

Karta przedmiotu

Nazwa i kod przedmiotu	Algorytmy i struktury danych, PG_00155275						
Kierunek studiów	Modelowanie matematyczne i analiza danych (O)						
Data rozpoczęcia studiów	październik 2024 r.		Rok akademicki realizacji przedmiotu		2025/2026		
Poziom kształcenia	I stopnia - licencjackie		Grupa zajęć		Grupa zajęć obowiązkowych z zakresu kierunku studiów		
Forma studiów	stacjonarne		Sposób realizacji		na uczelni		
Rok studiów	2		Język wykładowy		polski polski		
Semestr studiów	3		Liczba punktów ECTS		4.0		
Profil kształcenia	ogólnoakademicki		Forma zaliczenia		zaliczenie		
Jednostka prowadząca							
Imię i nazwisko wykładowcy (wykładowców)	Odpowiedzialny za przedmiot		dr hab. Błażej Szepietowski				
	Prowadzący zajęcia z przedmiotu						
Formy zajęć	Forma zajęć	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium	RAZEM
	Liczba godzin zajęć	30.0	0.0	30.0	0.0	0.0	60
	W tym liczba godzin zajęć na odległość: 0.0						
Aktywność studenta i liczba godzin pracy	Aktywność studenta	Udział w zajęciach dydaktycznych, objętych planem studiów		Udział w konsultacjach		Praca własna studenta	RAZEM
	Liczba godzin pracy studenta	60		5.0		35.0	100
Cel przedmiotu	Celem przedmiotu jest zapoznanie studenta z podstawowymi algorytmami i strukturami danych, metodami dowodzenia poprawności i określania złożoności czasowej algorytmów, konstruowania efektywnych algorytmów						

Efekty uczenia się przedmiotu	Efekt kierunkowy	Efekt z przedmiotu	Sposób weryfikacji i oceny efektu
	[MMiADL3_K03] jest gotów do pracy zespołowej; rozumie konieczność systematycznej pracy nad wszelkimi projektami, które mają długofalowy charakter	jest gotów skutecznie współpracować w zespole podczas realizacji zadania projektowego	[SK8] obserwacja samodzielnej lub zespołowej pracy studenta
	[MMiADL3_U09] potrafi wykorzystywać poznany pakiet oprogramowania lub poznany język programowania do rozwiązywania wybranych zagadnień z poznanych dziedzin, w szczególności z analizy matematycznej, algebry liniowej oraz statystyki	potrafi wykorzystywać poznany język programowania do rozwiązywania wybranych problemów obliczeniowych dobierając odpowiednie struktury danych	[SU5] realizacja zadania problemowego
	[MMiADL3_U13] potrafi wykorzystywać programy komputerowe w zakresie analizy danych	potrafi stosować zaimplementowane algorytmy i struktury danych do analizy informacji oraz rozwiązywania problemów obliczeniowych, a także wykorzystywać narzędzia programistyczne ułatwiające taką analizę.	[SU5] realizacja zadania problemowego
	[MMiADL3_W09] zna i rozumie podstawy technik obliczeniowych i programowania, wspomagających pracę matematyka i rozumie ich ograniczenia	zna podstawowe metody konstruowania efektywnych algorytmów oraz podstawowe struktury danych, rozumie ograniczenia stosowania poszczególnych struktur danych i algorytmów	[SW4] test/egzamin - ustny lub pisemny
	[MMiADL3_U11] potrafi ułożyć i analizować algorytm zgodny ze specyfikacją i zapisać go w wybranym języku programowania	potrafi opracować poprawny algorytm dla danego problemu obliczeniowego, przeanalizować jego złożoność czasową oraz zaimplementować go w wybranym języku programowania.	[SU5] realizacja zadania problemowego
	[MMiADL3_U10] potrafi rozpoznać problemy, w tym zagadnienia praktyczne, które można rozwiązać algorytmicznie; potrafi dokonać specyfikacji takiego problemu	potrafi przeanalizować treść zadania, zidentyfikować aspekty obliczeniowe, które można rozwiązać algorytmicznie.	[SU5] realizacja zadania problemowego
	[MMiADL3_U12] potrafi skompilować, uruchomić i testować napisany samodzielnie program komputerowy	stosuje wybrane środowisko programistyczne w procesie implementacji, debugowania i testowania programu	[SU8] obserwacja samodzielnej lub zespołowej pracy studenta
Treści przedmiotu	1. Podstawowe struktury danych: listy, stosy, kolejki, drzewa i ich implementacje. 2. Analiza algorytmów, ich poprawność semantyczna, złożoność czasowa pesymistyczna i średnia. 3. Sortowania przez porównania. Algorytmy o złożoności kwadratowej (insertion-sort), o złożoności liniowo-logarytmicznej (merge-sort, heap-sort), o średniej złożoności liniowo-logarytmicznej (quick-sort). Twierdzenie o ograniczeniu dolnym pesymistycznej złożoności czasowej. 4. Kopce binarne i zastosowania. Kolejka priorytetowa. 5. Sortowanie w czasie liniowym. 6. Struktury danych dla operacji słownikowych (wstaw, usuń, szukaj): tablice z haszowaniem, drzewa wyszukiwań binarnych, drzewa czerwono-czarne, B-drzewa. 7. Metody konstruowania efektywnych algorytmów: dziel i zwyciężaj, programowanie dynamiczne, strategia zachłanna. 8. Nomenklatura przedmiotu w języku angielskim		
Wymagania wstępne i dodatkowe	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.		
Sposoby i kryteria oceniania osiągniętych efektów uczenia się	Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa oceny końcowej
	zadania projektowe	51.0%	50.0%
	Obserwacja postawy studenta	51.0%	0.0%
	kolokwium	51.0%	50.0%
Zalecana lista lektur	Podstawowa lista lektur	T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein, <i>Wprowadzenie do algorytmów</i> , WNT 2007,	

	Uzupełniająca lista lektur	L. Banachowski, K. Diks, W. Rytter, <i>Algorytmy i struktury danych</i> , WNT 2004, N. Wirth, <i>Algorytmy+ struktury danych= programy</i> , WNT 2006,
	Adresy eZasobów	
Przykładowe zagadnienia/ przykładowe pytania/ realizowane zadania	1. Posortuj podany ciąg funkcji według asymptotycznego tempa wzrostu 2. Przedstaw na rysunku kolejne kroki procedury BUILD-MAX-HEAP przekształcającej daną tablicę w kopiec typu MAX. 3. Przedstaw działanie algorytmu QUICKSORT dla danego ciągu wejściowego. 4. Narysuj drzewo BST powstałe przez dodanie do początkowo pustego drzewa węzłów o kluczach.... 5. Zbuduj drzewo kodu Huffmana dla danego alfabetu i oblicz jego koszt 6. Napisz program, który sprawdza poprawność wyrażenia nawiasowego podanego przez użytkownika. 7. Wykaż poprawność algorytmu Sortowania przez wstawianie używając odpowiedniego niezmiennika pętli. 8. Napisz pseudokod funkcji , która rekurencyjnie oblicza wysokość węzła w drzewie binarnym.	
Praktyki zawodowe w ramach przedmiotu	Nie dotyczy	

Dokument wygenerowany elektronicznie. Nie wymaga pieczęci ani podpisu.