

Karta przedmiotu

Nazwa i kod przedmiotu	Laboratorium programistyczne, PG_00178065						
Kierunek studiów	Informatyka i ekonometria (O)						
Data rozpoczęcia studiów	październik 2025 r.		Rok akademicki realizacji przedmiotu		2026/2027		
Poziom kształcenia	I stopnia - licencjackie		Grupa zajęć		Grupa zajęć obowiązkowych z zakresu kierunku studiów Grupa zajęć fakultatywnych Grupa zajęć powiązanych z prowadzonymi badaniami naukowymi w dziedzinie nauki związanej z kierunkiem - profil ogólnoakademicki		
Forma studiów	stacjonarne		Sposób realizacji		na uczelni		
Rok studiów	2		Język wykładowy		polski		
Semestr studiów	4		Liczba punktów ECTS		3.0		
Profil kształcenia	ogólnoakademicki		Forma zaliczenia		zaliczenie		
Jednostka prowadząca	Rektor -> Wydział Zarządzania -> Katedra Informatyki Ekonomicznej						
Imię i nazwisko wykładowcy (wykładowców)	Odpowiedzialny za przedmiot		mgr inż. Dawid Jereczek				
	Prowadzący zajęcia z przedmiotu						
Formy zajęć	Forma zajęć	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium	RAZEM
	Liczba godzin zajęć	0.0	0.0	45.0	0.0	0.0	45
	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	45		3.0		27.0	75
Cel przedmiotu	Celem przedmiotu jest rozwijanie praktycznych umiejętności programistycznych studentów oraz pogłębianie wiedzy zdobytej w ramach wcześniejszych kursów z zakresu algorytmiki i programowania. W trakcie zajęć laboratoryjnych studenci samodzielnie projektują, implementują i testują programy komputerowe, rozwiązując złożone problemy z obszaru informatyki.						

Efekty uczenia się przedmiotu	Efekt kierunkowy	Efekt z przedmiotu	Sposób weryfikacji i oceny efektu
	[IiEL3_U12] Student potrafi projektować i implementować systemy informatyczne wspierające działalność przedsiębiorstw oraz wykorzystywać nowoczesne technologie ICT w zarządzaniu i komunikacji biznesowej.	Student potrafi projektować i implementować proste oraz złożone aplikacje komputerowe wspierające działalność przedsiębiorstw. Student umie wykorzystywać nowoczesne języki programowania, biblioteki oraz narzędzia ICT w realizacji praktycznych projektów informatycznych. Student potrafi integrować systemy informatyczne z bazami danych i innymi źródłami danych, wspierając procesy decyzyjne i analityczne w organizacjach. Student umie stosować dobre praktyki inżynierii oprogramowania (wersjonowanie kodu, testowanie, dokumentowanie), zapewniające wysoką jakość tworzonych rozwiązań.	[SU5] realizacja zadania problemowego [SU6] demonstracja umiejętności praktycznych [SU8] obserwacja samodzielnej lub zespołowej pracy studenta
	[IiEL3_U02] Student potrafi dobrać lub konstruować narzędzia ekonometryczne, informatyczne lub statystyczne oraz stosować je do opisu i rozwiązywania problemów ekonomicznych i społecznych.	Student potrafi dobrać odpowiednie narzędzia programistyczne oraz technologie informatyczne do rozwiązywania konkretnych problemów ekonomicznych i społecznych. Student umie wykorzystywać języki programowania oraz biblioteki analityczne do realizacji zadań z zakresu modelowania i prognozowania zjawisk ekonomicznych. Student umie projektować i implementować aplikacje wspomagające analizy ekonomiczne oraz przetwarzanie i wizualizację danych.	[SU5] realizacja zadania problemowego [SU6] demonstracja umiejętności praktycznych [SU8] obserwacja samodzielnej lub zespołowej pracy studenta
Treści przedmiotu	Zaawansowane techniki programowania strukturalnego i obiektowego: projektowanie i implementacja bardziej złożonych aplikacji z wykorzystaniem paradygmatów OOP (obiektość, hermetyzacja, dziedziczenie, polimorfizm). Projektowanie i implementacja struktur danych: listy dynamiczne, stosy, kolejki priorytetowe, drzewa (w tym drzewa BST, AVL), grafy (reprezentacje grafowe, algorytmy przeszukiwania grafów). Integracja aplikacji z relacyjnymi bazami danych: projektowanie modeli danych, obsługa transakcji, realizacja operacji CRUD, wykorzystanie zapytań parametryzowanych. Złożoność i efektywność algorytmiczna: profilowanie i optymalizacja kodu, analiza zużycia pamięci i czasu wykonania. Przetwarzanie i analiza danych: parsowanie danych z różnych formatów (JSON, XML, CSV), przetwarzanie danych nieustrukturyzowanych (tekst), operacje na zbiorach danych. Realizacja projektu zespołowego: samodzielne lub grupowe opracowanie projektu aplikacji wspierającej rzeczywisty proces lub problem biznesowy lub ekonomiczny od analizy wymagań, przez implementację i testowanie, po prezentację gotowego rozwiązania.		
Wymagania wstępne i dodatkowe	Podstawowa znajomość algorytmów i struktur danych Znajomość podstawowych pojęć z zakresu baz danych i SQL Umiejętność korzystania z podstawowych środowisk programistycznych (IDE) Podstawowa wiedza z zakresu inżynierii oprogramowania (wersjonowanie, testowanie, dokumentowanie kodu)		

Sposoby i kryteria oceniania osiągniętych efektów uczenia się	Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa oceny końcowej
	Projekt semestralny	51.0%	50.0%
	Zadania laboratoryjne	51.0%	50.0%
Zalecana lista lektur	Podstawowa lista lektur	Piotr Wróblewski, Algorytmy, struktury danych i techniki programowani, Helion Steve McConnell, <i>Code Complete: A Practical Handbook of Software Construction</i> , Microsoft Press. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, <i>Introduction to Algorithms</i> , MIT Press. Materiały wykładowe,	
	Uzupełniająca lista lektur	Martin Fowler, <i>Refactoring: Improving the Design of Existing Code</i> , Addison-Wesley. Stack Overflow, GitHub Discussions jako źródła wiedzy społecznościowej.	
	Adresy eZasobów		
	Przykładowe zagadnienia/ przykładowe pytania/ realizowane zadania		
Praktyki zawodowe w ramach przedmiotu	Nie dotyczy		

Dokument wygenerowany elektronicznie. Nie wymaga pieczęci ani podpisu.