



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 obiektowo-funkcyjne		11.3.1344					
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 Wiesław Pawłowski; dr Andrzej Borzyszkowski; dr Tomasz Borzyszkowski							
Formy zajęć, sposób ich realizacji i przypisana im liczba godzin		Liczba punktów ECTS					
Formy zajęć		4 15 godz. wykl., 30 godz. ćw. lab, praca własna studenta					
Wykład, Ćw. laboratoryjne							
Sposób realizacji zajęć							
zajęcia w sali dydaktycznej							
Liczba godzin							
Wykład: 15 godz., Ćw. laboratoryjne: 30 godz.							
Termin realizacji przedmiotu							
2020/2021 letni							
Status przedmiotu		Język wykładowy					
obowiązkowy		polski					
Metody dydaktyczne		Forma i sposób zaliczenia oraz podstawowe kryteria oceny lub wymagania egzaminacyjne					
<ul style="list-style-type: none"> - Projektowanie doświadczeń - Wykład konwersatoryjny - Wykład z prezentacją multimedialną - ćwiczenia laboratoryjne -- sporządzanie i uruchamianie programów komputerowych 		Sposób zaliczenia					
		Zaliczenie na ocenę					
		Formy zaliczenia					
		<ul style="list-style-type: none"> - zaliczenie ustne - ustalenie oceny zaliczeniowej na podstawie ocen cząstkowych otrzymywanych w trakcie trwania semestru 					
Podstawowe kryteria oceny							
Sposób weryfikacji założonych efektów kształcenia							
zakładany efekt kształcenia	egzamin	kolokwium	projekt	referat	raport	aktywność	obserwacja postawy i umiejętności
	Wiedza						
K_W04			X				
	Umiejętności						
K_U04			X				X
K_U08			X				X
K_U09			X				X
	Kompetencje						
K_K02							X
Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi							
A. Wymagania formalne							

Brak wymagań formalnych	
B. Wymagania wstępne Brak wymagań wstępnych	
Cele kształcenia Podstawowym celem jest zapoznanie studentów z najważniejszymi koncepcjami funkcyjnego podejścia do programowania oraz połączenie ich z mechanizmami strukturalizacji wywodzącymi się z podejścia obiektowego. Pozwala to na poszerzenie i pogłębienie wiedzy już posiadanej przez studentów oraz ułatwia im dokonywanie doboru odpowiednich technik programowania w zależności od charakteru zadania. Projekty realizowane podczas zajęć laboratoryjnych służą praktycznemu ugruntowaniu koncepcji poznanych podczas wykładu	
Treści programowe Podstawowe cechy języka Scala Typy podstawowe, wyrażenia, zmienne i wartości (niemutowalność) Funkcje, rekurencja, funkcje wyższego rzędu System/hierarchia typów i generyczność Kolekcje i ich wykorzystywanie Mechanizmy obiektowe – obiekty, klasy, cechy, dziedziczenie, polimorfizm Konstruktory i wzorce oraz ich wykorzystanie Aktorzy Akka jak przykład wykorzystania modelu obiektowo-funkcyjnego	
Wykaz literatury A. Literatura wymagana do ostatecznego zaliczenia zajęć (zdania egzaminu): Brak literatury obowiązkowej/wymaganej B. Literatura uzupełniająca: Martin Odersky, Lex Spoon, Bill Venner. Programming in Scala, 3rd edition, Artima Press, 2016. Harold Abelson, Gerald Jay Sussman, Julie Sussman. Structure and Interpretation of Computer Programs, 2nd edition, The Massachusetts Institute of Technology 1996-2016.	
Kierunkowe efekty kształcenia K_W04 ma uporządkowaną, podbudowaną teoretycznie wiedzę w zakresie programowania, algorytmów i złożoności, języków i paradygmatów programowania K_U04 potrafi tworzyć, uruchamiać i testować programy przy wykorzystaniu dedykowanych narzędzi oraz wzorców projektowych K_U08 ocenia przydatność różnych paradygmatów i narzędzi programistycznych do rozwiązywania problemów różnego typu K_U09 potrafi zgodnie z zadaną specyfikacją zaprojektować oraz zrealizować system informatyczny K_K02 potrafi precyzyjnie formułować pytania, służące pogłębieniu własnego zrozumienia danego tematu lub odnalezieniu brakujących elementów rozumowania	Wiedza ma uporządkowaną, podbudowaną teoretycznie wiedzę w zakresie programowania, algorytmów i złożoności, języków i paradygmatów programowania Umiejętności potrafi tworzyć, uruchamiać i testować programy przy wykorzystaniu dedykowanych narzędzi oraz wzorców projektowych ocenia przydatność różnych paradygmatów i narzędzi programistycznych do rozwiązywania problemów różnego typu potrafi zgodnie z zadaną specyfikacją zaprojektować oraz zrealizować system informatyczny Kompetencje społeczne (postawy) potrafi precyzyjnie formułować pytania, służące pogłębieniu własnego zrozumienia danego tematu lub odnalezieniu brakujących elementów rozumowania
Kontakt W.Pawlowski@inf.ug.edu.pl	