



KARTA PRZEDMIOTU

Kod przedmiotu	studia stacjonarne:	E-ID2C-14-s2, E-ID2C-14-s3
	studia niestacjonarne:	E-2IZ2C-1018-s3
Nazwa przedmiotu	Przetwarzanie odporne na błędy	
Nazwa przedmiotu w języku angielskim	Fault tolerant processing	
Obowiązuje od roku akademickiego	2022/23	

USYTUOWANIE PRZEDMIOTU W SYSTEMIE STUDIÓW

Kierunek studiów	Informatyka
Poziom kształcenia	II stopień
Profil studiów	Ogólnoakademicki
Forma i tryb prowadzenia studiów	Studia stacjonarne i niestacjonarne
Zakres	Cyberbezpieczeństwo
Jednostka prowadząca przedmiot	Katedra Systemów Informatycznych
Koordynator przedmiotu	Dr inż. Grzegorz Łukawski
Zatwierdził	Dziekan Wydziału Elektrotechniki, Automatyki i Informatyki dr hab. inż. Roman Deniziak, prof. PŚk

OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU

Przynależność do grupy/bloku przedmiotów		Przedmiot specjalnościowy
Status przedmiotu		Wybieralny
Język prowadzenia zajęć		Polski
Usytuowanie w planie studiów - semestr	studia stacjonarne	Semestr II lub Semestr III
	studia niestacjonarne	Semestr II lub Semestr III
Wymagania wstępne		brak
Egzamin (TAK/NIE)		NIE
Liczba punktów ECTS		3

Forma prowadzenia zajęć		wykład	ćwiczenia	laboratorium	projekt	inne
Liczba godzin w semestrze	studia stacjonarne:	30		15	15	
	studia niestacjonarne:	18		9	9	

EFEKTY UCZENIA SIĘ

Kategoria	Symbol efektu	Efekty uczenia się	Odniesienie do efektów kierunkowych
Wiedza	W01	Student potrafi wymienić rodzaje błędów występujących w systemach komputerowych, rozumie potrzebę zapewnienia odporności na błędy.	INF2_W02
	W02	Student zna podstawowe mechanizmy tolerowania błędów sprzętowych.	INF2_W06
	W03	Student zna podstawowe mechanizmy tolerowania błędów programowych.	INF2_W06
Umiejętności	U01	Student potrafi zaproponować metodę zapewnienia odporności na błędy dla wybranego problemu sprzętowego lub programowego.	INF2_U01, INF2_U05, INF2_U07, INF2_U08, INF2_U09, INF2_U11
Kompetencje społeczne	K01	Student umie podzielić problem programistyczny na elementy i współpracować w grupie przy jego implementacji.	INF2_K01 INF2_K02

TREŚCI PROGRAMOWE

Forma zajęć	Treści programowe
wykład	1. Wprowadzenie do zagadnień związanych z odpornością na błędy (rodzaje błędów, redundancja, odległość Hamminga, rodzaje systemów odpornych na błędy).
	2. Metody detekcji błędów sprzętowych (bity parzystości, kod Bergera, kody cykliczne, CRC).
	3. Metody korekcji błędów sprzętowych (kod Hamminga, kody Reeda-Solomona).
	4. Maskowanie błędów (redundancja bierna, dynamiczna i hybrydowa, JCT, TMR, NMR).
	5. Systemy rozproszone odporne na błędy (niezawodna komunikacja, protokoły spójności, transakcje).
	6. Metody detekcji i korekcji błędów programowych (programowanie alternatywne i wielowersyjne, wyjątki, watchdog, journaling).
laboratorium	1. Metody tolerowania błędów sprzętowych – symulacja programowa.
	2. Metody tolerowania błędów programowych.
projekt	Zadanie projektowe polegające na przygotowaniu aplikacji realizującej wybrane algorytmy odporności na błędy programowe lub sprzętowe, na drodze symulacji.

METODY WERYFIKACJI EFEKTÓW UCZENIA SIĘ

Symbol efektu	Metody sprawdzania efektów uczenia					
	Egzamin ustny	Egzamin pisemny	Kolokwium	Projekt	Sprawozdanie	Inne
W01			X			
W02			X			
W03			X			
U01			X			X
K01				X	X	

FORMA I WARUNKI ZALICZENIA

Forma zajęć	Forma zaliczenia	Warunki zaliczenia
wykład	zaliczenie z oceną	Uzyskanie co najmniej 50% punktów z kolokwium zaliczeniowego.
laboratorium	zaliczenie z oceną	Uzyskanie co najmniej 50% punktów za wykonane zadania laboratoryjne oraz z kolokwium końcowego, kolokwium w trakcie zajęć lub kolokwium końcowego i kolokwium w trakcie zajęć.
projekt	zaliczenie z oceną	Uzyskanie co najmniej 50% punktów za wykonane zadanie projektowe oraz sprawozdanie.

NAKŁAD PRACY STUDENTA

Bilans punktów ECTS												
Lp.	Rodzaj aktywności	Obciążenie studenta										Jednos tka
		studia stacjonarne					studia niestacjonarne					
1.	Udział w zajęciach zgodnie z planem studiów	W	C	L	P	S	W	C	L	P	S	h
		30		15	15		18		9	9		
2.	Inne (konsultacje, egzamin)	2		2	2		2		2	2		h
3.	Razem przy bezpośrednim udziale nauczyciela akademickiego	66					42					h
4.	Liczba punktów ECTS, którą student uzyskuje przy bezpośrednim udziale nauczyciela akademickiego	2,64					1,68					ECTS
5.	Liczba godzin samodzielnej pracy studenta	9					33					h
6.	Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy	0,36					1,32					ECTS
7.	Nakład pracy związany z zajęciami o charakterze praktycznym	30					18					h
8.	Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym	1,2					0,72					ECTS
9.	Sumaryczne obciążenie pracą studenta	75					75					h
10.	Punkty ECTS za moduł <i>1 punkt ECTS=25 godzin obciążenia studenta</i>	3										ECTS

LITERATURA

1. Sapięcha K.: Testowanie i diagnostyka systemów cyfrowych, PWN, Warszawa 1987.
2. Sosnowski J.: Testowanie i niezawodność systemów komputerowych, Akademicka Oficyna Wydawnicza EXIT, Warszawa 2005.
3. Tanenbaum Andrew S.: Strukturalna organizacja systemów komputerowych, Helion 2006.