

Graphs and networks in computer science - opis przedmiotu

Informacje ogólne	
Nazwa przedmiotu	Graphs and networks in computer science
Kod przedmiotu	11.9-WE-INF-D-GaNiCS-Er
Wydział	Wydział Informatyki, Elektrotechniki i Automatyki
Kierunek	Informatyka
Profil	ogólnoakademicki
Rodzaj studiów	Program Erasmus drugiego stopnia
Semestr rozpoczęcia	semestr zimowy 2020/2021

Informacje o przedmiocie	
Semestr	1
Liczba punktów ECTS do zdobycia	5
Typ przedmiotu	obowiązkowy
Język nauczania	angielski
Sylabus opracował	dr inż. Grzegorz Łabiak

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	30	2	-	-	Zaliczenie na ocenę
Laboratorium	30	2	-	-	Zaliczenie na ocenę

Cel przedmiotu

- Skills and competencies in the area of the basic notions of the graph theory and the most important (especially for applications in the computer science) graph algorithms
- Skills of applying the graph algorithms to solving the real life problems and software implementation of the graph algorithms
- Competencies in the area of Petri nets as a model of concurrent processes

Wymagania wstępne

- Principles of programming
- Theoretical foundations of computer science

Zakres tematyczny

Introduction to the graph theory. History of the area. Applications of the graph theory in computer science and software.

Directed and undirected graphs. Examples, definitions of the main notions of the graph theory. Important classes of graphs.

Ways of representing of the graphs in data structures - matrices and lists.

The main graph search algorithms: BFS, DFS. Selected applications of the graph search algorithms (topological sorting, finding the strongly connected components).

Eulerian paths, conditions of existence, methods of finding. Hamiltonian paths, conditions of existence. Applications.

Program evaluation and review technique, critical path method.

Minimal spanning trees, application, algorithms of constructing.

Shortest paths in the graphs (Dijkstra's, Bellman-Ford, Floyd-Warshall algorithms).

Maksimum flow in the flow networks, Ford–Fulkerson algorithm.

Graph coloring and its applications. Greedy algorithms of graph coloring.

Computational complexity of the graph algorithms. NP-hard graph problems, travelling salesman problem as an example. Heuristic and approximation algorithms for the NP-hard problems.

Binary decision diagrams: BDD, ROBDD, other modifications. BDD as an efficient way of representing the Boolean functions.

Elements of the Petri net theory: main notions, behavioral properties, methods of analysis, main applications.

Metody kształcenia

Lecture: conventional lecture, discussions, presenting of the selected algorithms by the students

Laboratory: computer laboratory exercises

Efekty uczenia się i metody weryfikacji osiągnięcia efektów uczenia się

Opis efektu	Symbole efektów	Metody weryfikacji	Forma zajęć
Student can represent the appropriate tasks in terms of the graph theory and apply the graph algorithms to solve them		- dyskusja - egzamin - ustny, opisowy, testowy i inne - sprawdzian	- Wykład - Laboratorium
Student is able to represent a system (if appropriate) in a graph form, and the parallel processes as a Petri net		- sprawdzian - wykonanie sprawozdań laboratoryjnych	Laboratorium
Student has skills in the area of graph theory which can be applied to formulation and solving the tasks of system modelling and optimization		- dyskusja - egzamin - ustny, opisowy, testowy i inne - kolokwium	Wykład
Student has skills in the area of graph algorithms and is able to implement such algorithms in one of the universal programming languages		- obserwacje i ocena umiejętności praktycznych studenta - wykonanie sprawozdań laboratoryjnych	Laboratorium

Warunki zaliczenia

Lecture – the passing condition is to obtain a positive mark from the final test.

Laboratory – the passing condition is to obtain positive marks from all laboratory exercises to be planned during the semester.

Calculation of the final grade: lecture 60% + laboratory 40%

Literatura podstawowa

1. Robin Wilson: Introduction to graph theory. Pearson Education Limited, 1996 (or other editions).
2. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein: Introduction to Algorithms. MIT Press and McGraw-Hill, 1990 (or other editions).
3. Marek Kubale (Ed.), Graph Colorings. American Mathematical Soc., 2004

Literatura uzupełniająca

1. Narsing Deo: Graph Theory with Application to Engineering and Computer Science, Prentice-Hall, Englewood Cliffs, N.J., 1974
2. Reinhard Diestel: Graph theory. Electronic edition, Springer Verlag New York, 2000.
3. Wolfgang Reisig: A Primer in Petri Net Design. Springer-Verlag, 1992.

Uwagi

Zmodyfikowane przez dr inż. Grzegorz Łabiak (ostatnia modyfikacja: 26-04-2020 20:44)

Wygenerowano automatycznie z systemu SyllabUZ