

Список вопросов к экзамену по курсу «Алгоритмы и структуры данных» (первый модуль)

Во всех вопросах нужно оценить время работы, используемую память, доказать корректность (если не сказано обратного).

1. \mathcal{O} -символика, мотивация, основные виды роста функций. Задача поиска элемента в множестве: наивное решение, решение сортировкой и бинарным поиском.
2. Вычисление чисел Фибоначчи: экспоненциальный рекурсивный алгоритм, полиномиальный алгоритм, более детальный анализ (оценка числа элементарных операций).
3. Метод «разделяй и властвуй». Сортировка слиянием. Мастер-теорема — основная теорема о рекуррентных соотношениях.
4. Сортировка слиянием. Нижняя оценка $\Omega(n \log n)$ на время работы сортировки сравнениями.
5. Сортировка подсчётом. Стабильная версия. Цифровая сортировка.
6. Интерфейс «очередь с приоритетами». Двоичная куча. Добавление элемента, извлечение минимума.
7. Двоичная куча. Изменение, удаление произвольного элемента. Построение кучи за линейное время. Сортировка кучей. Нижняя оценка на время работы извлечения минимума из кучи. Частичная сортировка кучей за $\mathcal{O}(n + k \log n)$.
8. Понятия вероятностного алгоритма, времени работы в среднем. Быстрая сортировка.
 - (a) Базовый алгоритм, оценка времени работы $\mathcal{O}(n^2)$.
 - (b) Эффективный алгоритм (inplace), оценка времени $\mathcal{O}(n^2)$, памяти, доказательство корректности.
 - (c) Анализ среднего времени работы.
 - (d) Массивы с малым количеством различных элементов, разбиение массива на три части. Средняя глубины рекурсии (без доказательства). Элиминация хвостовой рекурсии. IntroSort.
 - (e) Порядковые статистики: нахождение за линейное в среднем время. Частичная сортировка за $\mathcal{O}(n + k \log k)$.
9. Динамическое программирование. Общие принципы динамического программирования. Задачи (во всех нужно рассказать про восстановление ответа):
 - (a) Кратчайший путь в ациклическом ориентированном графе.
 - (b) Дискретная задача о рюкзаке с повторениями.
 - (c) Дискретная задачи о рюкзаке без повторений.
 - (d) Наибольшая возрастающая подпоследовательность за $\mathcal{O}(n^2)$.
 - (e) Наибольшая общая подпоследовательность за $\mathcal{O}(nm)$ [задача 4.1.4a].
 - (f) Редакционное расстояние.
 - (g) Паросочетание максимального веса в дереве [задача 4.1.2a].
 - (h) Задача коммивояжёра. Детали реализации на языке программирования.
10. Общая идея жадных алгоритмов. Задачи:
 - (a) Покрытие точек единичными отрезками.
 - (b) Задача о выборе заявок.
 - (c) Непрерывная задача о рюкзаке.
 - (d) Код Хаффмана: постановка задачи, алгоритм, доказательство корректности, оценка времени работы и памяти.

11. Графы: определения, способы хранения. Поиск в глубину (DFS). Доказательство корректности.
- (a) Подсчёт числа компонент связности. В ориентированном и неориентированном случае: поиск цикла (доказательство только в одну сторону: если считаем, что нашли цикл, то он действительно есть), классификация рёбер.
 - (b) Топологическая сортировка за $\mathcal{O}(|V| + |E|)$: с помощью очереди и с помощью поиска в глубину [обсуждалось на практике].

Формат проведения экзамена

Сначала все пишут тест. Тест определяет вашу максимальную оценку за экзамен (см. ниже).

Затем вы подключаетесь в выделенное вам время (ближе к экзамену появится таблица для записи), между тестом и временем подключения у вас перерыв. При подключении вам выдаётся несколько вопросов из списка выше, вы *без подготовки* их отвечаете одному экзаменатору.

Пусть s_t — оценка за тест, а s_e — оценка за устный ответ. Тогда оценка за экзамен — $\min(s_t, s_e)$.

В течение всего экзамена (как теста, так и устного ответа) пользоваться дополнительными материалами (конспектами, книгами, интернетом и т. д.) не разрешается.

В течение всего экзамена (как теста, так и устного ответа) нужно будет демонстрировать свой экран и демонстрировать свой стол (должно быть видно вас и пространство перед вами). Стабильно должен работать интернет. Подумайте заранее, как это организовать.

Во время ответа вам наверняка нужно будет что-нибудь писать так, чтобы это видел экзаменатор. Подумайте, как вам будет удобнее это делать — это может быть графический редактор, текстовый редактор. Если хотите писать формулы в \LaTeX , можно использовать какой-нибудь сервис для онлайн-отрисовки формул (или просто Overleaf, можно в режиме Rich Text).