

Algorytmy i struktury danych 2 - opis przedmiotu

Informacje ogólne	
Nazwa przedmiotu	Algorytmy i struktury danych 2
Kod przedmiotu	11.3-WK-IDP-ASD2-P-S14_pNadGen3IMGA
Wydział	Wydział Matematyki, Informatyki i Ekonometrii
Kierunek	Inżynieria danych
Profil	ogólnoakademicki
Rodzaj studiów	pierwszego stopnia z tyt. inżyniera
Semestr rozpoczęcia	semestr zimowy 2019/2020

Informacje o przedmiocie	
Semestr	3
Liczba punktów ECTS do zdobycia	2
Typ przedmiotu	obowiązkowy
Język nauczania	polski
Sylabus opracował	dr Florian Fabiś

Formy zajęć					
Forma zajęć	Liczba godzin w semestrze (stacjonarne)	Liczba godzin w tygodniu (stacjonarne)	Liczba godzin w semestrze (niestacjonarne)	Liczba godzin w tygodniu (niestacjonarne)	Forma zaliczenia
Projekt	30	2	-	-	Zaliczenie na ocenę

Cel przedmiotu

Wypracowanie u studentów umiejętności samodzielnego specyfikowania, implementowania i rozwiązywania problemów algorytmicznych z dziedziny przetwarzania danych.

Wymagania wstępne

Student powinien zaliczyć kurs: Algorytmy i struktury danych 1.

Zakres tematyczny

Implementacja i testowanie wybranych algorytmów przetwarzania danych z zakresu tematycznego zrealizowanego na przedmiocie Algorytmy i struktury danych 1. Każdy student w trakcie semestru musi zrealizować dwa projekty. Każdy z projektów polegać będzie na zaimplementowaniu wskazanego przez prowadzącego algorytmu do rozwiązania konkretnego praktycznego zadania, napisaniu do tego programu, przetestowaniu go oraz przedstawieniu dokumentacji zgodnie z zadaną specyfikacją. Nad jednym z tych dwóch projektów studenci będą pracowali w 2-3 osobowych grupach.

Metody kształcenia

Praktyczna realizacja projektów na zadane tematy.

Efekty uczenia się i metody weryfikacji osiągania efektów uczenia się

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Student zna i rozumie podstawowe pojęcia i opis matematyczny wykorzystywany w analizie algorytmów	K_W07	projekt, bieżąca kontrola na zajęciach	Projekt
Student umie przeanalizować zaprojektowane algorytmy pod kątem poprawności i złożoności obliczeniowej.	K_U05	projekt, bieżąca kontrola na zajęciach	Projekt
Student potrafi wprowadzić potrzebne pojęcia i obiekty matematyczne, np. funkcje, relacje, ciągi rekurencyjne, w celu rozwiązania problemu inżynierskiego.	K_K07	projekt, bieżąca kontrola na zajęciach	Projekt
Student zna różnorodne narzędzia informatyczne wspomagające przetwarzanie, analizę danych i wnioskowanie statystyczne.	K_W09	projekt, bieżąca kontrola na zajęciach	Projekt
Student rozumie potrzebę ciągłego podnoszenia swoich kwalifikacji poprzez poszerzanie swojej wiedzy i praktycznych umiejętności.	K_K01	projekt, bieżąca kontrola na zajęciach	Projekt
Student potrafi, zgodnie z zadaną specyfikacją, przeanalizować, zaprojektować oraz zrealizować prostą aplikację, w której stosowane będą wybrane algorytmy, używając właściwie dobranych metod, technik i narzędzi. Student umie przygotować dokumentację wykonanego projektu.	K_U04	projekt, bieżąca kontrola na zajęciach	Projekt

Warunki zaliczenia

Ocena końcowa jest średnią arytmetyczną ocen obu wykonanych projektów. Jednakże warunkiem otrzymania pozytywnej oceny końcowej jest otrzymanie ocen pozytywnych z obu projektów. Ocena za projekt: 40% za dokumentację projektu, 40% za stworzony program, 20% za dobór i realizację zastosowania praktycznego wskazanych algorytmów.

Literatura podstawowa

1. Aho A., Hopcroft J.E., Ullman J.D., : Projektowanie i analiza algorytmów komputerowych, PWN, Warszawa 1983.
2. Aho A., Hopcroft J.E., Ullman J.D. : Data structures and algorithms.
3. Banachowski L., Diks K., Rytter W., Algorytmy i struktury danych, WNT, W-wa 1996.
4. Cormen T.H., Leiserson C.E., Rivest R.L., Wprowadzenie do algorytmów, WNT, W-wa 1997.
5. Kingsley-Hughes A. : Programowanie od podstaw, Helion, Gliwice 2005.
6. Wirth N. : Algorithms and Data Structured, 1985.

Literatura uzupełniająca

1. Knuth D. : Sztuka programowania, t. 1-3, WNT, Warszawa 2001.
2. Błażewicz J. : Złożoność obliczeniowa problemów kombinatorycznych, WNT, Warszawa 1988.
3. P. Wróblewski: Algorytmy, struktury danych i techniki programowania, wyd. II popr., Helion, 2001.
4. Grębosz J. : Symfonia C++, Edition 2000, Kraków 2010.
5. Bloch J. : Java. Efektywne programowanie, Helion, Gliwice 2009.

Uwagi

Zmodyfikowane przez dr Robert Dylewski, prof. UZ (ostatnia modyfikacja: 19-09-2019 10:13)

Wygenerowano automatycznie z systemu SylabUZ