ITMO University

Analysis and development of algorithms

Course Workload		Assessment form (examination/ graded test/ ungraded test)
ECTS	Hours	
3	108	Written test

This course introduces students to the basic principles of computer science: algorithmic thinking and computational problem-solving. Studying this course will enable them to gain skills in applying computational methods to solve specific problems in various areas of mathematics and its applications. This course will provide an overview of the basic equations of mathematical physics and classical problems for them, the concept of the generalized solution of a problem for a partial differential equation, numerical algorithms for solving problems in algebra, analysis, problems related to differential equations, methods of implementing computations on modern computers. After successful completion of this section, students will be able to solve problems using the studied methods and analyze the resulting solution, apply numerical methods in solving applied problems, estimate the error of the obtained solution and prove the properties of partial differential equations.

Course structure:

1. Introduction to Analysis and Algorithm Design

1.1. Algorithm definition. Algorithm design and analysis process.
1.2. Algorithm design methods.
1.3. Important types of tasks.
1.4. Basic data structures and their applications.
1.5. Computational complexity of algorithms. Analytical and empirical estimates of complexity.

2. Algorithms for unconstrained nonlinear optimization. Direct methods

2.1. Optimization problem. Numerical optimization algorithms.
2.2. Direct methods of one-dimensional optimization: brute force, dichotomy, golden ratio.
2.3. Direct methods of multivariate optimization: brute force, Gauss, Nelder-Mead.

3. Algorithms for unconstrained nonlinear optimization. First and second order methods

3.1. First-order optimization techniques: gradient descent and conjugate gradient method.
3.2. Second-order optimization methods: Newton's method and Levenberg-Marquardt algorithm.

4. Algorithms for unconstrained nonlinear optimization. Stochastic and Metaheuristic Algorithms

4.1. Stochastic and metaheuristic optimization methods.
4.2. Simulated annealing method.
4.3. Differential evolution method.
4.4. Particle swarm method.

5. Graph algorithms. Introduction to graphs and basic algorithms on graphs

5.1. Graphs. Graph representation methods.
5.2. Algorithms on unweighted graphs: depth-first search and breadth-first search.

6. Graph algorithms. Pathfinding algorithms on weighted graphs

6.1. Pathfinding algorithms on weighted graphs: Algorithms of Dijkstra, A * and Bellman-Ford.

7. Graph algorithms. Network Analysis Tools

7.1. Graphs and their characteristics.
7.2. Modularity and the problem of identifying communities in a graph.
7.3. Network analysis software tools.

8. Practical analysis of modern algorithms

8.1. Analytical and experimental analysis of modern algorithms for solving specific practical problems of data analysis (optimization algorithms, algorithms on graphs, etc.)