

Zaawansowane techniki programowania - opis przedmiotu

Informacje ogólne	
Nazwa przedmiotu	Zaawansowane techniki programowania
Kod przedmiotu	11.3-WE-INFD-ZTP
Wydział	Wydział Informatyki, Elektrotechniki i Automatyki
Kierunek	Informatyka
Profil	ogólnoakademicki
Rodzaj studiów	drugiego stopnia z tyt. magistra inżyniera
Semestr rozpoczęcia	semestr zimowy 2020/2021

Informacje o przedmiocie	
Semestr	2
Liczba punktów ECTS do zdobycia	5
Typ przedmiotu	obowiązkowy
Język nauczania	polski
Sylabus opracował	dr hab. inż. Marek Sawerwain, prof. UZ

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
Wykład	15	1	9	0,6	Zaliczenie na ocenę
Laboratorium	30	2	18	1,2	Zaliczenie na ocenę
Projekt	15	1	9	0,6	Zaliczenie na ocenę

Cel przedmiotu

- zapoznanie studentów z wybranymi technikami zaawansowanego programowania wieloparadygmatowego, w modelu klasycznym oraz zaprezentowanie podstaw modelu kwantowego,
- omówienie wybranych technik jak np. prezentacja typów uogólnionych, programowanie reaktywne, model aktorów, mechanizm hot-swapping,
- zaprezentowanie kwantowego modelu obliczeniowego, definicji qubitów, superpozycji i splątania kwantowomechanicznego jako nowych zasobów przetwarzania informacji,
- prezentacja wybranych algorytmów kwantowych i protokołów, w tym kwantowych protokołów kryptograficznych.

Wymagania wstępne

Metody programowania, Algorytmy i struktury danych, Teoretyczne Podstawy Informatyki, Programowanie obiektowe

Zakres tematyczny

Prezentacja i przypomnienie obecnie stosowanych paradygmatów programowania.

Podkreślenie roli paradygmatu funkcyjnego i logicznego (deklaratywnego).

Typy uogólnione, programowanie reaktywne, programowanie w logice za pomocą języka programowania Erlang. Prezentacja modelu aktorów.

Nowe modele obliczeniowe jako odpowiedź na złożoność wykładniczą.

Prezentacja pojęć kwantowego modelu obliczeniowego, qubitów, rejestru kwantowego, superpozycji oraz splątania. Operacje unitarne oraz pomiaru realizowane na rejestrze kwantowym.

Omówienie wybranych algorytmów kwantowych oraz ich implementacja w postaci obwodów kwantowych oraz programów w języku Q# (w środowisku Visual Studio).

Symulacje obliczeń kwantowych oraz realizacja wybranych obwodów (protokołów) na dostępnych obecnie eksperymentalnych instalacjach obliczeń kwantowych.

Metody kształcenia

Wykład: wykład konwencjonalny/tradycyjny.

Laboratorium: ćwiczenia laboratoryjne, wg listy zadań.

Projekt: praca w grupach, metoda projektu.

Efekty uczenia się i metody weryfikacji osiągnięcia efektów uczenia się

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
potrafi wskazać zalety łączenie wielu paradygmatów w celu efektywniejszego rozwiązywania problemów	• K_W09	• sprawdzian z progami punktowymi	• Wykład
zna oraz potrafi zalety korzystania z typów uogólnionych oraz technik programowania reaktywnego	• K_W11	• bieżąca kontrola na zajęciach • projekt	• Laboratorium • Projekt

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
umie wykorzystać model aktorów do utworzenia współbieżnego serwera	K_U16	- projekt - sprawdzian z progami punktowymi	- Laboratorium - Projekt
jest świadom wyzwań obliczeniowych wymagających wykładniczego nakładu pracy	K_K01	sprawdzian z progami punktowymi	Wykład
zna podstawowe kwantowe protokoły komunikacji (teleportacja) oraz kryptografii kwantowej	K_U14 K_K06	- projekt - sprawdzian z progami punktowymi	- Laboratorium - Projekt
zna model obliczeń kwantowych	K_W01	sprawdzian z progami punktowymi	Wykład

Warunki zaliczenia

Wykład - warunkiem zaliczenia jest uzyskanie pozytywnej oceny ze sprawdzianu przeprowadzonego w formie pisemnej.

Laboratorium - warunkiem zaliczenia jest uzyskanie pozytywnych ocen ze wszystkich sprawdzianów pisemnych z ćwiczeń laboratoryjnych, przewidzianych do realizacji w ramach programu laboratorium.

Projekt - warunkiem zaliczenia jest wykonanie wszystkich zadań projektowych, przewidzianych do realizacji w ramach zajęć projektowych oraz przygotowanie pisemnego raportu ze zrealizowanego projektu.

Składowe oceny końcowej = wykład: 20% + laboratorium: 40% + projekt: 40%

Literatura podstawowa

1. Vladimir Silva, Practical Quantum Computing for Developers, Apress, 2018.
2. John Watrous, The Theory of Quantum Information, Cambridge University Press, 2018.
3. Simon St. Laurent: Introducing Erlang. Getting Started in Functional Programming. 2nd Edition, O'Reilly Media, 2017.
4. Marek Sawerwain, Joanna Wiśniewska: Informatyka kwantowa. Wybrane obwody i algorytmy, PWN, Warszawa 2015.
5. Max Bramer, Logic Programming with Prolog, Springer-Verlag London, 2013.
6. Michael A. Nielsen, Isaac L. Chuang: Quantum Computation and Quantum Information: 10th Anniversary Edition, Cambridge University Press, 2010.
7. Francesco Cesarini, Simon Thompson: Erlang Programming, O'Reilly Media, 2009.
8. Mika Hirvensalo: Algorytmy kwantowe, Wydawnictwa Szkolne i Pedagogiczne, Warszawa 2004.
9. Krzysztof Giaro, Marcin Kamiński: Wprowadzenie do algorytmów kwantowych, Akademicka Oficyna Wydawnicza EXIT, Warszawa 2003.

Literatura uzupełniająca

1. Ingemar Bengtsson, Karol Życzkowski: Geometry of Quantum States: An Introduction to Quantum Entanglement, 2nd Edition, Cambridge University Press, 2017.
2. Jacek Jurkowski: Korelacje Nieklasyczne. Kwantowe splątanie i dyskord, Wydawnictwo Naukowe Uniwersytetu Mikołaja Kopernika, Toruń 2014.
3. Akira Furusawa, Peter van Loock: Quantum Teleportation and Entanglement: A Hybrid Approach to Optical Quantum Information Processing, Wiley, 2011.

Uwagi

Zmodyfikowane przez prof. dr hab. inż. Andrzej Obuchowicz (ostatnia modyfikacja: 27-04-2020 08:37)

Wygenerowano automatycznie z systemu SylabUZ