

Практикум по алгоритмам

ПРОГРАММА КУРСА

OZON MASTERS 2021-22 г.

Оглавление

Общие положения	2
План курса	6
1 Базовые навыки	7
1.1 Введение. Работа с системой Codeforces, знакомство с C++	7
1.1.1 Некоторые планируемые задачи	7
1.2 Структуры данных	8
1.2.1 Некоторые планируемые задачи	8
2 Графы	10
2.1 Базовые алгоритмы на графах	10
2.2 Поиск кратчайших путей	10
3 Динамическое программирование и продвинутые темы	11
3.1 Динамическое программирование	11
3.2 Продвинутые темы	11

Общие положения

Темы курса разбиты на модули. Каждый модуль состоит из прохождения констестов (сдачи задач в виде программного кода в тестирующей системе). Констесты делятся на следующие типы.

- **Нулевые.** В этих констестах будет по 1-2 задачи, которые планируется разобрать на следующем занятии. Констесты по умолчанию весят 1%, дедлайн — начало следующего занятия.
- **Регулярные.** Проходят согласно программе лекций, стандартно длятся две недели. Содержат базовые задачи, решение которых позволяет закрепить текущие навыки.
- **Сквозные.** Проходят в течение всего модуля (или нескольких). Задачи сквозных констестов как правило охватывают сразу несколько тем или содержат нетривиальную часть, связанную с разработкой алгоритмов или выбора структуры данных.
- **Контрольные.** Контрольные констесты проходят в конце модуля, проходят во время занятий и длятся ровно три часа (или больше, если позволит расписание). Предполагается прохождение констеста в режиме реального времени, присутствуя в аудитории (если не сохранится онлайн-обучение). В случае невозможности личного присутствия, допускается онлайн прохождение констеста вне аудитории, но тогда за констест начисляется 50% баллов, а остальные 50% перераспределяются на зачётный констест.
- **Зачётный констест.** Констест, который проходит в самом конце курса. Скорее всего будет проходить в отдельное время. Те, кто пропускал очные контрольные констесты могут компенсировать на нём баллы за офлайн-сдачи. Также зачётный констест позволяет всем желающим набрать бонусные 15% к оценке. Прохождение зачётного констеста возможно исключительно при личном присутствии. Для студентов, не добравших балл до удовлетворительной оценки, этот констест является пересдачей (добор баллов может помочь получить удовлетворительную оценку).

Сдача констестов будет проходить в системе Codeforces. Курс строится из предположения того, что вы пишете на C++, но можно писать на любом языке программирования, который есть в системе Codeforces. Также

с этого года мы обеспечим поддержку Python (в теории возможны проблемы, если думаете, что что-то не так из-за использования питона — смело пишите в чат, будем разбираться). На первом же занятии мы обсудим проблемы с лимитами. Например, Python — достаточно медленный язык, но можно использовать PyPy. В прошлом прогоне проблем с питоном не было выявлено.

Помимо контестов будет ещё одно необязательное оценочное мероприятие — **подготовка задачи** в системе Polygon. Подготовка задачи позволяет достаточно хорошо почувствовать особенность алгоритмов. Задача будет готовиться группой (~ 3 человека). Нужно будет подготовить свою задачу и отревьюить задачу другой группы. За подготовленную задачу (+ревью) каждый участник группы получает 5% (возможно частичный балл, если нас не устроит качество) и ещё до 5% бонусных (будем поощрять хорошо сделанные задачи).

Оценивание

Регулярные и сквозные контесты будут в сумме давать 60% оценки, контрольные — 40%, зачётный — 15% (бонусные). Также могут быть бонусные баллы за некоторые задачи из регулярных и сквозных контестов.

Подготовка задачи в Полигон даёт от 5% до 10%. Это необязательное оценочное мероприятие, но мы очень рекомендуем принять в нём участие.

Уточнение. На каждый модуль (регулярные+сквозные) приходится по 20% оценки. Из них 12% оценки приходятся на регулярные контесты, а 8% на сквозные, не считая бонусов.

Для получения оценки за курс необходимо либо очное участие хотя бы в 50% контрольных контестов, либо участие в зачётном контесте, в котором возможно только очное участие.

Внимание! На этом курсе нулевая толерантность к списыванию. В случае обнаружения списывания факта списывания от набранных баллов отнимается 20% за каждый факт списывания (если поймали на списывании 2 раза — отнимается 40%) у всех участников.

Политика по некоторым вопросам

К списыванию не относится переиспользование собственного кода с более ранних посылок и использование кода с занятия (выкладывается в

репозиторий). Если будет обнаружено использование кода из сторонних источников, это будет приравнено к списыванию.

В чате можно обсуждать условия задач и технические проблемы, но нельзя обсуждать решения задач (рассказывать свой алгоритм, сложность задачи на ваш взгляд, описание своего кода). То есть нормально спросить «верно ли, что в условии числа меньше 10^7 », но ненормально спросить «Для этой задачи нужен линейный алгоритм с константной памятью»; нормально спросить «заработало ли у кого-то решение на питоне» и ненормально «я использовал на питоне линейный алгоритм с бинарным поиском у меня не сработало, у кого-то этот алгоритм зашёл».

Структура модулей

Модули будут состоять из нескольких тем, объединённых программистской тематикой. Считается, что сами алгоритмы уже изучены в рамках курса алгоритмов, поэтому базовая часть курса будет посвящена реализации изученных алгоритмов. Будет значительная часть «творческих» задач, предполагающих разработку алгоритмов и выбор структуры данных для задач, в которых исходно неясно какое решение нужно выбрать. Сложные задачи такого типа будут, в основном, в сквозных контекстах.

Контекст по каждой теме модуля стандартно длится две недели. При этом, на каждой неделе будут идти очные занятия, на которых будут обсуждаться аспекты реализации, изучаться темы, уместные на данный момент для практических задач. В процессе занятий будут также разбираться (как правило частично) задачи из текущих регулярных контекстов: обсуждаться, что именно и как планировалось реализовать.

Модуль заканчивается контрольным контекстом, в котором будет не очень много не очень сложных задач по пройденным темам, но которые необходимо сдать очно или хотя бы в онлайн-режиме.

План курса

Модуль 1

Базовые навыки

Время на модуль: 5 недель = 2 темы по 2 недели + контрольная

При изучении алгоритмов мы довольно быстро проскочили в самом начале целый ряд базовых вещей, не овладев которыми будет затруднительно двигаться дальше. Базовые алгоритмы включают в себя работу с циклами, массивами, написание рекурсивных процедур. Начало курса посвящено отработке этих базовых вещей на практике.

1.1 Введение. Работа с системой Codeforces, знакомство с C++

Время на тему: 1.5 занятия контест 2 недели

Темы на занятиях: Оценка времени работы программы и используемой памяти. Решение стандартных проблем, возникающих при сдаче задач на Codeforces. Имплементация жадных алгоритмов, работа с инвариантами

Темы по алгоритмам: жадные алгоритмы, индуктивные функции, онлайн-алгоритмы, рекурсивные алгоритмы, динамическое программирование сверху

1.1.1 Некоторые планируемые задачи

Жадность (+ одномерное ДП)

- Найти максимальное произведение, кратное 15

- Задачи типа LCS

Рекурсия (+ДП сверху)

- Бинарный поиск (интерактив в Полигоне)
- Quick sort
- Merge sort
- k-ая порядковая статистика

Рекурсия и итерация

- Расширенный алгоритм Евклида
- Итеративный Merge sort

1.2 Структуры данных

Время на тему: 1.5 недели (контеcт 2 недели)

Темы по алгоритмам: структуры данных

Темы на занятиях: абстракция, пишем на C++ (как на Си с классами), работа с классами и шаблонами, структуры данных из стандартной библиотеки шаблонов, амортизационный анализ, различия между C++ и Python, реализация очереди в Python, Декартовы деревья с реализацией на Python

1.2.1 Некоторые планируемые задачи

- Стековый калькулятор
- Реализация очереди с приоритетами через кучи
- Реализация деревьев поиска
- Реализация Декартова дерева по неявному ключу

Дополнительные темы

На четвёртой неделе модуля планируется разбор некоторых задач и обсуждение тем, нужных для сквозных констестов. В качестве дополнительной темы первого модуля, мы подробнее поговорим про амортизационный анализ и изучим алгоритмы обработки текста Кнута-Морриса-Пратта и Ахо-Корасик.

Модуль 2

Графы

Во второй и третий модуль входят темы из ядра курса алгоритмов, поэтому ожидается, что на занятиях придётся больше обсуждать задачи и разные аспекты связанные с реализацией. Если всё пойдёт более проще, чем ожидается — добавим дополнительных тем.

Как и в первом модуле пул задач для практикума будет близок по духу к содержанию курса алгоритмов, а потому здесь отдельно мы эти задачи не расписываем.

Время на модуль: $4 = 2 \times 1,5 + 1$

2.1 Базовые алгоритмы на графах

Время на тему: 1,5 недели

Темы по алгоритмам: Поиск в ширину, поиск в глубину, топологическая сортировка, динамика на DAG, Поиск мостов

2.2 Поиск кратчайших путей

Время на тему: 1,5 недели

Темы по алгоритмам: Дейкстра, Беллман-Форд, Флойд-Уоршелл, Прим и Крускал.

Модуль 3

Динамическое программирование и продвинутые темы

3.1 Динамическое программирование

Время на тему: 2 недели

3.2 Продвинутые темы

Время на тему: 2 недели

Темы на занятиях: Деревья отрезков, метод двоичных подъёмов