



Introduction to Programming Languages

Karta opisu przedmiotu

Informacje podstawowe

Kierunek studiów Informatyka i Systemy Inteligentne	Cykl dydaktyczny 2022/2023	
Specjalność -	Kod przedmiotu EISIS.li100.7ee3ddd63be2cc6d2a34f51cf264e288.22	
Jednostka organizacyjna Wydział Elektrotechniki, Automatyki, Informatyki i Inżynierii Biomedycznej	Języki wykładowe angielski	
Poziom kształcenia Studia inżynierskie I stopnia	Obligatoryjność Do wyboru	
Forma studiów Stacjonarne	Blok zajęciowy Przedmioty ogólne	
Profil studiów Ogólnoakademicki	Przedmiot powiązany z badaniami naukowymi Nie	
Koordynator przedmiotu	Weronika T. Adrian, Krystian Jobczyk	
Prowadzący zajęcia	Mateusz Ślażyński, Krystian Jobczyk	
Okres Semestr 5	Forma zaliczenia Zaliczenie	Liczba punktów ECTS 3
	Forma prowadzenia i godziny zajęć Wykład: 14 Ćwiczenia laboratoryjne: 14	

Cele kształcenia dla przedmiotu

C1	Zapoznać studentów z formalnymi aspektami języków programowania
C2	Przedstawić powiązanie między logiką a systemami typów w językach programowania

Efekty uczenia się dla przedmiotu

Kod	Efekty w zakresie	Kierunkowe efekty uczenia się	Metody weryfikacji
Wiedzy - Student zna i rozumie:			
W1	Student rozumie rolę rachunku lambda w informatyce	ISI1A_W05	Aktywność na zajęciach, Wykonanie ćwiczeń laboratoryjnych, Zaliczenie laboratorium
W2	Student zna różnicę między syntaktycznymi i semantycznymi cechami języków programowania	ISI1A_W05	Aktywność na zajęciach, Wykonanie ćwiczeń laboratoryjnych, Zaliczenie laboratorium
Umiejętności - Student potrafi:			
U1	Student potrafi definiować rekurencyjne obliczenia w rachunku lambda	ISI1A_U06	Wykonanie ćwiczeń laboratoryjnych, Zaliczenie laboratorium
U2	Student potrafi zdefiniować semantykę języka programowania o silnym systemie typów	ISI1A_U01	Wykonanie ćwiczeń laboratoryjnych, Zaliczenie laboratorium
U3	Student potrafi zaimplementować interpreter według formalnego opisu języka	ISI1A_U07	Wykonanie ćwiczeń laboratoryjnych, Zaliczenie laboratorium
Kompetencje społecznych - Student jest gotów do:			
K1	Student potrafi komunikować się w zespołach badawczych i programistycznych przy użyciu formalnego opisu semantyki języków programowania	ISI1A_K05	Aktywność na zajęciach, Wykonanie ćwiczeń laboratoryjnych, Zaliczenie laboratorium

Treści programowe zapewniające uzyskanie efektów uczenia się dla modułu zajęć

Głównym tematem kursu będzie formalny opis języków programowania z zastosowaniem semantyki operacyjnej. Zaczynając od rachunku Lambda, student będzie poznawał kolejne rozszerzenia budujące współczesne języki programowania. Wykładom będą towarzyszyć odpowiednio dobrane zadania programistyczne.

Nakład pracy studenta

Rodzaje zajęć studenta	Średnia liczba godzin* przeznaczonych na zrealizowane aktywności
Wykład	14
Ćwiczenia laboratoryjne	14
Samodzielne studiowanie tematyki zajęć	62
Łączny nakład pracy studenta	Liczba godzin 90
Liczba godzin kontaktowych	Liczba godzin 28

* godzina (lekcyjna) oznacza 45 minut

Treści programowe

Lp.	Treści programowe	Efekty uczenia się dla przedmiotu	Formy prowadzenia zajęć
1.	Wstęp do teorii języków programowania — motywacja, tło historyczne i preliminaria matematyczne	W1, W2, K1	Wykład
2.	Rachunek lambda — znaczenie języka, formalizacja w semantyce operacyjnej oraz jej implementacja	W1, U1, U3	Wykład, Ćwiczenia laboratoryjne
3.	Systemy typów — rozszerzenie rachunku lambda o proste typy	U2, U3, K1	Wykład, Ćwiczenia laboratoryjne
4.	Popularne funkcjonalności języków programowania: algebra typów, makra, zmienne lokalne	U2, U3, K1	Wykład, Ćwiczenia laboratoryjne
5.	Efekty uboczne — formalizacja operacji na pamięci	U2, U3, K1	Wykład, Ćwiczenia laboratoryjne
6.	Podtypy — definiowanie relacji wewnątrz systemu typów	U2, U3, K1	Wykład, Ćwiczenia laboratoryjne

Informacje rozszerzone

Metody i techniki kształcenia:

Mini wykład, Kształcenie zdalne, Metoda projektowa (Project based learning)

Rodzaj zajęć	Metody zaliczenia	Warunki zaliczenia przedmiotu
Wykład	Aktywność na zajęciach, Zaliczenie laboratorium	Pozytywna ocena z laboratorium.
Ćwiczenia laboratoryjne	Wykonanie ćwiczeń laboratoryjnych	Student zdobywa co najmniej 50% punktów z każdego laboratorium

Dodatkowy opis

Ocena z laboratorium będzie wynikiem pracy na zajęciach i zrealizowanych w ramach ich zadań programistycznych. Każde laboratorium będzie posiadało odrębne zadanie, które będzie podlegało automatycznemu sprawdzeniu przed kolejnymi zajęciami.

Warunki i sposób zaliczenia poszczególnych form zajęć, w tym zasady zaliczeń poprawkowych, a także warunki dopuszczenia do egzaminu

Warunkiem wystarczającym do zaliczenia zajęć jest realizacja wszystkich zadań laboratoryjnych (każdego z osobna) na przynajmniej 50% punktów. W wypadku braku zaliczenia w terminie podstawowym — student będzie mógł poprawić niezaliczone zadania.

Sposób obliczania oceny końcowej

Ocena końcowa będzie średnią z ocen otrzymanych za poszczególne zadania laboratoryjne. Ponadto, ocena może być podniesiona ze względu na aktywność w czasie zajęć.

Sposób i tryb wyrównywania zaległości powstałych wskutek nieobecności studenta na zajęciach

Nieobecności na zajęciach laboratoryjnych będą wymagać realizacji dodatkowych zadań wyznaczonych przez nauczyciela.

Wymagania wstępne i dodatkowe

Wymagana jest podstawowa znajomość języka Python.

Do pełnego zrozumienia kursu zalecane są podstawowa znajomość w obrębie:

1. logiki formalnej
2. języków formalnych
3. kompilatorów

Zasady udziału w poszczególnych zajęciach, ze wskazaniem, czy obecność studenta na zajęciach jest obowiązkowa

Obecność na wykładach nie jest obowiązkowa, ale w wypadku nieobecności student powinien nadrobić braki w oparciu o literaturę. Obecność na laboratoriach jest wymagana.

Literatura

Obowiązkowa

1. Types and Programming Languages (The MIT Press); Benjamin C. Pierce; The MIT Press, 2002

Dodatkowa

1. Advanced Topics in Types and Programming Languages (The MIT Press); Benjamin C. Pierce; The MIT Press, 2004
2. The Little Typer (The MIT Press); Daniel P. Friedman, David Thrane Christiansen, et al.; The MIT Press, 2018
3. The Little Prover (The MIT Press); Friedman, Daniel P., Eastlund, Carl; The MIT Press, 2015

Kierunkowe efekty uczenia się

Kod	Treść
ISI1A_K05	Dostrzega i rozumie konieczność nieustannego doskonalenia swojej wiedzy, umiejętności i kompetencji społecznych.
ISI1A_U01	Potrafi wykorzystać nabytą wiedzę matematyczną do opisu procesów, tworzenia modeli, analizy algorytmów oraz innych działań w obszarze informatyki.
ISI1A_U06	Potrafi algorytmizować wybrane problemy, ocenić ich złożoność obliczeniową, estymować czas wykonania, dobierać właściwe algorytmy do zadanego problemu, stosować metody i techniki Sztucznej Inteligencji.
ISI1A_U07	Potrafi projektować i rozwijać aplikacje z wykorzystaniem poznanych technologii oraz języków programowania. Potrafi doskonalić umiejętności nabyte w trakcie studiów.
ISI1A_W05	Ma uporządkowaną, podbudowaną teoretycznie wiedzę w zakresie języków formalnych, kompilatorów oraz języków programowania.