

| KARTA OPISU MODUŁU KSZTAŁCENIA | | |
|---|--|--|
| Nazwa modułu/przedmiotu Algorytmy ewolucyjne i metaheurystyczne | | Kod 1010512311010510201 |
| Kierunek studiów Informatyka | Profil kształcenia (ogólnoakademicki, praktyczny) ogólnoakademicki | Rok / Semestr 1 / 1 |
| Ścieżka obieralności/specjalność Inteligentne technologie informatyczne | Przedmiot oferowany w języku: polski | Kurs (obligatoryjny/obieralny) obligatoryjny |
| Stopień studiów: II stopień | Forma studiów (stacjonarna/niestacjonarna) stacjonarna | |
| Godziny Wykłady: 15 Ćwiczenia: - Laboratoria: - Projekty/seminaria: 15 | Liczba punktów 2 | |
| Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) kierunkowy | (ogólnouczelniany, z innego kierunku) ogólnouczelniany | |
| Obszar(y) kształcenia i dziedzina(y) nauki i sztuki nauki techniczne nauki techniczne | Podział ECTS (liczba i %) 2 100% 2 100% | |
| Odpowiedzialny za przedmiot / wykładowca: dr hab. inż. Andrzej Jaszkiewicz email: andrzej.jaszkiewicz@cs.put.poznan.pl tel. 61 6652933 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 algebry liniowej i geometrii (proste operacje na wektorach i macierzach) i analizy matematycznej (różniczkowanie prostych funkcji). |
| 2 | Umiejętności: | Powinien także posiadać umiejętność projektowania, implementowania i testowania prostych programów komputerowych (w dowolnym języku) 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, a także spora doza uczciwości i kultury osobistej. |
| Cel przedmiotu: 1. Przekazanie studentom szczegółowej wiedzy dotyczącej algorytmów ewolucyjnych i metaheurystycznych. 2. Rozwijanie u studentów umiejętności a) identyfikowania i formułowania zadań optymalizacyjnych b) projektowania i adaptacji algorytmów ewolucyjnych i metaheurystycznych dla konkretnych problemów b) tworzenia i testowania programów implementujących omawiane metody rozwiązywania przedstawionych problemów. | | |
| Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia | | |
| Wiedza: 1. ma zaawansowaną i pogłębioną wiedzę z zakresu szeroko rozumianych systemów informatycznych, podstaw teoretycznych ich budowania oraz metod, narzędzi i środowisk programistycznych wykorzystywanych do ich implementacji - [K2st_W1] 2. ma uporządkowaną i podbudowaną teoretycznie wiedzę ogólną związaną z kluczowymi zagadnieniami z zakresu algorytmów ewolucyjnych i metaheurystycznych - [K2st_W2] 3. ma zaawansowaną wiedzę szczegółową dotyczącą wybranych zagadnień z zakresu algorytmów ewolucyjnych i metaheurystycznych - [K2st_W3] 4. ma wiedzę o trendach rozwojowych i najistotniejszych nowych osiągnięciach w zakresie algorytmów ewolucyjnych i metaheurystycznych - [K2st_W4] 5. zna zaawansowane metody, techniki i narzędzia stosowane przy rozwiązywaniu złożonych zadań inżynierskich i prowadzeniu prac badawczych w obszarze algorytmów ewolucyjnych i metaheurystycznych - [K2st_W6] | | |
| Umiejętności: | | |

1. potrafi pozyskiwać informacje z literatury, baz danych oraz innych źródeł (w języku polskim i angielskim), integrować je, dokonywać ich interpretacji i krytycznej oceny, wyciągać wnioski oraz formułować i wyczerpująco uzasadniać opinie - [K2st_U1]
2. potrafi planować i przeprowadzać eksperymenty w zakresie algorytmów ewolucyjnych i metaheurystycznych, w tym pomiary i symulacje komputerowe, interpretować uzyskane wyniki i wyciągać wnioski oraz formułować i weryfikować hipotezy związane ze złożonymi problemami inżynierskimi i prostymi problemami badawczymi - [K2st_U3]
3. potrafi wykorzystać do formułowania i rozwiązywania zadań inżynierskich i prostych problemów badawczych metody analityczne, symulacyjne oraz eksperymentalne w zakresie algorytmów ewolucyjnych i metaheurystycznych - [K2st_U4]

Kompetencje społeczne:

1. rozumie, że w informatyce wiedza i umiejętności bardzo szybko stają się przestarzałe - [K2st_K1]
2. rozumie znaczenie wykorzystywania najnowszej wiedzy z zakresu algorytmów ewolucyjnych i metaheurystycznych w rozwiązywaniu problemów badawczych i praktycznych - [K2st_K2]

Sposoby sprawdzenia efektów kształcenia

Efekty kształcenia przedstawione wyżej weryfikowane są w następujący sposób:

Ocena formująca:

- a) w zakresie wykładów:
 - na podstawie aktywności na wykładach i 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 kolokwium pisemnym w formie pytań otwartych i testowych oraz zadań omówienie wyników kolokwium
- 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 zasadami i metodami
 - ocenę sprawozdania przygotowywanego częściowo w trakcie zajęć, a częściowo po ich zakończeniu; ocena ta obejmuje także umiejętność pracy w zespole

Uzyskiwanie punktów dodatkowych za aktywność podczas zajęć, a szczególnie za:

- wykazanie się ciekawymi umiejętnościami ponadprogramowymi,
- omówienia dodatkowych aspektów zagadnienia,
- uwagi związane z udoskonaleniem materiałów dydaktycznych,
- umiejętność współpracy w ramach zespołu praktycznie realizującego zadanie szczegółowe w laboratorium,
- uwagi związane z udoskonaleniem materiałów dydaktycznych.

Treści programowe

W ramach wykładu przedstawiane są następujące zagadnienia:

- przykłady problemów optymalizacji, rodzaje problemów optymalizacji, źródła trudności i złożoności
- klasyfikacja algorytmów optymalizacji
- metody dokładne, pełen przegląd, metody typu podziału i ograniczeń
- heurystyki konstrukcyjne, algorytmy zachłanne, algorytm GRASP, metody typu regret
- lokalne przeszukiwanie, pojęcie sąsiedztwa, wymagania wobec sąsiedztwa, rodzaje lokalnego przeszukiwania, efektywna implementacja algorytmów lokalnego przeszukiwania, efektywna ocena ruchów, wykorzystanie poprzednich ocen ruchów (cache), ruchy kandydackie, wykorzystanie dodatkowych ograniczeń
- lokalne przeszukiwanie z różnych punktów startowych - multiple start local search
- lokalne przeszukiwanie ze zmiennym sąsiedztwem - variable neighborhood local search
- iteracyjne przeszukiwanie lokalne - iterated local search
- adaptacyjne lokalne przeszukiwanie z różnych punktów startowych ? adaptive multistart local search
- przeszukiwanie tabu,
- symulowane wyżarzanie, algorytm wielkiej powodzi - great deluge
- algorytmy genetyczne i ewolucyjne, operatory ewolucyjne, zarządzanie populacją, metody selekcji, zapobieganie przedwczesnej zbieżności, hybrydowe algorytmy ewolucyjne, twierdzenie o schematach
- inne algorytmy populacyjne
- hiperheurystyki
- podstawy teoretyczne algorytmów metaheurystycznych, twierdzenie No Free Lunch, cechy problemów optymalizacji, globalna wypukłość, korelacja jakoś-odległość ? fitness distance correlation, autokorelacja, związki pomiędzy optymalizacją a uczeniem maszynowym

| | | |
|---|---------------|---------------------|
| <ul style="list-style-type: none"> - wykorzystanie analizy cech problemów w projektowaniu algorytmów optymalizacji, np. dobór operatorów sąsiedztwa, projektowanie operatorów rekombinacji - ocena eksperymentalna algorytmów optymalizacji, kryteria oceny, metodyka eksperymentów, analiza statystyczna - wielokryterialne algorytmy metaheurystyczne, zastosowanie wielokryterialnych algorytmów metaheurystycznych w procesie rozwiązywania problemu decyzyjnego, algorytmy oparte na lokalnym przeszukiwaniu, algorytmy ewolucyjne oparte na rankingu Pareto, algorytmy ewolucyjne oparte na dekompozycji/skalaryzacji, hybrydowe algorytmy ewolucyjne (treść opcjonalna) - ocena wielokryterialnych algorytmów metaheurystycznych, relacje zbiorów rozwiązań, miary jakości (treść opcjonalna) <p>W ramach ćwiczeń laboratoryjnych studenci będą pracować samodzielnie lub w niewielkich grupach dwuosobowych. Studenci będą implementować wybrane algorytmy spośród algorytmów poznanych na wykładach dla wybranych problemów optymalizacji. Studenci będą także wykonywać eksperymenty obliczeniowe i oceniać jakość i efektywność zaimplementowanych algorytmów. Wyniki ćwiczeń będą opisywane w systematycznie przygotowywanych sprawozdaniach.</p> | | |
| Literatura podstawowa: | | |
| <ol style="list-style-type: none"> 1. Jarosław Arabas, Wykłady z algorytmów ewolucyjnych, WNT, 2006. 2. Zbigniew Michalewicz, Algorytmy genetyczne + struktury danych = programy ewolucyjne, Helion, 2003. 3. Z. Michalewicz, Jak to Rozwiązać, czyli Nowoczesna Heurystyka, WNT, 2006 | | |
| Literatura uzupełniająca: | | |
| <ol style="list-style-type: none"> 1. Jaskiewicz A., Distance preserving recombination operator for earth observation satellites operations scheduling, Journal of Mathematical Modelling and Algorithms, Volume 7, Issue 1, March 2008, Pages 25-42. 2. Lust, T., Jaskiewicz, A., Speed-up techniques for solving large-scale biobjective TSP, 2010, Computers and Operations Research, 37(3), pp. 521-533. | | |
| Bilans nakładu pracy przeciętnego studenta | | |
| Czynność | | Czas (godz.) |
| 1. udział w zajęciach laboratoryjnych / ćwiczeniach | | 15 |
| 2. przygotowanie do ćwiczeń laboratoryjnych | | 12 |
| 3. dokończenie (w ramach pracy własnej) sprawozdań z ćwiczeń laboratoryjnych | | 2 |
| 4. udział w konsultacjach związanych z realizacją procesu kształcenia, w szczególności ćwiczeń laboratoryjnych / projektu (częściowo mogą być realizowane drogą elektroniczną) | | 2 |
| 5. napisanie sprawozdań i stworzenie oprogramowania (czas poza zajęciami laboratoryjnymi) | | 5 |
| 6. napisanie sprawozdań i stworzenie oprogramowania (czas poza zajęciami laboratoryjnymi) | | 15 |
| 6. udział w wykładach | | 1 |
| 7. omówienie wyników kolokwium | | 1 |
| 8. przygotowanie do kolokwium | | 4 |
| Obciążenie pracą studenta | | |
| forma aktywności | godzin | ECTS |
| Łączny nakład pracy | 56 | 2 |
| Zajęcia wymagające bezpośredniego kontaktu z nauczycielem | 33 | 1 |
| Zajęcia o charakterze praktycznym | 21 | 1 |