

Teoretyczne podstawy informatyki - opis przedmiotu

Informacje ogólne	
Nazwa przedmiotu	Teoretyczne podstawy informatyki
Kod przedmiotu	11.3-WI-INFP-TPI
Wydział	Wydział Informatyki, Elektrotechniki i Automatyki
Kierunek	Informatyka
Profil	ogólnoakademicki
Rodzaj studiów	pierwszego stopnia z tyt. inżyniera
Semestr rozpoczęcia	semestr zimowy 2021/2022

Informacje o przedmiocie	
Semestr	2
Liczba punktów ECTS do zdobycia	5
Typ przedmiotu	obowiązkowy
Język nauczania	polski
Sylabus opracował	• prof. dr hab. inż. Andrzej Obuchowicz

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
Ćwiczenia	30	2	18	1,2	Zaliczenie na ocenę
Wykład	30	2	18	1,2	Egzamin

Cel przedmiotu

- zapoznanie studentów z informacjami o podstawowych teoretycznych pojęciach dotyczących informatyki, jak algorytm, gramatyka, język i automaty, hierarchia Chomsky'ego,
- nauka podstawowych umiejętności w zakresie wyznaczania klasy złożoności algorytmów, sprawdzania własności stopu, dowodzenia częściowej poprawności,
- zapoznanie studentów z tezą Churcha-Turinga, klasyfikacją problemów algorytmicznych ze względu na ich złożoność.

Wymagania wstępne

Algorytmy i struktury danych, Logika dla informatyków, Podstawy systemów dyskretnych, Analiza matematyczna.

Zakres tematyczny

Wiadomości wstępne: algorytm i jego własności, notacja asymptotyczna. Poprawność algorytmów: algorytm poprawny, poprawność częściowa, własność określoności obliczeń, własność stopu; dowodzenie poprawności częściowej, dowodzenie własności stopu, logika Hoare'a.

Podstawy teorii automatów, gramatyk i języków: hierarchia Chomsky'ego, gramatyki i języki, automaty skończone i wyrażenia regularne, gramatyki bezkontekstowe, automaty ze stosem i języki bezkontekstowe, gramatyki kontekstowe i automaty liniowo-ograniczone, gramatyki i maszyny Turinga.

Prymitywne modele algorytmiczne. Teza Churcha--Turinga. Maszyna Turinga i jej warianty. Maszyna o dostępie swobodnym. Programy licznikowe.

Sprawność algorytmów. Miary efektywności algorytmów. Złożoność przestrzenna i czasowa. Złożoność pesymistyczna i średnia. Dolne i górne ograniczenie złożoności. Złożoność naturalna. Problemy algorytmicznie zamknięte i otwarte, luka algorytmiczna.

Klasyfikacja problemów algorytmicznych. Problemy łatwo-rozwiązywalne i trudno-rozwiązywalne. Klasy problemów algorytmicznych: logarytmiczne, wielomianowe, NP, NP-pełne i wykładnicze. Otwarte problemy związane z klasyfikacją problemów algorytmicznych. Dowodzenie NP-pełności. nierozwiązywalność i nierozstrzygalność.

Algorytmy współbieżne i probabilistyczne. Stała i rozszerzająca się współbieżność. Złożoność iloczynowa. Sieci - współbieżność o stałych połączeniach. Teza o obliczeniach równoległych. Klasa Nicka. Algorytmy RNC. Współbieżność rozproszona. Bezpieczeństwo i żywotność systemów współbieżnych.

Algorytmy probabilistyczne niektórych konwencjonalnych problemów algorytmicznych. Probabilistyczne klasy złożoności.

Metody kształcenia

Wykład: wykład konwencjonalny/tradycyjny.

Ćwiczenia: ćwiczenia rachunkowe.

Efekty uczenia się i metody weryfikacji osiągnięcia efektów uczenia się

Opis efektu	Symbolik efektów	Metody weryfikacji	Forma zajęć
-------------	------------------	--------------------	-------------

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Potrafi oszacować złożoność czasową zadanego algorytmu.	<ul style="list-style-type: none"> • K_W06 • K_U09 	<ul style="list-style-type: none"> • sprawdzian z progami punktowymi 	<ul style="list-style-type: none"> • Wykład
Jest kreatywny w rozwiązywaniu problemów uwzględniając dostępne techniki i środki.	<ul style="list-style-type: none"> • K_W06 • K_U09 	<ul style="list-style-type: none"> • obserwacja i ocena aktywności na zajęciach • sprawdzian z progami punktowymi 	<ul style="list-style-type: none"> • Ćwiczenia
Jest w stanie przeprowadzić analizę poprawności algorytmu sekwencyjnego zapisanego w postaci blokowej i w języku programowania strukturalnego.	<ul style="list-style-type: none"> • K_U10 • K_K10 	<ul style="list-style-type: none"> • obserwacja i ocena aktywności na zajęciach • sprawdzian z progami punktowymi 	<ul style="list-style-type: none"> • Ćwiczenia
Student zna pojęcie złożoności obliczeniowej dotyczące algorytmów równoległych.	<ul style="list-style-type: none"> • K_W06 	<ul style="list-style-type: none"> • sprawdzian z progami punktowymi 	<ul style="list-style-type: none"> • Wykład
Potrafi definiować i charakteryzować klasy złożoności problemów algorytmicznych. Zna podstawowe pojęcia teorii poprawności algorytmicznej, automatów, gramatyk i języków.	<ul style="list-style-type: none"> • K_W06 	<ul style="list-style-type: none"> • sprawdzian z progami punktowymi 	<ul style="list-style-type: none"> • Wykład

Warunki zaliczenia

Wykład - warunkiem zaliczenia jest uzyskanie pozytywnej oceny z egzaminu przeprowadzonego w formie pisemnej.

Ćwiczenia - warunkiem zaliczenia jest uzyskanie pozytywnych ocen z kolokwium pisemnych lub ustnych przeprowadzonych co najmniej raz w semestrze.

Składowe oceny końcowej = wykład: 50% + ćwiczenia: 50%

Literatura podstawowa

1. Banachowski L., Kreczmar A.: Elementy analizy algorytmów, WNT, Warszawa, 1982
2. Biłski T., Chmiel K., Stokłosa J.: Zbiór zadań ze złożoności obliczeniowej algorytmów, Wydawnictwo Politechniki Poznańskiej, Poznań, 1992
3. Cormen T. H., Leiserson C. E., Rivest R. L.: Wprowadzenie do algorytmów, WNT, Warszawa, 1997
4. Harel D.: Rzecz o istocie informatyki, WNT, Warszawa, 2000
5. Hopcroft J. E., Ullmann J.D.: Wprowadzenie do teorii automatów, języków i obliczeń, PWN, Warszawa, 2003
6. Aho A.V., Ullmann J.D.: Wykłady z informatyki z przykładami w języku C, Helion, Gliwice, 2003

Literatura uzupełniająca

1. Ben Ari M.: Logika matematyczna w informatyce, WNT, Warszawa, 2005
2. Graham R.L., Knuth D.E., Patashnik O.: Matematyka konkretna, PWN, Warszawa, 2002
3. Papadimitriou C. H.: Złożoność obliczeniowa, Helion, Gliwice, 2012
4. Ross K. A., Wright C. R. B.: Matematyka dyskretna, PWN, Warszawa, 2000
5. Wróblewski P.: Algorytmy, struktury danych i języki programowania, Helion, Gliwice, 1997

Uwagi

Zmodyfikowane przez prof. dr hab. inż. Andrzej Obuchowicz (ostatnia modyfikacja: 20-04-2021 11:43)

Wygenerowano automatycznie z systemu SyllabUZ