



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		11.3.0163	
Nazwa jednostki prowadzącej przedmiot			
Instytut Informatyki			
Studia			
wydział	kierunek	poziom	drugiego stopnia
Wydział Matematyki, Fizyki i Informatyki	Informatyka	forma	stacjonarne
		moduł	wszystkie
		specjalnościowy	wszystkie
		specjalizacja	wszystkie
Nazwisko osoby prowadzącej (osób prowadzących)			
prof. UG, dr hab. Christoph Schwarzweller; dr Stefan Sokołowski			
Formy zajęć, sposób ich realizacji i przypisana im liczba godzin		Liczba punktów ECTS	
Formy zajęć		6 Przedmiot w formie 30h wykładu i 30h laboratorium + praca własna studenta.	
Wykład, Ćw. laboratoryjne			
Sposób realizacji zajęć			
zajęcia w sali dydaktycznej			
Liczba godzin			
Wykład: 30 godz., Ćw. laboratoryjne: 30 godz.			
Cykl dydaktyczny			
2017/2018 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 		Sposób zaliczenia	
		<ul style="list-style-type: none"> - Zaliczenie na ocenę - Egzamin 	
		Formy zaliczenia	
		<ul style="list-style-type: none"> - egzamin pisemny z pytaniami (zadaniami) otwartymi - ustalenie oceny zaliczeniowej na podstawie ocen cząstkowych otrzymywanych w trakcie trwania semestru - egzamin pisemny (dłuższa wypowiedź pisemna / rozwiązanie problemu) 	
		Podstawowe kryteria oceny	
		kolokwium po laboratorium pisemny egzamin	
Sposób weryfikacji założonych efektów kształcenia			
Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi			
A. Wymagania formalne			
brak			
B. Wymagania wstępne			
brak			
Cele kształcenia			
Wprowadzenie do programowania funkcyjnego na podstawie języka Haskell			
Treści programowe			

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

Hutton; Programming in Haskell
 Tompson; Haskell: The Craft of Programming
 Bird; Introduction to Functional Programming
 Abelson, Sussman; Structure and Interpretation of Computer Programs

Efekty kształcenia (obszarowe i kierunkowe)

Student:
 K_W02 ma pogłębioną wiedzę na temat podstawowych paradygmatów programowania; zna również aktualne trendy w językach programowania
 K_W04 zna formalne modele obliczeń a także ich własności i znaczenie w praktycznych zastosowaniach informatycznych, ma wiedzę na temat barier obliczalności i trudności obliczeń
 K_W05 Zna najważniejsze konstrukcje programistyczne oraz struktury danych
 K_W06 zna zaawansowane metody projektowania i analizowania złożoności obliczeniowej algorytmów, zna zasady działania oraz praktycznego zastosowania najważniejszych algorytmów różnego typu w sensie ich treści jak i sposobu ich wykonywania
 K_W09 zna biegle co najmniej dwa języki programowania oraz biblioteki algorytmów i struktur danych oraz ma wiedzę na temat praktycznych uwarunkowań wydajnych implementacji algorytmów
 K_W10 zna metody algorytmicznego rozwiązywania problemów obliczeniowo trudnych (aproksymacja, szybkie algorytmy wykładnicze, heurystyki)
 K_W12 zna dobrze zasady bezpieczeństwa i higieny pracy w zawodzie informatyka

K_U02 ma umiejętność projektowania abstrakcyjnych struktur danych i ich wydajnych implementacji
 K_U06 projektuje, analizuje pod kątem poprawności i złożoności obliczeniowej oraz programuje algorytmy z wykorzystaniem różnych technik programistycznych
 K_U07 potrafi zastosować znane algorytmy w konkretnych sytuacjach, potrafi efektywnie dobrać rodzaj i sposób wykonania algorytmu w zależności od postawionego problemu

Wiedza

- Student:
- zna paradygmat programowania funkcyjnego
 - zna model programowania funkcyjnego
 - zna język programowania Haskell i jego system typów

Umiejętności

- Student:
- projektuje algorytmy z wykorzystaniem paradygmatu programowania funkcyjnego
 - umie rozwiązać problemy używając języka funkcyjnego

Kompetencje społeczne (postawy)

- Student:
- wykorzystuje angielską literaturę fachową
 - zna prawo autorskie związane z pisaniem programów

Kontakt

schwarz@inf.ug.edu.pl