



**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>		
Optymalizacja kombinatoryczna [2]		11.3.2106		
<b>Nazwa jednostki prowadzącej przedmiot</b>				
Instytut Informatyki				
<b>Studia</b>				
<b>wydział</b>	<b>kierunek</b>	<b>poziom</b>	<b>pierwszego stopnia</b>	
Wydział Matematyki, Fizyki i Informatyki	Informatyka	<b>forma</b>	stacjonarne	
		<b>moduł</b>	wszystkie	
		<b>specjalnościowy</b>	wszystkie	
		<b>specjalizacja</b>	wszystkie	
<b>Nazwisko osoby prowadzącej (osób prowadzących)</b>				
dr hab. Paweł Żyliński; mgr Radosław Ziemann				
<b>Formy zajęć, sposób ich realizacji i przypisana im liczba godzin</b>		<b>Liczba punktów ECTS</b>		
<b>Formy zajęć</b>		3 Godziny dydaktyczne: 30h Praca własna studenta: 45h Razem: 75h		
Wykład, Ćw. laboratoryjne				
<b>Sposób realizacji zajęć</b>				
zajęcia w sali dydaktycznej				
<b>Liczba godzin</b>				
Wykład: 15 godz., Ćw. laboratoryjne: 15 godz.				
<b>Termin realizacji przedmiotu</b>				
2025/2026 zimowy				
<b>Status przedmiotu</b>		<b>Język wykładowy</b>		
obowiązkowy		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>- Dyskusja</li> <li>- Implementacja wybranych algorytmów</li> <li>- Projektowanie doświadczeń</li> <li>- Rozwiązywanie zadań</li> <li>- Wykonywanie doświadczeń</li> <li>- Wykład problemowy</li> <li>- Wykład z prezentacją multimedialną</li> </ul>		<b>Sposób zaliczenia</b>		
		Zaliczenie na ocenę		
		<b>Formy zaliczenia</b>		
		<ul style="list-style-type: none"> <li>- implementacja wybranych algorytmów</li> <li>- zaliczenie ustne</li> <li>- ustalenie oceny zaliczeniowej na podstawie ocen cząstkowych otrzymywanych w trakcie trwania semestru</li> <li>- kolokwium</li> </ul>		
		<b>Podstawowe kryteria oceny</b>		
		<b>próg zaliczeniowy</b>	<b>składowa oceny końcowej</b>	
		kolokwium	50%	40%
		implementacje	50%	40%
		zaliczenie końcowe	50%	20%
<b>Sposób weryfikacji założonych efektów uczenia się</b>				

zakładany efekt kształcenia	Zaliczenie końcowe	Kolokwium	Implementacja wybranych algorytmów	Obserwacja postawy
Wiedza				
K_W03	X	X	X	
P_W01	X	X	X	
P_W02	X	X	X	
Umiejętności				
K_U01		X	X	
K_U02			X	X
K_U03			X	
K_U06			X	
P_U01		X	X	
P_U02			X	
Kompetencje				
K_K01	X	X	X	X
P_K01	X	X	X	X
P_K02		X	X	X

**Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi**

**A. Wymagania formalne**

- Matematyka dyskretna
- Algebra liniowa z geometrią analityczną
- Algorytmy i struktury danych
- Teoria grafów i sieci (podstawy)
- Umiejętność programowania

**B. Wymagania wstępne**

Brak wymagań wstępnych.

**Cele kształcenia**

Przegląd zaawansowanych metod algorytmicznych na przykładzie zastosowań w teorii grafów, geometrii obliczeniowej, a także innych zagadnień optymalizacyjnych.

**Treści programowe**

- Przepływy w sieciach wraz z zastosowaniami
- Algorytmy aproksymacyjne i schematy aproksymacyjne
- Algorytmy randomizowane
- Technika zamiatania
- Programowanie liniowe

**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)
- N. Megiddo: Linear-time algorithms for linear programming in R3 and related problems. SIAM J. Comput., 12(4), 759–776, 1983
- 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)
- M.M. Sysło, N. Deo i J.S. Kowalik: Algorytmy optymalizacji dyskretnej, PWN (1999)
- V.V. Vazirani: Algorytmy aproksymacyjne, WNT (2005)

**Kierunkowe efekty uczenia się**

K\_W03: ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie algorytmów i struktur danych, języków formalnych, teorii automatów i złożoności obliczeniowej oraz sztucznej inteligencji

**Wiedza**

P\_W01: zna wybrane metody algorytmicznego rozwiązywania problemów obliczeniowo trudnych, np. algorytmy aproksymacyjne, schematy aproksymacyjne, algorytmy randomizowane, programowanie liniowe całkowitoliczbowe (K\_W03)  
P\_W02: zna wybrane techniki algorytmiczne wraz z przykładami zastosowań: np.

<p>K_U01: potrafi zastosować wiedzę matematyczną do formułowania, analizowania i rozwiązywania problemów związanych z informatyką</p> <p>K_U02: potrafi precyzyjnie formułować pytania, służące pogłębieniu własnego zrozumienia danego tematu lub odnalezieniu brakujących elementów rozumowania</p> <p>K_U03: potrafi projektować i analizować algorytmy pod kątem ich poprawności i złożoności obliczeniowej, wykorzystując odpowiednie techniki algorytmiczne i struktury danych</p> <p>K_U06: potrafi projektować, tworzyć, uruchamiać i testować programy przy wykorzystaniu dedykowanych narzędzi oraz adekwatnych wzorców</p> <p>K_K01: zna ograniczenia własnej wiedzy i rozumie potrzebę dalszego uczenia się</p>	<p>technika zmiatania, przepływy w sieciach (K_W03)</p>
	<p><b>Umiejętności</b></p> <p>P_U01: potrafi zastosować poznane techniki przy rozwiązywaniu problemów algorytmicznych, w tym rozróżnić i wyodrębnić takie własności problemów, które pozwalają na zastosowanie konkretnej techniki (K_U01, K_U02, K_U03)</p> <p>P_U02: stara się analizować konstruowane rozwiązania pod względem złożoności obliczeniowej oraz optymalności rozwiązania (K_U01, K_U03)</p>
	<p><b>Kompetencje społeczne (postawy)</b></p> <p>P_K01: umie proponować i jasno formułować, bronić, ale także krytycznie patrzeć na własne rozwiązania (K_K01)</p> <p>P_K02: jest otwarty na stawianie pytań i dyskusję, w tym potrafi przyznać się do błędu (K_K01)</p>
<p><b>Kontakt</b></p> <p>pawel.zylinski@ug.edu.pl</p>	