



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

Projekt współfinansowany przez
Unię Europejską w ramach
Europejskiego Funduszu
Społecznego

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Nazwa przedmiotu		Kod ECTS	
Geometria, grafy, algorytmy		11.3.0718	
Nazwa jednostki prowadzącej przedmiot			
Faculty of Mathematics, Physics and Informatics			
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. Paweł Żyliński			
Formy zajęć, sposób ich realizacji i przypisana im liczba godzin		Liczba punktów ECTS	
Formy zajęć		6 Przegląd zaawansowanych technik algorytmicznych, struktur danych oraz algorytmów na przykładzie zastosowań w geometrii obliczeniowej i teorii grafów.	
Wykład, Ćw. laboratoryjne			
Sposób realizacji zajęć			
zajęcia w sali dydaktycznej			
Liczba godzin			
Ćw. laboratoryjne: 30 godz., Wykład: 30 godz.			
Cykl dydaktyczny			
2017/2018 letni			
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 - Rozwiązywanie zadań - Wykonywanie doświadczeń - Wykład problemowy - Wykład z prezentacją multimedialną - Ćwiczenia laboratoryjne - implementacja wybranych algorytmów 		Sposób zaliczenia	
		<ul style="list-style-type: none"> - Zaliczenie na ocenę - Egzamin 	
		Formy zaliczenia	
		<ul style="list-style-type: none"> - egzamin ustny - egzamin pisemny (dłuższa wypowiedź pisemna / rozwiązanie problemu) - wykonanie pracy zaliczeniowej - wykonanie określonej pracy praktycznej - wykonanie pracy zaliczeniowej - zadania teoretyczne i/lub implementacja wybranych algorytmów 	
		Podstawowe kryteria oceny	
		<p>Laboratorium (na ocenę). Na zaliczenie składa się (ewentualna modyfikacja i) implementacja kilku wybranych algorytmów omawianych na wykładzie oraz/lub rozwiązanie kilku zadań teoretycznych. Każdy program/zadanie punktowane jest niezależnie, a o ostatecznej ocenie decyduje suma uzyskanych punktów (skala ocen zgodnie z Regulaminem Studiów UG, tekst ujednolicony z dn. 7.10.2014, roz. 4, par. 23).</p> <p>Wykład (na ocenę). Egzamin składa się z dwóch części: praktycznej i teoretycznej. Część praktyczną (50% całości) stanowi ocena uzyskana z laboratorium, tj. jej wskaźnik procentowy. Natomiast część teoretyczną (50% całości) stanowi egzamin ustny, który składa się z trzech pytań, przy czym dwa pierwsze pytania są obowiązkowe, a (opcjonalne) trzecie pytanie - na (częstkową) ocenę DB+ lub BDB - jest pytaniem trudniejszym. Lista zagadnień obowiązujących na egzaminie ustnym wraz ze stopniem trudności udostępniona jest wcześniej przez wykładowcę.</p>	

Sposób weryfikacji założonych efektów kształcenia							
zakładany efekt kształcenia	egzamin	kolokwium	aktywność w dyskusji	implementacja algorytmów	obserwacja postawy studenta	wykład konwersatoryjny	rozwiązywanie zadań
Wiedza							
K_W05	x			x			x
K_W06	x			x			x
K_W10	x			x			x
Umiejętności							
K_U01	x			x			x
K_U02				x			x
K_U03	x			x			x
K_U06				x			x
K_U07				x		x	x
Kompetencje							
K_K03			x		x	x	x

Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi

A. Wymagania formalne

- Matematyka dyskretna.
- Algorytmy i struktury danych.
- Teoria grafów i sieci (podstawy).
- Umiejętność programowania.

B. Wymagania wstępne

Brak.

Cele kształcenia

Przegląd zaawansowanych technik algorytmicznych, struktur danych oraz algorytmów na przykładzie zastosowań w geometrii obliczeniowej oraz teorii grafów.

Treści programowe

1. Geometryczne struktury danych.
2. Programowanie dynamiczne.
3. Technika "dziel i zwyciężaj".
4. Technika zmiatania.
5. Algorytmy aproksymacyjne.
6. Schematy aproksymacyjne.
7. Algorytmy parametryzowane.
8. Algorytmy randomizowane.

Wykaz literatury

- M. de Berg, M. van Kreveld, M. Overmars, O. Schwarzkopf. Geometria obliczeniowa: algorytmy i zastosowania. WNT (2007).
- T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein. Wprowadzenie do algorytmów. WNT (2004).
- S. Dasgupta, C.H. Papadimitriou, U.V. Vazirani. Algorytmy. PWN (2012).
- R. Motwani, P. Raghavan. Randomized Algorithms. Cambridge University Press (1995).
- S. Har-Peled. Geometric Approximation Algorithms. AMS (2011).
- F.P. Preparata, M.I. Shamos. Geometria obliczeniowa - wprowadzenie. Helion (2003).
- V.V. Vazirani. Algorytmy aproksymacyjne. WNT (2005).

Efekty kształcenia

(obszarowe i kierunkowe)

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_W10: zna metody algorytmicznego rozwiązywania

Wiedza

Student:

- orientuje się w podstawowych praktycznych problemach z geometrii obliczeniowej oraz teorii grafów;
- potrafi omówić ważniejsze struktury danych, podać ich zastosowania oraz wskazać ich różnice;
- zna różne techniki algorytmiczne i umie je krótko scharakteryzować.

Umiejętności

Student:

<p>problemów obliczeniowo trudnych (aproxymacja, szybkie algorytmy wykładnicze, heurystyki)</p> <p>K_U01 posiada umiejętność konstruowania rozumowań matematycznych</p> <p>K_U02: ma umiejętność projektowania abstrakcyjnych struktur danych i ich wydajnych implementacji</p> <p>K_U03: potrafi wyrażać problemy obliczeniowe w języku matematyki</p> <p>K_U06: projektuje, analizuje pod kątem poprawności i złożoności obliczeniowej oraz programuje algorytmy z wykorzystaniem różnych technik programistycznych</p> <p>K_U07: potrafi zastosować znane algorytmy w konkretnych sytuacjach, potrafi efektywnie dobrać rodzaj i sposób wykonania algorytmu w zależności od postawionego problemu</p> <p>K_K03: potrafi precyzyjnie formułować pytania, służące pogłębieniu własnego rozumowania danego tematu lub odnalezieniu brakujących elementów rozumowania</p>	<ul style="list-style-type: none"> • potrafi zastosować poznane techniki i struktury danych przy rozwiązywaniu problemów algorytmicznych, w tym rozróżnić i wyodrębnić takie własności problemów, które pozwalają na zastosowanie konkretnej techniki; • stara się analizować konstruowane rozwiązania pod względem złożoności obliczeniowej oraz optymalności rozwiązania.
	<p>Kompetencje społeczne (postawy)</p> <p>Student:</p> <ul style="list-style-type: none"> • umie proponować i jasno formułować, bronić, ale także krytycznie patrzeć na własne rozwiązania; • jest otwarty na stawianie pytań i dyskusję, w tym potrafi przyznać się do błędów.
<p>Kontakt</p> <p>zylinski@inf.ug.edu.pl</p>	