwa jednostki prowadzącej przedmiot			
Instytut Informatyki			
Studia			
wydział	kierunek	poziom	pierwszego stopnia
Wydział Matematyki, Fizyki i Informatyki	Informatyka	forma	stacjonarne
		moduł	wszystkie
		specjalnościowy	wszystkie
		specjalizacja	wszystkie
Nazwisko osoby prowadzącej (osób prowadzących)			
dr Paweł Pączkowski			
Formy zajęć, sposób ich realizacji i przypisana im liczba godzin		Liczba punktów ECTS	
Formy zajęć		6 Przedmiot w wymiarze 30h wykładu i 30h lab. + praca własna studenta	
Wykład, Ćw. laboratoryjne			
Sposób realizacji zajęć			
zajęcia w sali dydaktycznej			
Liczba godzin			
Wykład: 30 godz., Ćw. laboratoryjne: 30 godz.			
Termin realizacji przedmiotu			
2022/2023 letni			
Status przedmiotu		Język wykładowy	
fakultatywny (do wyboru)		polski	
Metody dydaktyczne		Forma i sposób zaliczenia oraz podstawowe kryteria oceny lub wymagania egzaminacyjne	
<ul style="list-style-type: none"> - Wykład z prezentacją multimedialną - laboratorium - pisanie i testowanie programów 		Sposób zaliczenia	
		<ul style="list-style-type: none"> - Zaliczenie na ocenę - Egzamin 	
		Formy zaliczenia	
		<ul style="list-style-type: none"> - egzamin pisemny z pytaniami (zadaniami) otwartymi - egzamin pisemny testowy - ustalenie oceny zaliczeniowej na podstawie ocen cząstkowych otrzymywanych w trakcie trwania semestru 	
		Podstawowe kryteria oceny	
		Wykład - liczba punktów uzyskanych na egzaminie. Laboratorium - liczba punktów uzyskanych za napisane i zaliczone programy.	
Sposób weryfikacji założonych efektów uczenia się			

zakładany efekt kształcenia	egzamin	odpowiedź ustna	program	sprawdzian	referat	raport	aktywność w dyskusji	obserwacja postawy studenta
Wiedza								
K_W05	X	X	X					
K_W07	X		X					
K_W10								X
P_W1	X	X						
P_W2	X		X					
P_W3	X		X					
Umiejętności								
K_U02			X					X
K_U04			X					
K_U06	X		X					
P_U1	X		X					X
P_U2		X	X					
Kompetencje								
K_K01								X
K_K03							X	X
P_K1							X	X

Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi

A. Wymagania formalne

Zaliczony przedmiot Systemy operacyjne

B. Wymagania wstępne

Umiejętność programowania w języku C lub C++ lub Python pod systemem Linux oraz tworzenia skryptów w bash'u, znajomość języka Java

Cele kształcenia

Zapoznanie studentów z projektowaniem i tworzeniem programów wieloprocesowych i wielowątkowych oraz analizą ich działania.

Treści programowe

- Klasyczne problemy programowania współbieżnego: wzajemne wykluczanie, producent-konsument, pięciu filozofów
- Synchronizacja, unikanie blokady i głodzenia procesów
- Wnioskowanie o procesach współbieżnych - elementy logiki temporalnej
- Mechanizmy komunikacji i synchronizacji procesów w systemie Unix/Linux: łącza, kolejki, semaforey, pamięć dzielona
- Programowanie rozproszone przy użyciu gniazd
- Mechanizmy koordynacji wątków
- Monitory - wysokopoziomowy mechanizm synchronizacji wątków na przykładzie Javy

Wykaz literatury

1. M. Ben-Ari, Podstawy programowania współbieżnego i rozproszonego, WNT, 1996.
2. J.S. Gray, Komunikacja między procesami w UNIXie, RM, 1998
3. Z.J. Czech, Wprowadzenie do obliczeń równoległych, Wydawnictwo Naukowe PWN, 2013.
4. K. Barteczko, JAVA Uniwersalne techniki programowania, PWN, 2015

Kierunkowe efekty uczenia się

K_W05: ma ogólną wiedzę na temat różnych paradygmatów programowania i języków programowania; szczegółowo zna metody i wzorce projektowania i programowania obiektowego
 K_W07: zna najważniejsze elementy architektury systemów komputerowych oraz zasady działania systemów operacyjnych ze szczególnym uwzględnieniem współbieżności, szeregowania zadań i zarządzania pamięcią oraz procesami
 K_W10: zna podstawowe zasady bezpieczeństwa i higieny

Wiedza

Student zna klasyczne problemy i pojęcia z programowania współbieżnego, wybrane techniki programistyczne służące do tworzenia programów współbieżnych oraz wybrane narzędzia teoretyczne służące do wnioskowania o ich poprawności
 Efekty przedmiotowe
 P_W1: zna klasyczne pojęcia i problemy programowania współbieżnego (K_W05, K_W07)
 P_W2: zna mechanizmy komunikacji i synchronizacji procesów w systemie Unix/Linux (K_W07, K_W10)
 P_W3: zna mechanizmy tworzenia i koordynacji wątków (K_W05, K_W07)

Umiejętności

<p>pracy w zawodzie informatyka</p> <p>K_U02: potrafi precyzyjnie formułować pytania, służące pogłębieniu własnego zrozumienia danego tematu lub odnalezieniu brakujących elementów rozumowania</p> <p>K_U06: potrafi projektować, tworzyć, uruchamiać i testować programy przy wykorzystaniu dedykowanych narzędzi oraz adekwatnych wzorców</p> <p>K_U07: korzysta z zaawansowanych funkcjonalności systemów operacyjnych, w szczególności związanych z aspektami sieciowymi</p> <p>K_K01: zna ograniczenia własnej wiedzy i rozumie potrzebę dalszego uczenia się</p> <p>K_K03: potrafi i jest gotów formułować opinie na temat podstawowych zagadnień informatycznych</p>	<p>Student potrafi zaprojektować i zbudować nieduży program współbieżny przy użyciu narzędzi dostępnych w systemie Linux oraz ocenić jego poprawność.</p> <p>Efekty przedmiotowe</p> <p>P_U1: potrafi zaprojektować i zbudować nieduży program współbieżny przy użyciu narzędzi komunikacji i synchronizacji dostępnych w systemie Linux (K_U06, K_U07)</p> <p>P_U2: potrafi ocenić poprawność zbudowanego przez siebie programu współbieżnego i uzasadnić swoją ocenę (K_U02, K_U06)</p>
	<p>Kompetencje społeczne (postawy)</p> <p>Student jest w stanie współpracować z zespołem programistów tworzących wspólnie program współbieżny. Potrafi operować pojęciami w zakresie programowania współbieżnego w sposób umożliwiający mu porozumienie się z innymi informatykami w tym zakresie.</p> <p>Efekty przedmiotowe</p> <p>P_K1: potrafi formułować wypowiedzi na temat programowania współbieżnego i rozumie konieczność dalszego kształcenia się (K_K01, K_K03)</p>
<p>Kontakt</p> <p>pawel.paczkowski@inf.ug.edu.pl</p>	