



# Algorithms for Computationally Hard Problems

## Course description sheet

### Basic information

<b>Field of study</b> Computer Science	<b>Didactic cycle</b> 2022/2023	
<b>Major</b> -	<b>Course code</b> WIINFS.II20.02814.22	
<b>Organisational unit</b> Faculty of Computer Science	<b>Lecture languages</b> Polish	
<b>Study level</b> First-cycle (engineer) programme	<b>Mandatoriness</b> Obligatory	
<b>Form of study</b> Full-time studies	<b>Block</b> General Modules	
<b>Profile</b> General academic	<b>Course related to scientific research</b> Yes	
<b>Course coordinator</b>	Piotr Faliszewski	
<b>Lecturer</b>	Piotr Faliszewski, Marcin Kurdziel, Marek Gajęcki, Bartosz Kusek, Andrzej Kaczmarczyk	
<b>Period</b> Semester 6	<b>Method of verification of the learning outcomes</b> Completing the classes	<b>Number of ECTS credits</b> 3
	<b>Activities and hours</b> Lectures: 14 Laboratory classes: 14 Project classes: 14	

### Course's learning outcomes

Code	Outcomes in terms of	Learning outcomes prescribed to a field of study	Methods of verification
<b>Knowledge - Student knows and understands:</b>			

Code	Outcomes in terms of	Learning outcomes prescribed to a field of study	Methods of verification
W1	Student zna i rozumie pojęcia aproksymacji i algorytmów losowych.	INF1A_W02	Activity during classes
W2	Student zna i rozumie podstawowe pojęcia parametrycznej teorii złożoności obliczeniowej.	INF1A_W02	Activity during classes
<b>Skills - Student can:</b>			
U1	Student potrafi stworzyć program komputerowy rozwiązujący problem dużej złożoności obliczeniowej.	INF1A_U05, INF1A_U07	Completion of laboratory classes
<b>Social competences - Student is ready to:</b>			
K1	Student potrafi zaplanować wdrożenie programu komputerowego w sytuacji niepewności co do optymalnego rozwiązania.	INF1A_K04	Completion of laboratory classes

## Program content ensuring the achievement of the learning outcomes prescribed to the module

W ramach przedmiotu studenci poznają metody rozwiązywania problemów NP-trudnych (algorytmy dokładne, algorytmy klasy FPT, algorytmy aproksymacyjne).

### Student workload

Activity form	Average amount of hours* needed to complete each activity form
Lectures	14
Laboratory classes	14
Project classes	14
Realization of independently performed tasks	32
Examination or final test/colloquium	2
Contact hours	5
<b>Student workload</b>	<b>Hours</b> 81
<b>Workload involving teacher</b>	<b>Hours</b> 42

\* hour means 45 minutes

### Program content

No.	Program content	Course's learning outcomes	Activities
1.	<p>1. Implementacja algorytmów dokładnych (np. dla problemu Vertex Cover; 2 godz.)</p> <p>2. Implementacja algorytmów aproksymacyjnych (np. dla problemu Vertex Cover; 2 godz.)</p> <p>3. Implementacja redukcji do SAT (np. dla Vertex Cover lub dla Graph-k-Coloring) oraz wykorzystanie solwera SAT (2 godz.)</p> <p>4. Implementacja redukcji do ILP (np. dla Vertex Cover lub dla Graph-k-Coloring) oraz wykorzystanie solwera ILP (2 godz.)</p> <p>5. Implementacja algorytmu FPT parametryzowanego szerokością drzewową (2 godz.)</p> <p>6. Zaproponowanie metody rozwiązania oraz implementacja programu rozwiązującego zadany problem obliczeniowy (4 godz.)</p> <p>Oceny wyznaczane są na podstawie aktywności na laboratorium oraz skuteczności i efektywności programu stworzonego w punkcie 6. powyżej.</p>	U1, K1	Laboratory classes, Project classes
2.	<p>1. Wykładnicze algorytmy dla problemów NP-zupełnych (4 godz.) Metody pełnego przeglądu przestrzeni rozwiązań. Optymalizacja metod pełnego przeglądu. Algorytmy randomizowane. Przykłady algorytmów dla problemu spełnialności formuł logicznych.</p> <p>2. Algorytmy aproksymacyjne dla problemów NP-zupełnych (4 godz.) Pojęcie algorytmu aproksymacyjnego. Przykłady algorytmów o stałym współczynniku aproksymacji oraz o współczynniku aproksymacji zależnym od rozmiaru problemu. Pojęcie w pełni wielomianowego schematu aproksymacji.</p> <p>3. Twierdzenie o probabilistycznie weryfikowanych dowodach (1 godz.) Twierdzenie PCP. Równoważność probabilistycznie weryfikowanych dowodów i klasy NP. Znaczenie twierdzenia PCP dla wyznaczania granic aproksymacji problemów NP-zupełnych.</p> <p>4. Parametryczna teoria złożoności obliczeniowej (4 godz.) Klasy parametrycznej złożoności obliczeniowej (FPT, hierarchia <math>W1, W2, \dots</math>). Metody konstrukcji algorytmów FPT (przebieg przestrzeni rozwiązań, programowanie całkowitoliczbowe, kodowanie kolorami, kernelizacja). Przykłady problemów <math>W1</math> i <math>W2</math> trudnych.</p> <p>5. Algorytmy randomizowane (1 godz.) Pojęcie algorytmu randomizowanego. Podstawy matematyczne. Przykłady algorytmów.</p>	W1, W2	Lectures

### Extended information/Additional elements

#### Teaching methods and techniques :

Lectures are held on-line, Lectures, Lectures are held on-line

Activities	Methods of verification	Credit conditions
Lectures	Activity during classes	
Lab. classes	Completion of laboratory classes	
Project classes	Completion of laboratory classes	

### **Conditions and the manner of completing each form of classes, including the rules of making retakes, as well as the conditions for admission to the exam**

Ćwiczenia laboratoryjne zaliczane są na podstawie udziału w zajęciach oraz samodzielnej implementacji programu rozwiązującego zadany problem. Zaliczenia poprawkowe: Realizacja programu rozwiązującego zadany problem.

### **Method of determining the final grade**

Ocena końcowa, zaliczenia ćwiczeń laboratoryjnych w pierwszym terminie jest oceną z ćwiczeń laboratoryjnych, ale nie niższą niż 3.5. W przypadku zaliczenia ćwiczeń laboratoryjnych w terminie późniejszym niż pierwszy, ocena końcowa nie może być wyższa niż 3.0.

### **Manner and mode of making up for the backlog caused by a student justified absence from classes**

Wyrównywanie zaległości spowodowanych nieobecnością na zajęciach: W przypadku nieobecności na zajęciach studenci mogą uczestniczyć w zajęciach dla innej grupy. Jeśli jest to niemożliwe, studenci są zobowiązani opanować materiał ćwiczeń we własnym zakresie.

## **Prerequisites and additional requirements**

Ogólna znajomość matematyki z naciskiem na matematykę dyskretną. Podstawowa wiedza dotycząca: logiki, teorii grafów, metod konstruowania algorytmów. Znajomość materiału przedmiotu Teoria Obliczeń i Złożoności Obliczeniowej.

### **Rules of participation in given classes, indicating whether student presence at the lecture is obligatory**

Wykład: Studenci uczestniczą w zajęciach poznając kolejne treści nauczania zgodnie z sylabusem przedmiotu. Studenci winni na bieżąco zadawać pytania i wyjaśniać wątpliwości. Rejestracja audiowizualna wykładu wymaga zgody prowadzącego. Ćwiczenia laboratoryjne: Studenci wykonują ćwiczenia laboratoryjne zgodnie z materiałami udostępnionymi przez prowadzącego. Student jest zobowiązany do przygotowania się w przedmiocie wykonywanego ćwiczenia, co może zostać zweryfikowane kolokwium w formie ustnej lub pisemnej. Zaliczenie zajęć odbywa się na podstawie zaprezentowania rozwiązania postawionego problemu. Zaliczenie modułu jest możliwe po zaliczeniu wszystkich zajęć laboratoryjnych. Ćwiczenia projektowe: Studenci wykonują prace praktyczne mające na celu uzyskanie kompetencji zakładanych przez sylabus. Ocenie podlega sposób wykonania projektu oraz efekt końcowy.

## **Literature**

### **Obligatory**

1. Literatura podstawowa:
2. 1. Sipser M.: Wprowadzenie do teorii obliczeń. WNT 2009
3. 2. Vazirani V.: Algorytmy aproksymacyjne, WNT 2016
4. Literatura uzupełniająca:
5. 3. Papadimitriou C.H.: Złożoność obliczeniowa. Helion 2012
6. 4. Bovet D.P, Crescenzi P.: Introduction to the theory of complexity. Prentice Hall, 1994
7. 5. Wegener I.: Complexity Theory, Springer, 2005
8. 6. Michalewicz Z., Fogel D.: Jak to rozwiązać czyli nowoczesna heurystyka. WNT 2006
9. 7. Motwani R., Raghavam P.: Randomized Algorithms, Cambridge University Press 1995

## Scientific research and publications

### Publications

1. 1. Piotr Skowron, Piotr Faliszewski, Arkadii M. Slinko: Achieving fully proportional representation: Approximability results. *Artif. Intell.* 222: 67-103 (2015)
2. 2. Piotr Faliszewski, Edith Hemaspaandra, Lane A. Hemaspaandra: The complexity of manipulative attacks in nearly single-peaked electorates. *Artif. Intell.* 207: 69-99 (2014)
3. 3. Robert Bredereck, Jiehua Chen, Piotr Faliszewski, André Nichterlein, Rolf Niedermeier: Prices Matter for the Parameterized Complexity of Shift Bribery. *AAAI 2014*: 1398-1404

## Learning outcomes prescribed to a field of study

Code	Content
INF1A_K04	Potrafi odpowiednio określić priorytety służące realizacji określonego przez siebie lub innych zadania oraz adekwatnie zaplanować pracę.
INF1A_U05	Potrafi wykorzystać poznane metody i modele do tworzenia programów o charakterze użytkowym, a także potrafi adekwatnie wykorzystać znane algorytmy i struktury danych w budowie systemu komputerowego.
INF1A_U07	Potrafi ocenić, dobrać i stosować właściwe metody i narzędzia stosowane przy realizacji zadań związanych z budową systemu komputerowego, potrafi ocenić przydatność i korzystać z dostępnych bibliotek oraz komponentów oprogramowania.
INF1A_W02	Ma szczegółową wiedzę w zakresie podstaw algorytmiki, struktur danych oraz złożoności obliczeniowej, a także w zakresie podstaw teoretycznych budowy wybranych narzędzi i systemów informatycznych.