



## KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Metody optymalizacji [S1MwT1>MO]

### Przedmiot

Kierunek studiów

Matematyka w technice

Rok/Semestr

3/6

Studia w zakresie (specjalność)

–

Profil studiów

ogólnoakademicki

Poziom studiów

pierwszego stopnia

Język oferowanego przedmiotu

polski

Forma studiów

stacjonarne

Wymagalność

obligatoryjny

### Liczba godzin

Wykład

30

Laboratorium

30

Inne (np. online)

0

Ćwiczenia

0

Projekty/seminaria

0

### Liczba punktów ECTS

4,00

### Koordynatorzy

dr inż. Karol Gajda

karol.gajda@put.poznan.pl

### Wykładowcy

### Wymagania wstępne

Student rozpoczynający ten przedmiot powinien posiadać wiedzę i umiejętności kursu programowania liniowego i kwadratowego. Powinien znać ograniczenia własnej wiedzy i rozumieć potrzebę dalszego kształcenia.

### Cel przedmiotu

Prezentacja wybranych metody optymalizacji, algorytmów grafowych, szeregowania zadań oraz przepływów w sieciach.

### Przedmiotowe efekty uczenia się

Wiedza:

1. ma rozszerzoną i pogłębioną wiedzę z różnych działów matematyki wyższej oraz szczegółową wiedzę dotyczącą zastosowań metod i narzędzi matematycznych w naukach technicznych [K\_W01 (P6S\_WG)]
2. ma poszerzoną i pogłębioną wiedzę dotyczącą modelowania matematycznego [K\_W02 (P6S\_WG)]
3. ma uporządkowaną wiedzę dotyczącą terminologii z zakresu matematyki i wybranych zagadnień z obszaru nauk technicznych związanych z kierunkiem studiów, również w języku obcym [K\_W03 (P6S\_WG)]

4. ma uporządkowaną i podbudowaną teoretycznie wiedzę z informatyki, w tym z metod numerycznych; zna co najmniej jeden pakiet oprogramowania lub język programowania [K\_W06 (P6S\_WG)]

Umiejętności:

1. potrafi posługiwać się wiedzą z matematyki wyższej [K\_U01 (P6S\_UW)]
2. potrafi budować i analizować proste modele matematyczne [K\_U02 (P6S\_UW)]
- +3. potrafi wykorzystywać narzędzia i metody matematyczne, w tym numeryczne do rozwiązywania problemów inżynierskich [K\_U03 (P6S\_UW)]
4. potrafi skonstruować algorytm rozwiązania prostego zadania inżynierskiego oraz zaimplementować i przetestować go w wybranym środowisku programistycznym [K\_U04 (P6S\_UW)]
5. umie posługiwać się językiem obcym w stopniu wystarczającym do porozumiewania się, a także czytania ze zrozumieniem tekstów matematycznych, dokumentacji technicznych oraz podobnych dokumentów [K\_U13 (P6S\_UK)]
- +6. potrafi pracować indywidualnie i w zespole; umie oszacować czas potrzebny na realizację zleconego zadania; potrafi opracować i zrealizować harmonogram prac zapewniający dotrzymanie terminu [K\_U14 (P6S\_UK)]
7. potrafi samodzielnie planować i realizować samokształcenie w celu podnoszenia i aktualizacji swoich kompetencji [K\_U15 (P6S\_UU)]

Kompetencje społeczne:

1. ma świadomość poziomu swojej wiedzy w odniesieniu do prowadzonych badań w naukach ścisłych i technicznych [K\_K01 (P6S\_KK)]
2. ma świadomość pogłębiania i poszerzania wiedzy do rozwiązywania nowopowstałych problemów technicznych [K\_K02 (P6S\_KK)]
3. potrafi pracować zespołowo; rozumie konieczność systematycznej pracy nad wszelkimi projektami, które mają długofalowy charakter. - [K\_K03 (P6S\_KO)]

### Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Wiedza nabyta w ramach wykładu jest weryfikowana przez 45-minutowe kolokwium składające się z różnie punktowanych pytań (testowych i otwartych). Próg zaliczeniowy: 50% punktów. Zagadnienia zaliczeniowe, na podstawie których opracowywane są pytania zostaną przekazane studentom na wykładzie poprzedzającym kolokwium, lub przesłane drogą mailową z wykorzystaniem systemu uczelnianej poczty elektronicznej.

Umiejętności nabyte w ramach zajęć laboratoryjnych weryfikowane są podstawie opracowanych projektów oraz kolokwium zaliczeniowego. Próg zaliczeniowy: 50% punktów.

### Treści programowe

Grafy nieskierowane:

- przeszukiwanie w głąb i wszerz;
- najkrótsze ścieżki z jednego źródła.

Grafy skierowane:

- osiągalność z jednego źródła i z wielu źródeł;
- ścieżki skierowane z jednego źródła;
- najkrótsze ścieżki skierowane z jednego źródła;
- wykrywanie cykli skierowanych;
- porządki wierzchołków przy przeszukiwaniu w głąb;
- szeregowanie z ograniczeniami pierwszeństwa;
- sortowanie topologiczne.

Minimalne drzewa rozpinające:

- algorytm Prima;
- algorytm Kruskala;
- algorytm Fredmana-Tarjana.

Algorytmy wyznaczania najkrótszych ścieżek:

- Dijkstry;
- sortowanie topologiczne;
- Bellmana-Forda.

Algorytmy dla sieci przepływowych:

- algorytm Forda-Fulkersona do wyznaczania przepływu maksymalnego.

## Metody dydaktyczne

1) wykłady:

- wykład z prezentacją uzupełniany przykładami podawanymi na tablicy,
- wykład prowadzony w sposób interaktywny z formułowaniem pytań do grupy studentów lub do wskazywanych konkretnych studentów,
- uwzględnia się aktywność studentów w czasie zajęć przy wystawianiu oceny końcowej,
- w trakcie wykładu inicjowanie dyskusji,
- teoria przedstawiana w ścisłym powiązaniu z praktyką,
- teoria przedstawiana w powiązaniu z aktualną wiedzą studentów,
- przedstawianie nowego tematu poprzedzone przypomnieniem treści powiązanych, znanych studentom z innych przedmiotów.

2) laboratorium:

- laboratoria uzupełniane prezentacjami multimedialnymi,
- szczegółowe recenzowanie sprawozdań przez prowadzącego laboratoria i dyskusje nad komentarzami,
- korzystanie z narzędzi umożliwiających studentom wykonanie zadań w domu,
- demonstracje,
- praca w zespołach,
- eksperymenty obliczeniowe.

## Literatura

Podstawowa

1. Sedgewick R., Wayne K., Algorytmy. Wydanie IV, Helion 2012,
2. Kusiak J., Danielewska-Tulecka A., Oprocha P., Optymalizacja. Wybrane metody z przykładami zastosowań, PWN, 2019
3. Horla D., Metody obliczeniowe optymalizacji w zadaniach, WPP, Poznań, 2016
4. Jędrzejczyk Z., Kukuła K., Skrzypek J., Walkosz A., Badania operacyjne w przykładach i zadaniach, PWN, Warszawa, 2016

Uzupełniająca

1. Kincaid D., Cheney W., Analiza numeryczna [Numerical Analysis: Mathematics of Scientific Computing (The Sally Series; Pure and Applied Undergraduate Texts, Vol. 2)], WNT, Warszawa 2006.
2. Cormen T. H., Leiserson C. E., Rivest R. L., Stein C., Wprowadzenie do algorytmów [Introduction to Algorithms], PWN, Warszawa, 2018

## Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	102	4,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	62	2,00
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu)	40	2,00