

Graph and Networks in Computer Science - course description

General information

Course name	Graph and Networks in Computer Science
Course ID	11.9-WI-INF-D-GiSwI
Faculty	Faculty of Computer Science, Electrical Engineering and Automatics
Field of study	Computer Science / Computer Systems Engineering
Education profile	academic
Level of studies	Second-cycle studies leading to MSc degree
Beginning semester	summer term 2016/2017

Course information

Semester	1
ECTS credits to win	5
Course type	obligatory
Teaching language	polish
Author of syllabus	dr hab. inż. Andrei Karatkevich, prof. UZ

Classes forms

The class form	Hours per semester (full-time)	Hours per week (full-time)	Hours per semester (part-time)	Hours per week (part-time)	Form of assignment
Laboratory	30	2	18	1,2	Credit with grade
Lecture	30	2	18	1,2	Exam

Aim of the course

- zapoznanie studentów z podstawami teorii grafów i najważniejszymi (w zastosowaniach informatycznych) algorytmami grafowymi
- ukształtowanie podstawowych umiejętności w zakresie zastosowania algorytmów grafowych do problemów informatycznych
- zapoznanie studentów z sieciami Petriego jako modelem procesów współbieżnych oraz ich zastosowaniem w projektowaniu systemów sterowania logicznego

Prerequisites

Podstawy programowania, Teoretyczne podstawy informatyki

Scope

Nieformalne wprowadzenie do teorii grafów. Rys historyczny. Zastosowania teorii grafów w informatyce.

Grafy skierowane i nieskierowane. Intuicyjne przykłady i formalne definicje. Podstawowe pojęcia teorii grafów. Rodzaje grafów: grafy planarne, dwudzielne, pełne, drzewa.

Sposoby reprezentacji grafów w pamięci komputera: macierzowe i listowe reprezentacje grafów.

Najważniejsze algorytmy przeszukiwania grafów: BFS, DFS. Wybrane zastosowania algorytmów przeszukiwania (obliczenie silnie spójnych składowych, sortowanie topologiczne)

Cykle Eulera (warunki istnienia i algorytm konstruowania). Cykle Hamiltona, warunki istnienia.

Metoda PERT, analiza krytycznej ścieżki.

Metody konstruowania minimalnych drzew rozpinających (algorytmy Prima, Kruskala, Boruvki).

Metody obliczania najkrótszych ścieżek w grafach (algorytmy Dijkstry, Bellmana-Forda, Floyda-Warshalla). Zastosowania.

Sieci przepływowe i obliczenie maksymalnego przepływu w sieciach – metoda Forda-Fulkersona. Zastosowania.

Kolorowanie grafów i jego zastosowania. Heurystyczne metody kolorowania grafów.

Złożoność obliczeniowa algorytmów grafowych. Algorytmy zachłanne. Problemy NP-trudne w kontekście grafów, przykłady.

Binarne diagramy decyzyjne: klasyczny graf BDD, zredukowany binarny diagram ROBDD. Grafy BDD jako efektywny sposób reprezentacji funkcji boolowskich.

Elementy teorii sieci Petriego: podstawy formalne, właściwości behawioralne, metody badania. Sieci Petriego w modelowaniu systemów współbieżnych i projektowaniu współbieżnych algorytmów sterowania.

Algorytmy do zaimplementowania na zajęciach laboratoryjnych:

1. Przeszukiwanie wszcz
2. Przeszukiwanie w głąb
3. Obliczanie silnie spójnych składowych
4. Sortowanie topologiczne
5. Wyznaczenie krytycznej ścieżki
6. Znajdowanie minimalnego drzewa rozpinającego

7. Najkrótsze ścieżki z jednym źródłem (algorytm Dijkstry)
8. Kolorowanie grafów za pomocą wybranego algorytmu zachłannego
9. Obliczanie maksymalnego przepływu w sieciach

Teaching methods

wykład: wykład konwencjonalny, dyskusja, przedstawienie wybranych algorytmów przez studentów

laboratorium: ćwiczenia laboratoryjne z wykorzystaniem sprzętu komputerowego

Learning outcomes and methods of theirs verification

Outcome description	Outcome symbols	Methods of verification	The class form
Ma podbudowaną teoretycznie szczegółową wiedzę na temat algorytmów grafowych i sieciowych oraz sposobów reprezentacji grafów w pamięci komputera; potrafi zaimplementować najważniejsze algorytmy grafowe w jednym z uniwersalnych języków programowania	• K_W04	- an observation and evaluation of the student's practical skills - carrying out laboratory reports	Laboratory
Potrafi prowadzić w razie potrzeby problemy informatyczne do zagadnień grafowych i stosować algorytmy grafowe do ich rozwiązywania	• K_U08	- a discussion - a quiz - an exam - oral, descriptive, test and other	- Lecture - Laboratory
Umie opisać relacje w systemie lub strukturze przy pomocy modeli grafowych, a dynamiczny proces współbieżny, np. sterowania logicznego - przy pomocy sieci Petriego	• K_U08	- a quiz - carrying out laboratory reports	Laboratory
Ma rozszerzoną i pogłębianą wiedzę w zakresie teorii grafów, przydatną do formułowania i rozwiązywania zadań w zakresie modelowania matematycznego i optymalizacji	• K_W01	- a discussion - an evaluation test - an exam - oral, descriptive, test and other	Lecture

Assignment conditions

Wykład - warunkiem zaliczenia jest uzyskanie pozytywnej oceny z egzaminu pisemnego lub ustnego.

Laboratorium - warunkiem zaliczenia jest uzyskanie pozytywnych ocen ze wszystkich zadań laboratoryjnych.

Składowe oceny końcowej = wykład: 60% + laboratorium: 40%

Recommended reading

1. Robin Wilson: Wprowadzenie do teorii grafów. Wydawnictwo Naukowe PWN, Warszawa, 2007.
2. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein: Wprowadzenie do algorytmów, PWN, Warszawa, 2012 (albo inne wydania).
3. Marek Libura, Jarosław Sikorski: Wykłady z matematyki dyskretnej. Cz. II: Teoria grafów. WSISZ, Warszawa, 2002.
4. Marek Kubala (Red.), Optymalizacja dyskretna: modele i metody kolorowania grafów. WNT, Warszawa, 2002

Further reading

1. Narsingh Deo: Teoria grafów i jej zastosowanie w technice i informatyce, PWN, Warszawa, 1980.
2. Bogdan Korzan: Elementy teorii grafów i sieci. WNT, Warszawa, 1978.
3. Reinhard Diestel: Graph theory. Electronic edition, Springer Verlag New York, 2000.
4. Marcin Szpyrka: Sieci Petriego w modelowaniu i analizie systemów współbieżnych, WNT Warszawa, 2008.

Notes

Modified by prof. dr hab. inż. Krzysztof Patan (last modification: 21-09-2016 10:04)

Generated automatically from SyllabUZ computer system