

KARTA PRZEDMIOTU

1. Informacje ogólne

Nazwa przedmiotu i kod (wg planu studiów):	Języki i paradygmaty programowania C7
Nazwa przedmiotu (j. ang.):	Programming languages and paradigms
Kierunek studiów:	Informatyka
Specjalność/specjalizacja:	Sieciowe systemy informatyczne/Technologie internetowe i bazy danych/Informatyka praktyczna
Poziom kształcenia:	studia I stopnia
Profil kształcenia:	praktyczny (P)
Forma studiów:	studia stacjonarne / studia niestacjonarne
Obszar kształcenia:	nauki techniczne
Dziedzina:	nauki techniczne
Dyscyplina nauki:	Informatyka
Koordinator przedmiotu:	mgr inż. Łukasz Sanokowski

2. Ogólna charakterystyka przedmiotu

Przynależność do modułu:	kształcenia kierunkowego
Status przedmiotu:	obowiązkowy
Język wykładowy:	polski
Rok studiów, semestr:	II, 4
Forma i wymiar zajęć według planu studiów:	stacjonarne - wykład 15 h, ćw. laboratoryjne 15 h niestacjonarne - wykład 15 h, ćw. laboratoryjne 15 h
Interesariusze i instytucje partnerskie (nieobowiązkowe)	
Wymagania wstępne / Przedmioty wprowadzające:	Algorytmy i struktury danych / Programowanie I / Programowanie II

3. Bilans punktów ECTS

Całkowita liczba punktów ECTS (wg planu studiów; 1 punkt =25-30 godzin pracy studenta, w tym praca na zajęciach i poza zajęciami):	3 (A + B)	stacjonarne	Niestacjonarne
A. Liczba godzin wymagających bezpośredniego udziału nauczyciela (kontaktowych, w czasie rzeczywistym, w tym testy, egzaminy etc) z podziałem na typy zajęć oraz całkowita liczba punktów ECTS osiągniętych na tych zajęciach	obecność na wykładach obecność na ćwiczeniach laboratoryjnych udział w konsultacjach w sumie: ECTS	15 15 1 31 1,3	15 15 1 31 1,3
B. Poszczególne typy zadań do samokształcenia studenta (niewymagających bezpośredniego udziału nauczyciela) wraz z planowaną średnią liczbą godzin na każde i sumaryczną liczbą ECTS (np. praca w bibliotece, w sieci, na platformie e-learningowej, w laboratorium, praca nad projektem końcowym, przygotowanie ogólne; suma poszczególnych godzin powinna zgadzać się z liczbą ogólną)	przygotowanie ogólne przygotowanie do ćwiczeń laboratoryjnych wykonanie sprawozdań przygotowanie do kolokwium zaliczeniowego praca w bibliotece praca w sieci w sumie: ECTS	5 15 3 15 5 2 45 1,7	5 15 3 15 5 2 45 1,7
C. Liczba godzin praktycznych/laboratoryjnych w ramach przedmiotu oraz związana z tym liczba punktów ECTS (ta liczba nie musi być powiązana z liczbą godzin kontaktowych, niektóre zajęcia praktyczne/laboratoryjne mogą odbywać się bez udziału nauczyciela):	udział w ćwiczeniach laboratoryjnych praca praktyczna samodzielna w sumie: ECTS	15 20 25 1,0	15 20 25 1,0

4. Opis przedmiotu

Cel przedmiotu:

Osiągnięcie rozszerzonej wiedzy z zakresu języków i paradygmatów programowania, a w szczególności programowania funkcyjnego na przykładzie języka Haskell oraz programowania logicznego na przykładzie języka Prolog.

Metody dydaktyczne: wykład informacyjny, wykład problemowy, ćwiczenia laboratoryjne,

np. podające (wykład), problemowe (konwersatorium, seminarium), aktywizujące (symulacja, metoda przypadków itp.), eksponujące (pokaz, film), praktyczne (ćwiczenia, metoda projektów itp) – pełniejszy wykaz poniżej (prosimy wybrać najstosowniejsze - jedną lub więcej, można dodać własne metody)

Treści kształcenia (w rozbiciu na formę zajęć (jeśli są różne formy) i najlepiej w punktach):

Wykłady / ćwiczenia laboratoryjne:

1. Przegląd i historia paradygmatów programowania. Omówienie popularnych języków programowania. Paradygmat funkcyjny. Wprowadzenie do języka Haskell. / Zapoznanie się ze środowiskiem uruchomieniowym Haskell. Podstawowe operacje. Definiowanie własnych funkcji. Instrukcje warunkowe, tworzenie i cechy list, podstawowe operacje na listach.
2. Wykorzystanie wbudowanych funkcji operujących na listach, listy nieskończone, zakresy danych.
3. Konstruktor list - pobieranie parametrów, instrukcje warunkowe, filtrowanie, liczenie ilości kombinacji liczb, przetwarzanie ciągu znaków za pomocą konstruktora list.
Krotki, porównanie cech wspólnych i różniących krotek i list. Operacje na krotkach, przykłady zastosowania krotek.
4. Typy i klasy typów, polimorfizm, wnioskowanie typów. Konwersja i deklaracja typów. Dopasowanie wzorców.
5. Strażnicy wzorców. Rekurencja - funkcje rekurencyjne, rekurencja zastępująca pętle.
6. Paradygmat logiczny. Historia i przykłady praktycznego zastosowanie języka Prolog. / Wprowadzenie do środowiska Prolog. Budowanie bazy wiedzy, fakty, reguły, predykaty, zapytania, zmienne, rekurencja
7. Zawansowane zastosowanie języka Prolog: kolorowanie map, rozwiązywanie sudoku.

5. Efekty kształcenia i sposoby weryfikacji

<p>Efekty kształcenia (w sumie wymienić ok. od 3 do 9 efektów - podać numery efektów z listy dla danego kierunku/specjalności – opublikowane na stronie uczelni; podać TYLKO te efekty (tam gdzie to możliwe i stosowne w trzech kategoriach, np. kompetencje społeczne mogą nie być realizowane w tym przedmiocie), na których osiągnięcie kładzie się nacisk w ramach przedmiotu, wybrane efekty kierunkowe powinny być bardziej szczegółowo sformułowane niż te dla całej specjalności, tak aby były weryfikowalne – dlatego mają osobne symbole jako efekty przedmiotu)</p>		
Efekt przedmiotu (kod przedmiotu + kod efektu kształcenia)	Student, który zaliczył przedmiot (spełnił minimum wymagań)	Efekt kierunkowy
C7_W01	<p>Wiedza:</p> <p>1. Zna podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu prostych zadań informatycznych z zakresu analizy złożoności obliczeniowej algorytmów, budowy systemów komputerowych, systemów operacyjnych, sieci komputerowych i technologii sieciowych, implementacji języków programowania, grafiki i komunikacji człowiek-komputer, sztucznej inteligencji, baz danych, inżynierii oprogramowania oraz systemów wbudowanych.</p>	K_W08
C7_U01	<p>Umiejętności:</p> <p>1. Ma umiejętność formułowania algorytmów i ich programowania z użyciem przynajmniej jednego z typowych narzędzi.</p>	K_U10
C7_U02	<p>2. Potrafi ocenić, na podstawowym poziomie, przydatność rutynowych metod i narzędzi informatycznych oraz wybrać i zastosować właściwą metodę i narzędzia do typowych zadań informatycznych.</p>	K_U30
C7_K01	<p>Kompetencje społeczne</p> <p>1. Potrafi zadbać o poprawność językową formułowanych wniosków i opinii.</p>	K_K07

Sposoby weryfikacji efektów kształcenia:				
<i>(np. dyskusja, gra dydaktyczna, zadanie e-learningowe, ćwiczenie laboratoryjne, projekt indywidualny/grupowy, zajęcia terenowe, referat studenta, praca pisemna, kolokwium, test zaliczeniowy, egzamin, opinia eksperta zewnętrznego, etc. Dodać do każdego wybranego sposobu symbol zakładanego efektu, jeśli jest ich więcej)</i>				
Lp.	Efekt przedmiotu	Sposób weryfikacji	Ocena formująca	Ocena końcowa
1	C7_W01	kolokwium zaliczeniowe	sprawdzian wiedzy	średnia z ocen formujących
2	C7_U01	kolokwium zaliczeniowe	sprawdzian wiedzy	średnia z ocen formujących
3	C7_U02	kolokwium zaliczeniowe	sprawdzian wiedzy	średnia z ocen formujących
4	C7_K01	ćwiczenia laboratoryjne	ocena na podstawie obserwacji pracy studenta na zajęciach	średnia z ocen formujących
Kryteria oceny (oceny 3,0 powinny być równoważne z efektami kształcenia, choć mogą być bardziej szczegółowo opisane):				
w zakresie wiedzy				Efekt kształcenia
Na ocenę 3,0	Znajomość podstawowych pojęć z paradygmatów i języków programowania.			C7_W01
Na ocenę 5,0	Znajomość podstawowych oraz zaawansowanych pojęć z paradygmatów i języków programowania.			
w zakresie umiejętności				
Na ocenę 3,0	Potrafi napisać program w język Haskell lub Prolog w celu rozwiązania przykładowych problemów.			C7_U01
Na ocenę 5,0	Potrafi rozwiązywać problemy za pomocą języka Haskell lub Prolog za pomocą samodzielnie zaprojektowanych algorytmów.			
Na ocenę 3,0	Zna różnice pomiędzy paradygmatami i językami programowania. Potrafi wskazać które środowiska nadają się			C7_U02

	do rozwiązania konkretnego problemu.	
Na ocenę 5,0	Potrafi wybrać optymalny język programowania i metodę rozwiązania wskazanego problemu.	
w zakresie kompetencji społecznych		
Na ocenę 3,0	Potrafi zaprojektować algorytm w oparciu o tekstowy opis problemu.	C7_K01
Na ocenę 5,0	Potrafi zaprojektować algorytm w oparciu o tekstowy opis problemu, uwzględniając wszystkie szczegóły.	

Kryteria oceny końcowej (zaleca się podział procentowy poszczególnych kryteriów składających się na ocenę końcową, który może współgrać z powyższymi kryteriami: np. aktywność za zajęciach.. %, kolokwia ...%, samodzielne ćwiczenia ...%, laboratoria ... % **ocena z projektu (szczególnie istotna)**- ...%, zajęcia terenowe...%, zaliczenie, egzamin pisemny... %, opinia eksperta zewnętrznego ...% itp.)

kolokwia: 90 %

samodzielne wykonanie ćwiczeń laboratoryjnych: 5%,

aktywność za zajęciach: 5%,

Zalecana literatura

Literatura podstawowa:

1. Learn You a Haskell for Great Good!, No Starch Press Inc, 2011- dostępna pod adresem: <http://learnyouahaskell.com/>
2. Learn Prolog Now!, College Publications, 2006 - dostępna pod adresem: <http://www.learnprolognow.org/>
3. Seven Languages in Seven Weeks, Pragmatic Bookshelf, 2010

Informacje dodatkowe:

Dodatkowe obowiązki prowadzącego wraz z szacowaną całkowitą liczbą godzin:
Przygotowanie do wykładów i ćwiczeń laboratoryjnych – 30 godzin
Konsultacje – 15 godzin
Poprawa sprawozdań z ćwiczeń laboratoryjnych – 10 godzin

Przygotowanie i poprawa kolokwiów zaliczeniowych – 5 godzin
Poprawa prac projektowych – 5 godzin
W sumie: 65 godzin