

Project topics

Algorithms and data structures

1. Basic algorithmic techniques (greedy algorithms, binary search, sorting and dynamic programming).
2. Basic data structures (stack, queue, hash table, priority queue, binary search tree, graph and string).
3. String algorithms (suffix trees, suffix arrays, Burrows-Wheeler transform, Knuth-Morris-Pratt algorithm).
4. Complexity classes of algorithms.

Optimization algorithms

5. Linear programming algorithms (Simplex algorithm, Ellipsoid algorithm, Projective algorithm).
6. Constraint nonlinear optimization algorithms (Trust Region Reflective algorithm, Preconditioned Conjugate Gradient method, Active Set algorithm)*.
7. Quasi-Newton methods for unconstrained non-linear optimization (Symmetric Rank 1, BHHH algorithm, BFGS algorithm, Limited-memory BFGS)*.
8. Discrete optimization algorithms (Sequential Search, Depth-First Search, Depth-First Branch-and-Bound, Iterative Deepening Search, Best-First Search).
9. Karush-Kuhn-Tucker conditions in nonlinear optimization.

Algorithms on graphs

10. Community detection algorithms for attributed graphs (Weight-based and distance-based models)*.
11. Community detection algorithms for attributed graphs (Modified Louvain and Normalised Cut)*.
12. Community detection algorithms for attributed graphs (Non-negative matrix factorization)*.
13. Community detection algorithms for attributed graphs (Probabilistic models)*.
14. Consensus clustering algorithms*.
15. Epidemic processes on graphs (SIRS model, SEIR model)*.
16. Models for random graphs (Erdős-Rényi model, Barabási-Albert model)*.

Requirements

Choose a topic and form a group of 3 students per topic (not every topic is required to be covered, some groups may be different in size if there is a problem with the partition). Every group member gets the same score after the defence. The topics and methods indicated may be slightly varied during the project preparation.

The topics with * are recommended for students with strong background in mathematics and programming.

The project should be at most 20 pages A4 (plus the front page and source code if necessary). In the first part, theoretical information on the topic should be provided (e.g. description of algorithms, methods, models, etc.). It is important to explain main ideas but not full technical details. Include corresponding formulas, schemes or/and plots. In the second part, practical applications should be given e.g. particular problems and their solutions by the methods from the first part. You are free to choose the applications and data. Program or use existing implementations to demonstrate the work of the methods/ideas under consideration. Include corresponding numerical results or/and plots. The third part should include the bibliography you used. Appendixes with the source code and additional data may be added, if necessary.

Presentation and project defence

The presentation (the defence) of the project is at most 15 minutes and should include the main concepts and results obtained within the project.

The last lecture and lab are reserved to the defence of the projects. The pdf files of the project reports should be sent to chunaev@itmo.ru two weeks before the defence (i.e. the last lecture) for evaluation. The corrections proposed to be done after the evaluation (if any) should be done before the defence. Bring the printed corrected version to the defence.

Темы для проектов

Алгоритмы и структуры данных

1. Основные алгоритмические техники (жадные алгоритмы, бинарный поиск, сортировка и динамическое программирование).
2. Базовые структуры данных (стек, очередь, хеш-таблица, очередь приоритетов, двоичное дерево, граф и строка).
3. Строковые алгоритмы (суффиксные деревья, суффиксные массивы, преобразование Барроуза-Уилера, алгоритм Кнута-Морриса-Пратта).
4. Классы сложности алгоритмов.

Алгоритмы оптимизации

5. Алгоритмы линейного программирования (Симплекс-метод, Эллипсоидный алгоритм, Проективный алгоритм).
6. Алгоритмы нелинейной оптимизации с ограничениями (Trust Region Reflective algorithm, Preconditioned Conjugate Gradient method, Active Set algorithm) *.
7. Квазиньютоновские методы нелинейной оптимизации без ограничений (Symmetric Rank 1, BHHN algorithm, BFGS algorithm, Limited-memory BFGS) *.
8. Дискретные алгоритмы оптимизации (Sequential Search, Depth-First Search, Depth-First Branch-and-Bound, Iterative Deepening Search, Best-First Search).
9. Условия Каруша-Куна-Такера в нелинейной оптимизации.

Алгоритмы на графах

10. Алгоритмы обнаружения сообщества для графов с атрибутами (модели на основе веса и расстояния) *.
11. Алгоритмы обнаружения сообщества для графов с атрибутами (Modified Louvain и Normalized Cut) *.
12. Алгоритмы обнаружения сообщества для графов с атрибутами (неотрицательная матричная факторизация) *.
13. Алгоритмы обнаружения сообщества для графов с атрибутами (вероятностные модели) *.
14. Ансамблевые алгоритмы кластеризации*.
15. Эпидемические процессы на графах (модель SIRS, модель SEIR) *.
16. Модели для случайных графов (модель Эрдеша-Реньи, модель Барабаши-Альберта) *.

Требования

Выберите тему и сформируйте группы по 3 человека на каждую тему (не каждая тема должна быть освещена, количество человек в группе может быть изменено, если возникает проблема с разделением). Каждый участник группы получает одинаковые баллы после защиты. Указанные темы и методы могут незначительно меняться во время подготовки проекта.

Темы с * рекомендуются для студентов с сильным опытом в математике и программировании.

Проект должен быть не более 20 страниц формата A4 (плюс титульный лист и исходный код при необходимости). В первой части должна быть представлена теоретическая информация по теме (например, описание алгоритмов, методов, моделей и т.д.). Важно объяснить основные идеи, а не привести полные технические детали. Включите соответствующие формулы, схемы и / или графики. Во второй части должны быть даны практические приложения, например, конкретные проблемы и пути их решения методами из первой части. Вы можете выбирать приложения и

данные. Запрограммируйте или используйте существующие реализации, чтобы продемонстрировать работу рассматриваемых методов/идей. Включите соответствующие числовые результаты и/или графики. Третья часть должна включать использованную вами библиографию. При необходимости могут быть добавлены приложения с исходным кодом и дополнительными данными.

Презентация и защита проекта

Презентация (защита) проекта занимает не более 15 минут и должна включать основные концепции и результаты, полученные в рамках проекта.

Последняя лекция и лабораторная работа предназначены для защиты проектов. PDF-файлы отчетов проекта должны быть отправлены на chunaev@itmo.ru за две недели до защиты (то есть до последней лекции) для проверки. Исправления, которые предлагается сделать после проверки (если таковые будут), должны быть сделаны до защиты.