



3. KARTA PRZEDMIOTU

Kod przedmiotu	Z-IDN-U-305
Nazwa przedmiotu	Algorytmy i struktury danych
Nazwa przedmiotu w języku angielskim	Algorithms and Data Structures
Obowiązuje od roku akademickiego	2019/2020

USYTUOWANIE MODUŁU W SYSTEMIE STUDIÓW

Kierunek studiów	INŻYNIERIA DANYCH
Poziom kształcenia	I stopień
Profil studiów	Praktyczny
Forma i tryb prowadzenia studiów	Studia niestacjonarne
Zakres	Wszystkie specjalności
Jednostka prowadząca przedmiot	Katedra Informatyki i Matematyki Stosowanej
Koordynator przedmiotu	Dr inż. Marcin Detka
Zatwierdził	Dr hab. inż. Artur Bartosik, prof. PŚk

OGÓLNA CHARAKTERYSTYKA PRZEDMIOTU

Przynależność do grupy/bloku przedmiotów	Przedmiot kierunkowy
Status przedmiotu	Obowiązkowy
Język prowadzenia zajęć	Polski
Usytuowanie modułu w planie studiów - semestr	Semestr III
Wymagania wstępne	Podstawy Informatyki, Bazy danych, Logika
Egzamin (TAK/NIE)	TAK
Liczba punktów ECTS	4

Forma prowadzenia zajęć	wykład	ćwiczenia	laboratorium	projekt	inne
Liczba godzin w semestrze	9		18		

EFEKTY UCZENIA SIĘ

Kategoria	Symbol efektu	Efekty kształcenia	Odniesienie do efektów kierunkowych
Wiedza	W01	Student zna podstawowe metody rozwiązywania problemów o charakterze algorytmicznym, w tym obliczeniowych. Student zna podstawowe struktury danych i ich właściwości w kontekście tworzenia i budowy algorytmów.	ID1_W09
	W02	Student zna zasady dotyczące tworzenia algorytmów sekwencjach oraz rozumie zasadę działania rekurencji i zna zalety i zagrożenia tej metody.	ID1_W09 ID1_W10
Umiejętności	U01	Student posiada umiejętności doboru algorytmów i struktur danych w zależności od rodzaju i złożoności problemu. Student posiada umiejętność doboru metody odpowiedniej dla problemu obliczeniowego.	ID1_U01 ID1_U03 ID1_U09
	U02	Student potrafi formułować algorytmy w języku programowania i dobierać odpowiednie struktury danych.	ID1_U01 ID1_U03 ID1_U09
Kompetencje społeczne	K01	Student rozumie znaczenie doskonalenia nabytej wiedzy i umiejętności z zakresu struktur danych i algorytmów operujących na tych strukturach, potrafi je uzupełniać i doskonalić.	ID1_K01

TREŚCI PROGRAMOWE

Forma zajęć	Treści programowe
wykład	1. Podstawowe zasady analizy algorytmów. Poprawność oraz złożoność obliczeniowa algorytmu pesymistyczna i oczekiwana.
	2. Algorytmy rekurencyjne Zapis algorytmów rekurencyjnych w języku programowania. Zagrożenia rozwiązań rekurencyjnych. Rekurencja jako ukryta pamięć. Derekursywacja.
	3. Algorytmy numeryczne: całkowanie numeryczne, różniczkowanie numeryczne, rozwiązywanie równań nieliniowych, rozwiązywanie układów równań.
	4. Podstawowe struktury danych: stosy, listy, kolejki.
	5. Algorytmy sortowania: sortowanie przez porównania (InsertionSort, QuickSort, MergeSort), proste kolejki priorytetowe (kopce binarne HeapSort), sortowanie pozycyjne. Złożoność problemu sortowania.
	6. Algorytmy przeszukiwania: przeszukiwanie liniowe, przeszukiwanie binarne, transformacja kluczowa (hashing).
	7. Przeszukiwanie tekstów, algorytmy typu brute-force, algorytm-K-M-P, algorytm Boyera i Moore'a, algorytm Rabina i Karpa.
	8. Zaawansowane techniki programowania: programowanie dynamiczne, algorytmy zachłanne.
laboratorium	1. Zapis algorytmów w różnych notacjach. Implementacja w wybranych językach programowania. Ocena poprawności i złożoności obliczeniowej algorytmów.
	2. Implementacja algorytmów rekurencyjnych (silnia, ciąg Fibonacciego, algorytm Euklidesa, wieże Hanoi, inne). Definiowanie rozwiązań problemów z użyciem rekurencji. Śledzenie wywołań rekurencyjnych. Derekursywacja algorytmów.
	3. Implementacja wybranych algorytmów numerycznych. Ocena zbieżności algorytmów oraz dokładności obliczeń. Definiowanie rozwiązań problemów z użyciem algorytmów numerycznych
	4. Implementacja algorytmów sortowania. Generowanie zbiorów testowych dla algorytmów. Pomiar czasu wykonywania algorytmów dla różnych zbiorów testowych.
	5. Implementacja wybranych algorytmów przeszukujących. Porównanie efektywności poszczególnych podejść do problemu przeszukiwania.

	6. Implementacja i testowanie algorytmów przeszukiwanie tekstów. Ocena efektywności algorytmów przeszukujących teksty.
	7. Implementacja algorytmów dynamiczny i zachłannych do wybranych problemów (problem rozcinania pręta, problem plecakowy, inne). Ocena efektywności wybranych rozwiązań.

METODY WERYFIKACJI EFEKTÓW UCZENIA SIĘ

Symbol efektu	Metody sprawdzania efektów kształcenia <i>(zaznaczyć X)</i>					
	Egzamin ustny	Egzamin pisemny	Kolokwium	Projekt	Sprawozdanie	Inne
W01		X	X		X	
W02		X	X		X	
U01		X	X		X	
U02		X	X		X	
K01		X	X		X	

FORMA I WARUNKI ZALICZENIA

Forma zajęć	Forma zaliczenia	Warunki zaliczenia
wykład	egzamin	Uzyskanie 50% punktów z pisemnej pracy zaliczeniowej, której zakres dotyczy zarówno wykładów jak i laboratoriów.
laboratorium	zaliczenie z oceną	Student zdobywa punkty za aktywność na laboratoriach, za wykonanie sprawozdań do wybranych laboratoriów (wg wskazań prowadzącego) oraz za dwa sprawdziany praktyczne przy komputerach. Warunkiem zaliczenia jest uzyskanie 50% punktów.

NAKŁAD PRACY STUDENTA

Bilans punktów ECTS							
Lp.	Rodzaj aktywności	Obciążenie studenta					Jednostka
		W	C	L	P	S	
1.	Udział w zajęciach zgodnie z planem studiów	9		18			h
2.	Inne (konsultacje, egzamin)	2		2			h
3.	Razem przy bezpośrednim udziale nauczyciela akademickiego	31					h
4.	Liczba punktów ECTS, którą student uzyskuje przy bezpośrednim udziale nauczyciela akademickiego	1,2					ECTS
5.	Liczba godzin samodzielnej pracy studenta	69					h
6.	Liczba punktów ECTS, którą student uzyskuje w ramach samodzielnej pracy	2,8					ECTS
7.	Nakład pracy związany z zajęciami o charakterze praktycznym	67					h
8.	Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym	2,7					ECTS
9.	Sumaryczne obciążenie pracą studenta	100					h
10.	Punkty ECTS za moduł <i>1 punkt ECTS=25 godzin obciążenia studenta</i>	4					ECTS

LITERATURA

1. Banachowski L., Diks K.M., Rytter W., *Algorytmy i struktury danych*, Wydawnictwo Naukowe PWN S.A, Warszawa 2019.
2. Cormen T. H., Leiserson C. E., Rivest R. L., Clifford S., *Wprowadzenie do algorytmów*, Wydawnictwo Naukowe PWN S.A, Warszawa 2018.
3. Wróblewski P., *Algorytmy, struktury danych i techniki programowania*, Helion, Gliwice 2003.