

KARTA OPISU MODUŁU KSZTAŁCENIA		
Nazwa modułu/przedmiotu Optymalizacja ciągła		Kod 1010514371010510098
Kierunek studiów Informatyka	Profil kształcenia (ogólnoakademicki, praktyczny) ogólnoakademicki	Rok / Semestr 4 / 7
Ścieżka obieralności/specjalność -	Przedmiot oferowany w języku: polski	Kurs (obligatoryjny/obieralny) obieralny
Stoień studiów: I stopień	Forma studiów (stacjonarna/niestacjonarna) niestacjonarna	
Godziny Wykłady: 16 Ćwiczenia: - Laboratoria: 16 Projekty/seminaria: -		Liczba punktów 4
Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) kierunkowy		(ogólnouczelniany, z innego kierunku) z danego kierunku
Obszar(y) kształcenia i dziedzina(y) nauki i sztuki nauki techniczne		Podział ECTS (liczba i %) 4 100%
Odpowiedzialny za przedmiot / wykładowca:		
dr inż. Wojciech Kotłowski email: Wojciech.Kotlowski@cs.put.poznan.pl tel. 61 6652936 Instytut Informatyki ul. Piotrowo 2, 60-965 Poznań		Odpowiedzialny za przedmiot / wykładowca: dr inż. Robert Susmaga email: Robert.Susmaga@cs.put.poznan.pl tel. 61 6652934 Instytut Informatyki ul. Piotrowo 2, 60-965 Poznań
Wymagania wstępne w zakresie wiedzy, umiejętności, kompetencji społecznych:		
1	Wiedza:	Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę z zakresu: a) algebry liniowej i geometrii (interpretacje wektorów i macierzy, proste operacje na wektorach i macierzach), b) analizy matematycznej (właściwości, przebiegi i pochodne podstawowych funkcji analitycznych).
2	Umiejętności:	Powinien także posiadać umiejętność projektowania, implementowania i testowania prostych programów komputerowych (w dowolnym języku programowania) realizujących podstawowe operacje wektorowo-macierzowe.
3	Kompetencje społeczne	Z kolei w zakresie kompetencji społecznych do pożądanych cech należą: ciekawość poznawcza i wytrwałość w dążeniu do poszerzania swojej wiedzy.
Cel przedmiotu:		
Zaprezentowanie wybranych elementów problematyki optymalizacji, stanowiącej podstawy merytoryczne sprawnego działania rozwiązań współczesnej inżynierii. Przedmiot uczy jak formułować zagadnienia w taki sposób, aby możliwe było ich ulepszenie pod ustalonymi, wybranymi względami, informując też jednoznacznie, kiedy kontynuowanie tego procesu nie przyniesie dalszej poprawy. Prezentowany przedmiot skupia się na klasycznych problemach optymalizacji ciągłej, zajmującej się modelowymi problemami (np. problemem znajdowania wartości minimalnych funkcji kwadratowej), których rozwiązania są relatywnie łatwe dzięki licznie przyjmowanym założeniom, ale które mogą posłużyć do konstruowania rozwiązań bardziej złożonych, praktycznych problemów (np. jak optymalnie dobrać elementy portfela inwestycyjnego). Ich poznanie pozwala na wyrobienie sobie dobrego rozeznania w szeroko rozumianej dziedzinie optymalizacji, zapewniając doskonale podstawy pojęciowe i algorytmiczne do dalszego kształcenia się w tym kierunku.		
Bardziej szczegółowe cele przedmiotu obejmują przekazanie wiedzy niezbędnej do rozwijania umiejętności: a) identyfikowania, formułowania i rozwiązywania problemów optymalizacji ciągłej, b) projektowania i tworzenia programów implementujących prezentowane metody.		
Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia		
Wiedza:		

<p>1. ma rozszerzoną i pogłębioną wiedzę z matematyki przydatną do formułowania i rozwiązywania złożonych zadań informatycznych dotyczących problemów optymalizacji - [K_W1]</p> <p>2. ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie algorytmów i ich złożoności oraz języków programowania w dziedzinie optymalizacji - [K_W4]</p> <p>3. ma szczegółową wiedzę nt. algorytmiki, w zastosowaniu do metod optymalizacji ciągłej - [K_W5]</p> <p>4. ma wiedzę o trendach rozwojowych i najistotniejszych nowych osiągnięciach w informatyce, w szczególności w zakresie optymalizacji, oraz w wybranych pokrewnych dyscyplinach naukowych, wykorzystujących mechanizmy optymalizacji - [K_W6]</p> <p>5. ma podstawową wiedzę o cyklu życia programowych systemów informatycznych implementujących algorytmy optymalizacji - [K_W7]</p>
Umiejętności:
<p>1. potrafi pozyskiwać informacje z literatury oraz innych źródeł w języku ojczystym i angielskim), integrować je, dokonywać ich interpretacji i krytycznej oceny, wyciągać wnioski oraz formułować i wyczerpująco uzasadniać opinie - [K_U1]</p> <p>2. potrafi przygotować, w języku ojczystym i angielskim, dobrze udokumentowane opracowanie problemów z zakresu informatyki - [K_U3]</p> <p>3. potrafi planować i przeprowadzać eksperymenty, w tym pomiary i symulacje komputerowe, interpretować uzyskane wyniki i wyciągać wnioski - [K_U7]</p> <p>4. potrafi wykorzystać do formułowania i rozwiązywania zadań informatycznych metody analityczne, symulacyjne i eksperymentalne - [K_U8]</p> <p>5. potrafi ocenić złożoność obliczeniową algorytmów i problemów optymalizacji ciągłej - [K_U13]</p> <p>6. ma umiejętność formułowania algorytmów i ich programowania z użyciem przynajmniej jednego z popularnych narzędzi - [K_U22]</p>
Kompetencje społeczne:
<p>1. rozumie, że w informatyce wiedza i umiejętności bardzo szybko stają się przestarzałe - [K_K1]</p> <p>2. potrafi odpowiednio określić priorytety służące realizacji określonego przez siebie lub innych zadania - [K_K6]</p>

Sposoby sprawdzenia efektów kształcenia

Efekty kształcenia przedstawione wyżej weryfikowane są w następujący sposób:

Ocena formująca:

- a) w zakresie wykładów:
- na podstawie odpowiedzi na pytania dotyczące materiału omówionego na poprzednich wykładach,
- b) w zakresie laboratoriów / ćwiczeń:
- na podstawie oceny bieżącego postępu realizacji zadań.

Ocena podsumowująca:

- a) w zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez:
- ocenę wiedzy i umiejętności wykazanych na zaliczeniu pisemnym w formie kilku (4-6) zadań (analogicznych do zadań rozwiązywanych na zajęciach); czas przewidziany na zaliczenie to 60-90 minut; aby uzyskać ocenę pozytywną trzeba zdobyć przynajmniej $1 + \lfloor m/2 \rfloor$ (zaokrąglenie w dół) punktów, gdzie m jest punktacją maksymalną (np. aby uzyskać ocenę pozytywną przy $m = 30$ należy zdobyć przynajmniej 16 punktów).
- b) w zakresie laboratoriów / ćwiczeń weryfikowanie założonych efektów kształcenia realizowane jest przez:
- ocenianie ciągle, na każdych zajęciach (odpowiedzi ustne) - premiowanie przyrostu umiejętności posługiwania się poznanymi metodami,
 - ocenę sprawozdania przygotowywanego częściowo w trakcie zajęć, a częściowo po ich zakończeniu.

Treści programowe

Program przedmiotu obejmuje następujące zagadnienia:

Idea optymalizacji, problem optymalizacji matematycznej i jego składowe (funkcja celu i ograniczenia), minima lokalne i globalne, problemy łatwe i trudne. Zbiory i funkcje wypukłe i ich właściwości, programowanie wypukłe. Wektory i podstawowe operacje na wektorach, interpretacja wektorów jako elementów przestrzeni wielowymiarowej. Iloczyn skalarny wektorów, norma wektora, odległość euklidesowa w przestrzeniach wielowymiarowych i jej właściwości. Macierze i podstawowe operacje na macierzach, odwrotności macierzy. Zapisywanie funkcji wielowymiarowych w notacji macierzowo-wektorowej. Gradient funkcji wielowymiarowej i jego interpretacja. Hessian funkcji wielowymiarowej i jego interpretacja. Szeregi Taylora i MacLaurina oraz ich zastosowanie w przybliżaniu funkcji.

Idea metod iteracyjnych w problemach optymalizacji bez ograniczeń. Rząd zbieżności jako charakterystyka metod iteracyjnych. Warunki stopu w metodach iteracyjnych. Aproksymacyjnej metoda Newtona (poszukiwanie miejsc zerowych funkcji jednowymiarowych) jako podstawowa, jednowymiarowa metoda newtonowska. Przykłady niezbieżności (i implikowanego chaosu) metod jednowymiarowych. Optymalizacyjna metoda Newtona (poszukiwanie minimów funkcji jednowymiarowych) jako optymalizacyjny analog metody aproksymacyjnej.

Metoda Newtona-Raphsona jako podstawowa, wielowymiarowa metoda newtonowska. Przykłady niezbieżności (i implikowanego chaosu) metod wielowymiarowych. Inne wielowymiarowe metody newtonowskie jako modyfikacje metody Newtona-Raphsona: uogólniona metoda Newtona, metoda Cauchy'ego, metoda Levenberga-Marquarda i ich podstawowe właściwości. Przykłady zastosowań metody Newtona-Raphsona: metoda najmniejszych kwadratów regresji liniowej i jej równoważność z metodą Newtona-Raphsona; regresja logistyczna w klasyfikacji, rozwiązywanie regresji logistycznej iteracyjnie ważoną metodą najmniejszych kwadratów i jej równoważność z metodą Newtona-Raphsona. Metoda stochastycznego spadku wzdłuż gradientu: ogólna postać, zbieżność, zastosowanie w regresji i klasyfikacji.

Zajęcia laboratoryjne zaplanowane są w formie siedmiu dwugodzinnych spotkań (plus jednego jednogodzinnego), i polegają na samodzielnym rozwiązywaniu zadań i problemów związanych zagadnieniami poruszonymi na wykładach oraz projektowaniu, kodowaniu i testowaniu implementacji komputerowych tych rozwiązań.

Metody dydaktyczne:

1. wykład: prezentacja multimedialna, ilustrowana przykładami i zadaniami
2. zajęcia laboratoryjne: rozwiązywanie zadań i implementowanie ich rozwiązań, wykonywanie eksperymentów, dyskusja rezultatów i wyników

Literatura podstawowa:

1. W. Findeisen, J. Szymanowski, A. Wierzbicki: Teoria i metody obliczeniowe optymalizacji, PWN, Warszawa, 1977
2. W. Grabowski: Programowanie matematyczne, PWE, Warszawa, 1980
3. R. Wit: Metody programowania nieliniowego, WNT, Warszawa, 1986

Literatura uzupełniająca:

1. D. Bertsekas: Nonlinear Programming, Athena Scientific, Nashua, USA, 1999
2. Z. Jędrzejczak, J. Skrzypek, K. Kukuła, A. Walkost: Badania operacyjne w przykładach i zadaniach, PWN, Warszawa, 2002
3. S. Boyd, L. Vandenberghe: Convex Optimization, Cambridge University Press, 2004

Bilans nakładu pracy przeciętnego studenta

Czynność	Czas (godz.)
1. 1. udział w zajęciach laboratoryjnych / ćwiczeniach:	16
2. 2. przygotowanie do zajęć laboratoryjnych:	16
3. 3. dokończenie (w ramach pracy własnej) sprawozdania z zajęć laboratoryjnych	20
4. 4. udział w konsultacjach związanych z realizacją procesu kształcenia, w szczególności zajęć laboratoryjnych	2
5. 5. udział w wykładach: 15 x 1 godz.	16
6. 6. przygotowanie do zaliczenia	22
7. 7. obecność na zaliczeniu	2
8. 8. omówienie wyników zaliczenia	2

Obciążenie pracą studenta

forma aktywności	godzin	ECTS
Łączny nakład pracy	96	4
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	38	1
Zajęcia o charakterze praktycznym	52	2