Zaawansowane algorytmy rozproszone

Sylabus zajęć

Informacje podstawowe

Kierunek studiów Informatyka	**Cykl dydaktyczny** 2023/24
Specjalność -	**Kod zajęć** 06INFN.42S.01024.23
Jednostka organizacyjna Wydział Matematyki i Informatyki	**Języki wykładowe** polski
Poziom studiów studia drugiego stopnia poinżynierskie	**Obligatoryjność** Fakultatywny
Forma studiów studia niestacjonarne	**Blok zajęciowy** Przedmioty specjalnościowe
Profil studiów profil ogólnoakademicki	

Koordynator zajęć	Michał Hanćkowiak
Prowadzący zajęcia	Michał Hanćkowiak

Okres	**Forma zajęć / liczba godzin / forma zaliczenia**	**Liczba punktów ECTS**
Semestr 2	• Laboratorium: 15, Zaliczenie z oceną	3

Cele kształcenia dla zajęć

Kod	Cel
C1	Pokazanie, jakie są możliwości i ograniczenia obliczeń w systemie z wieloma procesorami połączonymi siecią lub wspólną pamięcią (tzw. model rozproszony).
C2	Pokazanie związku między "zabawkowymi" teoretycznymi problemami a światem realnych problemów w dziedzinie algorytmów rozproszonych.
C3	Zapoznanie się z różnymi technikami projektowania algorytmów rozproszonych, w szczególności ze sposobami projektowania optymalizacyjnych algorytmów grafowych.
C4	Pokazanie, jak się dowodzi poprawności, szacuje współczynnik aproksymacji i czas działania algorytmów rozproszonych.
C5	Zapoznanie z metodami/narzędziami do modelowania i automatycznego dowodzenia poprawności algorytmów rozproszonych.

Wymagania wstępne

Podstawowa wiedza z zakresu teorii grafów. Wiedza i umiejętności z klasycznych zajęć typu "Algorytmy i struktury danych". Podstawowe umiejętności programistyczne. Zdolność szybkiego nauczenia się nowego języka programowania.

Efekty uczenia się dla zajęć

Kod	Efekty uczenia się dla zajęć w zakresie	Efekty uczenia się dla kierunku	Metody weryfikacji osiągnięcia efektów uczenia się dla zajęć
Wiedzy – Student/ka:			
W1	rozumie, w jaki sposób modeluje się obliczenia w sieciach różnego typu.	INF_K4_W01, INF_K4_W02, INF_K4_W03, INF_K4_W04	Kolokwium pisemne
W2	zna i rozumie zaawansowane techniki projektowania algorytmów rozproszonych, złożone z wielu faz, często wykorzystujące pomocnicze struktury danych (spannery, klastry).	INF_K4_W01, INF_K4_W02, INF_K4_W03, INF_K4_W04	Kolokwium pisemne
W3	zna metody ułatwiające tworzenie algorytmów asynchronicznych.	INF_K4_W01, INF_K4_W02, INF_K4_W03, INF_K4_W04	Kolokwium pisemne
Umiejętności – Student/ka:			
U1	potrafi twórczo rozwinąć znane algorytmy i/lub znaleźć dla nich nowe zastosowania.	INF_K4_U01, INF_K4_U02, INF_K4_U05, INF_K4_U06, INF_K4_U12	Projekt, Raport
U2	potrafi się posługiwać narzędziem do automatycznej analizy algorytmów rozproszonych/współbieżnych.	INF_K4_U03, INF_K4_U05	Projekt, Raport
U3	potrafi się posługiwać symulatorem algorytmów rozproszonych w celu eksperymentów, analizy i testowania własnych algorytmów.	INF_K4_U05	Projekt, Raport

Treści programowe dla zajęć

Lp.	Treści programowe dla zajęć	Efekty uczenia się dla zajęć	Formy zajęć
1.	Modelowanie obliczeń rozproszonych w sieciach (model z przesyłaniem komunikatów); wersja synchroniczna i asynchroniczna; pojęcia: konfiguracja, zdarzenie, egzekucja; złożoność komunikatowa i czasowa algorytmów; implementowanie najprostszych algorytmów rozproszonych w symulatorze.	W1, U3	Laboratorium
2.	Problem wyboru lidera (LE) w cyklu, wersje o złożoności komunikatowej O(n^2) i O(n logn) w modelu asynchronicznym; dowód optymalności algorytmu o złożoności O(n logn); algorytmy LE w modelu synchronicznym o złożoności O(n); implementacja w symulatorze.	U1, U3	Laboratorium
3.	Obliczanie 3-kolorowania wierzchołków drzewa metodą Cole&Vishkin oraz różne uogólnienia (w szczególności (Delta+1)-kolorowanie grafu ogólnego) oraz zastosowania tego algorytmu; implementacja w symulatorze; dolne oszacowanie na 2 i 3-kolorowanie wierzchołkowe cyklu.	U1, U3	Laboratorium
4.	Obliczanie skojarzeń w grafie metodą skojarzenia frakcyjnego i deterministycznego zaokrąglania (algorytm M. Fischer z 2017); ten sam problem z użyciem tzw. “spannerów”; inne zastosowania spannerów; uogólnienia skojarzeń I ich zastosowania.	W2	Laboratorium
5.	Zastosowania struktury danych zwanej “klastrami” (z małym brzegiem) do aproksymowania problemów grafowych: MIS, Maximum Matching, MDS; procedura szybkiego budowania klastów; implementacja elementów tej procedury w symulatorze.	W3	Laboratorium
6.	Narzędzia do budowania algorytmów asynchronicznych: synchronizatory alfa, beta, gamma, ni1, eta; uogólnienie synchronizatora alfa; implementacja w symulatorze synchronizatorów alfa I beta.	U2	Laboratorium
7.	Algorytmy odporne na błędy sieci: problem Consensus; problem “2 generałów”; dolne oszacowania związane z problemem Consensus; implementacja w symulatorze algorytmu Consensus.	W1, U1	Laboratorium
8.	Narzędzie do automatycznej weryfikacji algorytmów rozproszonych: sieci Petriego, spin i język Promela, CSP czyli Communicating Sequential Processes; zastosowanie tych narzędzi do analizy algorytmów asynchronicznych.	U2	Laboratorium

Informacje dodatkowe

Forma zajęć	Metody i formy prowadzenia zajęć
Laboratorium	Dyskusja, Rozwiązywanie zadań (np.: obliczeniowych, artystycznych, praktycznych), Metoda ćwiczeniowa, Metoda laboratoryjna, Metoda badawcza (dociekania naukowego), Metoda projektu

Forma zajęć	Warunki zaliczenia zajęć
Laboratorium	Końcowa ocena składa się z następujących elementów: 1. kolokwium pisemne – 1/3 łącznej liczby punktów, 2. projekt – 1/3 łącznej liczby punktów, 3. raport – 1/3 łącznej liczby punktów. Skala ocen: 1. bardzo dobry (bdb; 5,0) – od 90% punktów, 2. dobry plus (db plus; 4,5) – od 80% punktów, 3. dobry (db; 4,0) – od 70% punktów, 4. dostateczny plus (dst plus; 3,5) – od 60% punktów, 5. dostateczny (dst; 3,0) – od 50% punktów, 6. niedostateczny (ndst; 2,0) – poniżej 50% punktów.

Literatura

Obowiązkowa

1. Nancy Lynch "Distributed Algorithms"
2. Hagit Attiya, Jennifer Welch, "Distributed Computing - Fundamentals, Simulations, and Advanced Topics"
3. C. A. R. Hoare "Communicating Sequential Processes"

Nakład pracy studenta i punkty ECTS

Rodzaje zajęć studenta	Średnia liczba godzin* przeznaczonych na zrealizowane rodzaje zajęć
Laboratorium	15
Przygotowanie do zajęć	20
Czytanie wskazanej literatury	10
Przygotowanie raportu	10
Przygotowanie projektu	10
Przygotowanie do zaliczenia	10
Łączny nakład pracy studenta	**Liczba godzin** 75
Liczba punktów ECTS	**ECTS** 3

* godzina (lekcyjna) oznacza 45 minut

Efekty uczenia się dla kierunku

Kod	Treść
INF_K4_U01	Absolwent/ka potrafi zastosować zaawansowaną wiedzę matematyczną do formułowania, analizowania i rozwiązywania złożonych i nietypowych zadań związanych z informatyką
INF_K4_U02	Absolwent/ka potrafi adaptować istniejące oraz tworzyć nowe metody informatyczne do rozwiązywania nieszablonowych problemów praktycznych i teoretycznych
INF_K4_U03	Absolwent/ka potrafi stosować zaawansowane metody budowy oprogramowania, rozstrzyga o ich przydatności, w tym podejmuje decyzje dotyczące wyboru technik prowadzących do otrzymania oprogramowania wysokiej jakości
INF_K4_U05	Absolwent/ka potrafi formułować i testować nowe algorytmy i metody rozwiązywania problemów w wybranych obszarach informatyki na potrzeby prowadzenia prac badawczo-rozwojowych z uwzględnieniem aktualnego stanu wiedzy
INF_K4_U06	Absolwent/ka potrafi rozwiązywać złożone problemy z wybranych obszarów informatyki oraz proponować nowe algorytmy, narzędzia i metody wykorzystując odpowiednio dobrane źródła, które poddaje krytycznej analizie, syntezie i twórczej interpretacji
INF_K4_U12	Absolwent/ka potrafi samodzielnie pogłębiać i aktualizować wiedzę i umiejętności z zakresu informatyki oraz określać kierunki dalszego rozwoju zawodowego
INF_K4_W01	Absolwent/ka zna i rozumie w pogłębionym stopniu pojęcia z działów matematyki niezbędne do rozwiązywania zaawansowanych problemów w informatyce
INF_K4_W02	Absolwent/ka zna i rozumie współczesny stan badań i tendencje rozwojowe w wiodących obszarach informatyki
INF_K4_W03	Absolwent/ka zna i rozumie w pogłębionym stopniu współczesne metody, narzędzia i technologie informatyczne właściwe dla wybranych obszarów zastosowań niezbędne przy budowie złożonych systemów informatycznych oraz przy prowadzeniu prac badawczo-rozwojowych
INF_K4_W04	Absolwent/ka zna i rozumie zasady rozwiązywania problemów z wykorzystaniem zaawansowanych algorytmów i metod informatycznych