



IV. Opis programu studiów

3. KARTA PRZEDMIOTU

Kod przedmiotu	E-ID1G-06-s5
Nazwa przedmiotu	Programowanie współbieżne
Nazwa przedmiotu w języku angielskim	Concurrent Programming
Obowiązuje od roku akademickiego	2019/20

USYTUOWANIE MODUŁU W SYSTEMIE STUDIÓW

Kierunek studiów	
Poziom kształcenia	
Profil studiów	
Forma i tryb prowadzenia studiów	
Zakres	
Jednostka prowadząca przedmiot	
Koordynator przedmiotu	Dr inż. Paweł Paduch
Zatwierdził	Dziekan Wydziału Elektrotechniki Automatyki i Informatyki Dr hab. inż. Antoni Różowicz, prof. PŚk

OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU

Przynależność do grupy/bloku przedmiotów	
Status przedmiotu	
Język prowadzenia zajęć	
Usytuowanie modułu w planie studiów - semestr	
Wymagania wstępne	
Egzamin (TAK/NIE)	
Liczba punktów ECTS	

Forma prowadzenia zajęć	wykład	ćwiczenia	laboratorium	projekt	Inne
Liczba godzin w semestrze	30	0	30	0	0

EFEKTY UCZENIA SIĘ

Kategoria	Symbol efektu	Efekty kształcenia	Odniesienie do efektów kierunkowych
Wiedza	W01	Znajomość problematyki programowania współbieżnego	INF1_W09
	W02	Znajomość sposobów synchronizacji i komunikacji między procesami i wątkami	INF1_W09
	W03	Poznanie zasad analizy programów współbieżnych pod kątem efektywności i bezpieczeństwa	INF1_W09
	W04	Znajomość wybranych algorytmów współbieżnych	INF1_W09
Umiejętności	U01	Umiejętność programowania współbieżnego	INF1_U16

TREŚCI PROGRAMOWE

Forma zajęć*	Treści programowe
wykład	1. Wstęp do programowania współbieżnego
	2. Semafore i bufore
	3. Procesy, sygnały i łącza
	4. Monitory i zmienne warunkowe
	5. Kolejki komunikatów, semafore i pamięć dzielona IPC
	6. Wątki i muteksy
	7. Metody synchronizacji Posix
	8. Algorytmy współbieżne, programowanie współbieżne w środowisku .NET
	9. Programowanie współbieżne w środowisku .NET
	10. Synchronizacja, obiekty niezmiennie
	11. Współbieżny dostęp do kolekcji
	12. Synchronizatory
	13. Anulowanie zadań
	14. Wzorce dotyczące współbieżności
	15. Kolokwium zaliczeniowe
laboratorium	1. Podstawy systemu Linux
	2. Procesy
	3. Łącza PIPE
	4. Łącza FIFO
	5. Kolejki komunikatów IPC_MSG
	6. Semafore
	7. Pamięć dzielona
	8. Wątki i muteksy
	9. Programowanie współbieżne w C# (cz. 1)
	10. Programowanie współbieżne w C# (cz. 2)
	11. Synchronizacja, obiekty niezmiennie
	12. Współbieżny dostęp do kolekcji
	13. Synchronizatory
	14. Anulowanie zadań
	15. Wzorce dotyczące współbieżności

*) zostawić tylko realizowane formy zajęć

METODY WERYFIKACJI EFEKTÓW UCZENIA SIĘ

Symbol efektu	Metody sprawdzania efektów uczenia się					
	Egzamin ustny	Egzamin pisemny	Kolokwium	Projekt	Sprawozdanie	Inne
W01			X			
W02			X			
W03			X			

W04			X			
U01			X			X

A.

FORMA I WARUNKI ZALICZENIA

Forma zajęć*	Forma zaliczenia	Warunki zaliczenia
wykład		Uzyskanie co najmniej 50% punktów z kolokwium zaliczeniowego.
laboratorium		Uzyskanie co najmniej 50% punktów z realizacji zadań laboratoryjnych oraz kolokwium.

*) zostawić tylko realizowane formy zajęć

NAKŁAD PRACY STUDENTA

Bilans punktów ECTS							
Lp.	Rodzaj aktywności	Obciążenie studenta					Jednostka
		W	C	L	P	S	
1.	Udział w zajęciach zgodnie z planem studiów	30		30			h
3.	Inne (konsultacje, egzamin)*	4		2			h
4.	Razem przy bezpośrednim udziale nauczyciela akademickiego	66					h
5.	Liczba punktów ECTS, którą student uzyskuje przy bezpośrednim udziale nauczyciela akademickiego	2,64					ECTS
6.	Liczba godzin samodzielnej pracy studenta	59					h
7.	Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy	2,36					ECTS
8.	Nakład pracy związany z zajęciami o charakterze praktycznym	30					h
9.	Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym	2,27					ECTS
10.	Sumaryczne godzinowe obciążenie pracą studenta	125					h
11.	Punkty ECTS za moduł <i>1 punkt ECTS=25 godzin obciążenia studenta</i>	5					

* wszelkie formy weryfikacji efektów, w tym egzaminy oraz nie więcej niż 2 godziny konsultacji dla każdej formy zajęć

LITERATURA

1. M. Ben-Ari, Podstawy programowania współbieżnego i rozproszonego, WNT 2009
2. W. Richard Stevens – Programowanie zastosowań sieciowych, WNT 1995
3. W. Iszkowski M. Maniecki. - Programowanie współbieżne, WNT 1982
4. Andrew Troelsen, Język C# 6.0 i platforma .NET 4.6, PWN 2017
5. Mateusz Warczak - Programowanie równoległe i asynchroniczne w C# 5.0, Helion
6. Herbert Schildt, Java. Kompendium programisty. Wydanie X, Helion, 2019
7. Herbert Schildt, Java. Przewodnik dla początkujących. Wydanie VII, Helion, 2018
8. Cay S. Horstmann, Java 9. Przewodnik doświadczonego programisty. Wydanie II, Helion, 2018
9. Goetz B., Peierls T., Bloch J., Bowbeer J., Holmes D., Lea D., Java Współbieżność dla praktyków, Helion 2007
10. D. C. Schmidt, M. Stal, H. Rohnert, F. Buschmann, Pattern-Oriented Software Architecture: Patterns for Concurrent and Networked Objects, Wiley & Sons, 2000

Uwaga: wykaz literatury winien uwzględniać aktualne i dostępne publikacje