



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

Projekt współfinansowany przez
Unię Europejską w ramach
Europejskiego Funduszu
Społecznego

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Nazwa przedmiotu		Kod ECTS					
Programowanie liniowe		11.0.0149					
Nazwa jednostki prowadzącej przedmiot							
Instytut Informatyki							
Studia							
wydział	kierunek	poziom	pierwszego 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		3					
Wykład, Ćw. laboratoryjne							
Sposób realizacji zajęć							
zajęcia w sali dydaktycznej							
Liczba godzin							
Ćw. laboratoryjne: 15 godz., Wykład: 15 godz.							
Termin realizacji przedmiotu							
2020/2021 letni							
Status przedmiotu		Język wykładowy					
obowiązkowy		polski					
Metody dydaktyczne		Forma i sposób zaliczenia oraz podstawowe kryteria oceny lub wymagania egzaminacyjne					
<ul style="list-style-type: none"> - Projektowanie doświadczeń - Wykład problemowy - Wykład z prezentacją multimedialną 		Sposób zaliczenia					
		Zaliczenie na ocenę					
		Formy zaliczenia					
		<ul style="list-style-type: none"> - wykonanie pracy zaliczeniowej - projekt lub prezentacja - egzamin pisemny z pytaniami (zadaniami) otwartymi - egzamin pisemny testowy - kolokwium - egzamin pisemny (dłuższa wypowiedź pisemna / rozwiązanie problemu) - wykonanie pracy zaliczeniowej - wykonanie określonej pracy praktycznej 					
		Podstawowe kryteria oceny					
Sposób weryfikacji założonych efektów kształcenia							
zakładany efekt kształcenia	egzamin	kolokwium	projekt	referat	raport	aktywność w dyskusji	obserwacja
Wiedza							
K_W03		X					X
Umiejętności							
K_U02						X	X
K_U03					X	X	X
Kompetencje							
Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi							

<p>A. Wymagania formalne Algorytmy i struktury danych.</p>	
<p>B. Wymagania wstępne</p>	
<p>Cele kształcenia</p> <p>Celem przedmiotu jest zapoznanie studentów z problemami optymalizacyjnymi, których ograniczenia i funkcja celu są w postaci liniowej, i metodami ich rozwiązywania.</p>	
<p>Treści programowe</p> <ol style="list-style-type: none"> 1. Przykłady zagadnień optymalizacyjnych prowadzących do programowania liniowego 2. Postaci główna, standardowa i kanoniczna problemu programowania liniowego. 3. Algorytm sympleks, złożoność obliczeniowa, przykład Klee-Minty'ego. 4. Problem dualny. 5. Wariant całkowito-liczbowy. 6. Wybrane przykłady zagadnień optymalizacyjnych prowadzących do programowania całkowito-liczbowego wraz z omówieniem rozwiązań. 	
<p>Wykaz literatury</p> <ol style="list-style-type: none"> 1. M.M. Sysło, N. Deo i J.S. Kowalik, Algorytmy optymalizacji dyskretnej, PWN, Warszawa 1999. 2. M. Kubale, Introduction to Computational Complexity and Algorithmic Graph Coloring, Wydawnictwo GTN, Gdańsk 1998. 3. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein: Wprowadzenie do algorytmów. Warszawa: Wydawnictwa Naukowo-Techniczne, 2003. ISBN 83-204-2800-9. 	
<p>Kierunkowe efekty kształcenia</p> <p>P6S_W P6S_WG K_W03 ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie algorytmów i struktur danych, języków formalnych, teorii automatów i złożoności obliczeniowej</p> <p>P6S_U P6S_UW K_U02 potrafi projektować i analizować algorytmy pod kątem ich poprawności i złożoności obliczeniowej wykorzystując odpowiednie techniki algorytmiczne i struktury danych</p> <p>K_U03 potrafi zaplanować i wykonać proste obserwacje, wykonać analizy ilościowe oraz formułować na tej podstawie wnioski jakościowe</p>	<p>Wiedza</p> <ul style="list-style-type: none"> • Student zna przykłady problemów optymalizacyjnych z ograniczeniami liniowymi. • Student zna metodę sympleks, jej zastosowanie i ograniczenia. • Student zna przykłady całkowitoliczbowych problemów optymalizacyjnych z ograniczeniami liniowymi. • Student zna przykłady pokrewnych problemów takich jak: programowanie całkowito-liczbowe (ang. integer programming), pakowanie plecaka (ang. knapsack problem), pakowanie pudełek (ang. bin packing), problem przepływu i metody ich rozwiązywania.
	<p>Umiejętności</p> <ul style="list-style-type: none"> • Student potrafi zapisać wskazany problem programowania liniowego wykorzystując poznany formalizm matematyczny. • Student potrafi znaleźć rozwiązanie problemu programowania liniowego korzystając z gotowego oprogramowania (solvera). • Student potrafi wykorzystać narzędzia programowania liniowego do znalezienia przybliżonych rozwiązań problemów pokrewnych takich jak: programowanie całkowito-liczbowe (ang. integer programming), pakowanie plecaka (ang. knapsack problem), pakowanie pudełek (ang. bin packing), problem przepływu.
	<p>Kompetencje społeczne (postawy)</p>
<p>Kontakt</p> <p>zylinski@inf.ug.edu.pl</p>	