



**KAPITAŁ LUDZKI**  
NARODOWA STRATEGIA SPÓJNOŚCI

Projekt współfinansowany przez  
Unię Europejską w ramach  
Europejskiego Funduszu  
Społecznego

**UNIA EUROPEJSKA**  
EUROPEJSKI  
FUNDUSZ SPOŁECZNY



<b>Nazwa przedmiotu</b>		<b>Kod ECTS</b>	
Programowanie funkcyjne NS		11.3.1306	
<b>Nazwa jednostki prowadzącej przedmiot</b>			
Instytut Informatyki			
<b>Studia</b>			
<b>wydział</b>	<b>kierunek</b>	<b>poziom</b>	<b>drugiego stopnia</b>
Wydział Matematyki, Fizyki i Informatyki	Informatyka	<b>forma</b>	niestacjonarne (zaoczne)
		<b>moduł</b>	wszystkie
		<b>specjalnościowy</b>	wszystkie
		<b>specjalizacja</b>	wszystkie
<b>Nazwisko osoby prowadzącej (osób prowadzących)</b>			
prof. UG, dr hab. Christoph Schwarzweller			
<b>Formy zajęć, sposób ich realizacji i przypisana im liczba godzin</b>		<b>Liczba punktów ECTS</b>	
<b>Formy zajęć</b>		7 Przedmiot w formie 20h wykładu i 20h laboratorium + praca własna studenta.	
Wykład, Ćw. laboratoryjne			
<b>Sposób realizacji zajęć</b>			
zajęcia w sali dydaktycznej			
<b>Liczba godzin</b>			
Wykład: 20 godz., Ćw. laboratoryjne: 20 godz.			
<b>Termin realizacji przedmiotu</b>			
2020/2021 letni			
<b>Status przedmiotu</b>		<b>Język wykładowy</b>	
fakultatywny (do wyboru)		polski	
<b>Metody dydaktyczne</b>		<b>Forma i sposób zaliczenia oraz podstawowe kryteria oceny lub wymagania egzaminacyjne</b>	
<ul style="list-style-type: none"> <li>- Projektowanie doświadczeń</li> <li>- Wykonywanie doświadczeń</li> <li>- Wykład z prezentacją multimedialną</li> </ul>		<b>Sposób zaliczenia</b>	
		<ul style="list-style-type: none"> <li>- Zaliczenie na ocenę</li> <li>- Egzamin</li> </ul>	
		<b>Formy zaliczenia</b>	
		<ul style="list-style-type: none"> <li>- egzamin pisemny z pytaniami (zadaniami) otwartymi</li> <li>- ustalenie oceny zaliczeniowej na podstawie ocen cząstkowych otrzymywanych w trakcie trwania semestru</li> <li>- egzamin pisemny (dłuższa wypowiedź pisemna / rozwiązanie problemu)</li> </ul>	
		<b>Podstawowe kryteria oceny</b>	
		kolokwium po laboratorium pisemny egzamin	
<b>Sposób weryfikacji założonych efektów kształcenia</b>			

zakładany efekt kształcenia	egzamin	kolokwium	projekt	referat	raport	aktywność	obserwacja postawy
Wiedza							
K_W02	x	x					
K_W03	x	x					
K_W04	x	x					
K_W06							x
Umiejętności							
K_U03	x	x					x
K_U05	x	x					x

**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

**Kierunkowe efekty kształcenia**

K\_W02: ma pogłębioną wiedzę w zakresie języków formalnych, modeli obliczeń oraz zagadnień złożoności obliczeniowej; zna aparat formalny pozwalający na formułowanie i badanie własności obiektów informatycznych  
K\_W03: ma pogłębioną wiedzę na temat paradygmatów programowania oraz zaawansowanych konstrukcji programistycznych; zna aktualne trendy w językach programowania  
K\_W04: zna złożone struktury danych oraz zaawansowane metody algorytmicznego rozwiązywania problemów obliczeniowo trudnych (algorytmy wykładnicze, aproksymacja, heurystyki)  
K\_W06: zna dobrze zasady bezpieczeństwa i higieny pracy w zawodzie informatyka  
K\_U03: projektuje, analizuje pod kątem poprawności i złożoności obliczeniowej oraz buduje algorytmy z wykorzystaniem zaawansowanych technik programistycznych i struktur danych  
K\_U05: potrafi zastosować znane algorytmy w konkretnych sytuacjach, potrafi efektywnie dobrać rodzaj 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 pisanie programów

## Kontakt

[schwarz@inf.ug.edu.pl](mailto:schwarz@inf.ug.edu.pl)