CURRICULUM

I confirm:
Dean:

CS 201

THEORY OF ALGORITHMS

Updated: Protocol No 7/30.01.2019
Lecturer: Associate Prof. Yanislav Zhelev,PhD

ANNOTATION

The discipline provides knowledge of the basic concepts, apparatus and methodology of the creation, evaluation and implementation of algorithms. Consider the most popular algorithms in the field of sorting, split and master algorithms, greedy algorithms, algorithms on graphs and trees, flow theory, parallel algorithms. The methodology for assessing the complexity of algorithms and the dependence on the hardware configuration is considered.

MAIN OBJECTIVES

The main objective of the course is to master the basic concepts, apparatus and some of the most popular methods of algorithm theory.

The course introduces terminology and basic methods for assessing the complexity of algorithms and the mathematical apparatus for finding asymptotic estimates/functions O(n), (n), Ω(n), o(n) and (n)/. Insertion Sort, Selection Sort, Merge Sort, Quick Sort, Heap Sort,Radix Sort/, Split & Rule algorithms, greedy algorithms, graph/covering algorithms, crawls, shortest paths in graphs, algorithms to realise maximum graph flow, parallel algorithms and parallel computer architectures are analysed. The algorithms under consideration are realised on a pseudocode.

The knowledge gained is also used by other programming courses.

PREREQUISITES

Students should have completed the courses: Programming: C++ and Object Oriented Programming.

Support courses: Further Mathematics (Mathematics HL) and Discrete Mathematics.

STATUS AND STRUCTURE

specialty	status	ECTS	full-time training				part-time training			
			L	C	t	total	L	C	t	total
Software Engineering	Mandatory	7	30	40		70	15	20		35

COURSE CONTENT

Topic 1. Algorithms. Concept of algorithm. Complexity of algorithms. Benchmarking. Asymptotics. Upper and lower asymptotic limits. Asymptotic scores – functions O(n), (n), Ω(n), o(n) and (n). Dependencies between asymptotic functions.

Topic 2. Sortings. Slow sorting methods. Insertion Sort, Bubble sort, Selection sort.

Application of sortings. Search in a sorted array. Analysis of execution time.

Topic 3. Sortings. Quick sorting methods. Counting sort, Radix sort. Analysis of execution time.

Topic 4. Sortings. Other sorting methods. Merge Sort, Quick Sort,Heap Sort, Execution Time Analysis.

Topic 5. Elementary data structures. It's a steak. Operations on stacks.

Topic 6. Elementary data structures. It's a tail. Queue operations.

Topic 7. Successive structures. Presentation of successive structures. A single-linked linear list. Operations on single-linked lists.

Topic 8. Successive structures. A double-linked linear list. Operations on double-linked lists.

Topic 9. Hashing. Tables with direct access. Hash tables and hash functions. Hash table with a list for continuation. Hash table with open addressing. Methods of constructing hash functions.

Topic 10. Trees, binary trees, binary search trees. Definition, presentation crawl.

Topic 11. Binary Trees of Search. Algorithms on binary search trees.

Topic 12. It's Counts. Basic concepts. Presentation of Counts. Crawling in width and depth.

Topic 13. Minimum covering trees. Kruscal's algorithm. Prim's algorithm. Applications of the minimum covering trees.

Topic 14. It's Counts. Shortest roads in Count. The road and the shortest time. Find the shortest path from fixed peak to all other peaks in graph. Dijkstra algorithm. Belmann- Ford algorithm. Find all the shortest paths in the count. Floyd-Warshall algorithm.

Topic 15. It's Counts. Streams in graphs. Ford's Fulkerson algorithm for maximum flow.

Topic 16. Innovative structures. B-Trees. Basic concepts. Algorithms on the B-trees.

SEMINAR EXERCISES

Topic 1. Asymptotic behavior of functions. Rules for calculating the execution time of operators. Rules for calculating the execution time of an algorithm. Methods for calculating O(n).

Topic 2. Sorting methods. Analysis of execution time. Annexes. Examples.

Topic 3. Elementary data structures. The concept of a structure with direct access. Relay stack, tail and DEQ on array.

Topic 4. Successive structures. The concept of a consistent structure. Relay of a single and two-linked linear list. Cyclical lists.

Topic 5. Hashing. Examples of hash tables and construction of hash functions.

Topic 6. Trees, binary trees and binary search trees. Examples, implementation, operations. Annexes.

Topic 7. It's Counts. Examples. Crawling. Topological sorting.

Topic 8. Minimum covering trees. Annexes. Examples.

Topic 9. Shortest roads in counts. Examples and Applications.

Topic 10. Streaming algorithms. Examples and Applications.

Topic 11. In-trees. Implementation and Applications.

ASSESSMENT METHODS

The work of each student during the semester is assessed with an ongoing assessment. It is the result of the evaluation of two control works. Each student is also assessed according to their participation in solving tasks during exercises.

The semester exam is oral. The final score is formed as a result of the semester exam, the current evaluation and the tasks solved on the exam.

To form the assessment, the student earns points, the maximum value of which is 100. The allocation of points for the activities is up to:

1. Ongoing control..... 50
    a. 2 control works..... 40
    b. active participation in the exercises.....10

2. Examination procedure..... 50
    a. Theoretical questions.....30
    b. Practical tasks.....20

WithCala for assessment:

- Up to 50 points including - Weak (2);
- 51 to 60 points including - Average (3);
- 61 to 70 points including - Good (4);
- 71 to 80 points including - Very good (5);
- 81 to 100 points including - Excellent (6).

RECOMMENDED LITERATURE

1. Thomas Cormen, Charles Leiserson, Ronald Rivest, Introduction to Algorithms, Third Edition, The MIT Press, Cambridge, Massachusetts, 2009.
2. Thomas Cormen, Algorithms Unlocked, MIT 2013.
3. Robert Sedgewick, Kevin Wayne, Algorithms, Fourth Edition, Pearson Education 2011.
4. Preslav Nakov, Panayot Dobrikov, Programming = ++Algorithms, Top Team Co, Sofia, 2003.
5. Emil Kelevedzhiev, Dynamic Optimisation, Anubis, Sofia, 2000.
6. Krasimir Manev, Introduction to Discrete Mathematics, NBU Publishing, Sofia, 1996.