



**KAPITAŁ LUDZKI**  
NARODOWA STRATEGIA SPÓJNOŚCI

Projekt współfinansowany przez  
Unię Europejską w ramach  
Europejskiego Funduszu  
Społecznego

**UNIA EUROPEJSKA**  
EUROPEJSKI  
FUNDUSZ SPOŁECZNY



<b>Nazwa przedmiotu</b>		<b>Kod ECTS</b>	
Algorytmy i struktury danych		11.1.0346	
<b>Nazwa jednostki prowadzącej przedmiot</b>			
null			
<b>Studia</b>			
<b>wydział</b>	<b>kierunek</b>	<b>poziom</b>	<b>pierwszego stopnia</b>
Wydział Matematyki, Fizyki i Informatyki	Matematyka	<b>forma</b>	stacjonarne
		<b>moduł</b>	matematyka nauczycielska, matematyka
		<b>specjalnościowy</b>	
		<b>specjalizacja</b>	wszystkie
<b>Nazwisko osoby prowadzącej (osób prowadzących)</b>			
dr hab. Błażej Szepietowski; dr Ewa Tyszkowska; prof. UG, dr hab. Andreas Zastrow			
<b>Formy zajęć, sposób ich realizacji i przypisana im liczba godzin</b>		<b>Liczba punktów ECTS</b>	
<b>Formy zajęć</b>		5 Przedmiot w wymiarze 30h wykładu + 30h laboratorium + praca własna studenta	
Wykład, Ćw. laboratoryjne			
<b>Sposób realizacji zajęć</b>			
zajęcia w sali dydaktycznej			
<b>Liczba godzin</b>			
Ćw. laboratoryjne: 30 godz., Wykład: 30 godz.			
<b>Cykl dydaktyczny</b>			
2019/2020 zimowy			
<b>Status przedmiotu</b>		<b>Język wykładowy</b>	
obowiązkowy		polski	
<b>Metody dydaktyczne</b>		<b>Forma i sposób zaliczenia oraz podstawowe kryteria oceny lub wymagania egzaminacyjne</b>	
<ul style="list-style-type: none"> <li>- Wykład z prezentacją multimedialną</li> <li>- Ćwiczenia laboratoryjne w pracowni komputerowej</li> </ul>		<b>Sposób zaliczenia</b>	
		<ul style="list-style-type: none"> <li>- Zaliczenie na ocenę</li> <li>- Egzamin</li> </ul>	
		<b>Formy zaliczenia</b>	
		<ul style="list-style-type: none"> <li>- egzamin pisemny z pytaniami (zadaniami) otwartymi</li> <li>- Oczekuje się aktywnego uczestnictwa w zajęciach, wykonania projektów programistycznych potwierdzających nabycie wskazanych umiejętności oraz opanowanie treści programowych.</li> <li>- ustalenie oceny zaliczeniowej na podstawie ocen cząstkowych otrzymywanych w trakcie trwania semestru</li> </ul>	
		<b>Podstawowe kryteria oceny</b>	
		Aby zaliczyć przedmiot student powinien posiadać wiedzę zawartą w treściach programowych oraz umiejętność jej stosowania do rozwiązywania zadań teoretycznych. Powinien umieć implementować w postaci programów algorytmy omawiane na wykładzie.	
<b>Sposób weryfikacji założonych efektów kształcenia</b>			

zakładany efekt kształcenia	Egzamin	Projekt	Obserwacja postawy studenta
	Wiedza		
K_W10	+		
K_W12			+
	Umiejętności		
K_U09	+		
K_U10		+	
K_U11		+	
K_U12		+	
K_U13		+	
K_U14		+	
	Kompetencje		
K_K03		+	

## Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi

### A. Wymagania formalne

analiza matematyczna, elementarny rachunek prawdopodobieństwa

### B. Wymagania wstępne

Student zna metody obliczania i szacowania sum, podstawowe definicje i notacje związane ze zbiorami, relacjami, grafami i drzewami, podstawowe pojęcia kombinatoryczne: permutacja, kombinacja, itp., definicje i twierdzenia podstawowego rachunku prawdopodobieństwa.

## Cele kształcenia

Celem przedmiotu jest zapoznanie studenta z podstawowymi algorytmami i strukturami danych, metodami dowodzenia poprawności i określania złożoności czasowej algorytmów, konstruowania efektywnych algorytmów

## Treści programowe

1. Podstawowe struktury danych :listy, stosy, kolejki, drzewa. Implementacje przy użyciu tablic i struktur dowiązaniowych.
2. Analiza algorytmów: poprawność semantyczna, złożoność czasowa pesymistyczna i średnia.
3. Sortowania przez porównania. Algorytmy o złożoności kwadratowej, o złożoności liniowo-logarytmicznej ( merge sort i heap sort), o średniej złożoności liniowo-logarytmicznej( quick sort). Twierdzenie o ograniczeniu dolnym pesymistycznej złożoności czasowej.
4. Sortowanie w czasie liniowym.
5. Struktury danych dla operacji słownikowych( wstaw, usuń, szukaj) tablice z haszowaniem, drzewa wyszukiwań binarnych, drzewa czerwono-czarne, B-drzewa.
6. Metody konstruowania efektywnych algorytmów: dziel i zwyciężaj, programowanie dynamiczne, strategia zachłanna.
7. Reprezentacja grafów, przeszukiwanie, problem najkrótszej ścieżki.

## Wykaz literatury

- T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein, Wprowadzenie do algorytmów, WNT 2007,  
L. Banachowski, K. Diks, W. Rytter, Algorytmy i struktury danych, WNT 2004,  
N. Wirth, Algorytmy+ struktury danych= programy, WNT 2006,

## Efekty kształcenia (obszarowe i kierunkowe)

### Wiedza

Student zna:

1. podstawowe struktury danych( listy, stosy, kolejki, drzewa, kopce)
2. algorytmy sortujące przez porównanie o złożoności czasowej kwadratowej( insertion-sort, heap-sort) liniowo-logarytmicznej (merge-sort), o średniej złożoności liniowo-logarytmicznej( r-quick-sort)
3. twierdzenie o ograniczeniu dolnym pesymistycznej złożoności czasowej algorytmów sortujących przez porównania
4. algorytmy sortujące w czasie liniowym (counting-sort, radix-sort, bucket-sort)
5. struktury danych dla operacji słownikowych( wstaw, usuń, szukaj) tablice z haszowaniem, drzewa wyszukiwań binarnych, drzewa czerwono-czarne, B-drzewa
6. metody reprezentacji i przeszukiwania grafów
7. zna podstawy technik obliczeniowych i programowania wspomagających pracę matematyka i rozumie ich ograniczenia K\_W10

8. zna podstawowe zasady bezpieczeństwa i higieny pracy K\_W12

### Umiejętności

Student:

1. potrafi stosować metody dziel i zwyciężaj, programowania dynamicznego, strategii zachłannej do konstruowania efektywnych algorytmów. Umie analizować poprawność semantyczną oraz złożoność czasową algorytmów (pesymistyczna, średnia) przy pomocy technik matematyki wyższej
2. potrafi zaplanować sposób rozwiązania określonego problemu oraz sporządzić poprawny zapis tego rozwiązania, podając ścisłe i precyzyjne uzasadnienia poprawności swoich rozumowań K\_U09
3. potrafi wykorzystywać poznany pakiet oprogramowania lub poznany język programowania do rozwiązywania wybranych zagadnień z poznanych dziedzin K\_U10
4. rozpoznaje problemy, w tym zagadnienia praktyczne, które można rozwiązać algorytmicznie, potrafi dokonać specyfikacji takiego problemu K\_U11
5. umie ułożyć i analizować algorytm zgodny ze specyfikacją i zapisać go w wybranym języku programowania K\_U12
6. potrafi skompilować, uruchomić i testować napisany samodzielnie program komputerowy K\_U13
7. umie wykorzystywać programy komputerowe w zakresie analizy danych K\_U14

### Kompetencje społeczne (postawy)

Student umie pracować zespołowo, rozumie konieczność systematycznej pracy  
K\_K03

### Kontakt

Blazej.Szepietowski@mat.ug.edu.pl