



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 funkcyjne		11.1.0569	
Nazwa jednostki prowadzącej przedmiot			
Instytut Informatyki			
Studia			
wydział	kierunek	poziom	drugiego stopnia, pierwszego stopnia
Wydział Matematyki, Fizyki i Informatyki	Modelowanie matematyczne i analiza danych	forma	stacjonarne
		moduł	wszystkie
		specjalnościowy	wszystkie
		specjalizacja	wszystkie
Nazwisko osoby prowadzącej (osób prowadzących)			
prof. UG, dr hab. Christoph Schwarzweller			
Formy zajęć, sposób ich realizacji i przypisana im liczba godzin		Liczba punktów ECTS	
Formy zajęć		5	
Wykład, Ćw. audytoryjne			
Sposób realizacji zajęć			
zajęcia w sali dydaktycznej			
Liczba godzin			
Wykład: 30 godz., Ćw. audytoryjne: 30 godz.			
Termin realizacji przedmiotu			
2021/2022 zimowy			
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"> - Projektowanie doświadczeń - Wykonywanie doświadczeń - Zajęcia w laboratorium komputerowym 		Sposób zaliczenia	
		<ul style="list-style-type: none"> - Zaliczenie na ocenę - Egzamin 	
		Formy zaliczenia	
		<ul style="list-style-type: none"> - zaliczenie ustne - egzamin pisemny z pytaniami (zadaniami) otwartymi - kolokwium - egzamin pisemny (dłuższa wypowiedź pisemna / rozwiązanie problemu) 	
		Podstawowe kryteria oceny	
Sposób weryfikacji założonych efektów uczenia się			
Egzamin i kolokwium oraz obserwacje studenta			
Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi			
A. Wymagania formalne			
Brak			
B. Wymagania wstępne			
Brak			
Cele kształcenia			
Celem przedmiotu jest wprowadzenie do programowania funkcyjnego na podstawie języka Haskell.			

Treści programowe	
<ol style="list-style-type: none"> 1. Wprowadzenie 2. Wstęp do języka programowania Haskell 3. Typy i klasy typów w Haskell 4. Podstawy programowania funkcyjnego 5. Techniki i zastosowania programowania funkcyjnego 	
Wykaz literatury	
<ul style="list-style-type: none"> • Hutton; Programming in Haskell • Tompson; Haskell: The Craft of Programming • Bird; Introduction to Functional Programming • Abelson, Sussman; Structure and Interpretation of Computer Programs 	
Kierunkowe efekty uczenia się	Wiedza
	<p>Student zna i rozumie:</p> <ul style="list-style-type: none"> • paradygmat programowania funkcyjnego • model programowania funkcyjnego • język programowania Haskell i jego system typów
	Umiejętności
	<p>Student potrafi:</p> <ul style="list-style-type: none"> • projektować algorytmy z wykorzystaniem paradygmatu programowania funkcyjnego • rozwiązać problemy używając języka funkcyjnego
	Kompetencje społeczne (postawy)
	<p>Student:</p> <ul style="list-style-type: none"> • wykorzystuje angielską literaturę fachową • zna prawo autorskie związane z pisaniem programów
Kontakt	
christoph.schwarzweiler@ug.edu.pl	