

1 Сортировки

- 1.1. Даны два отсортированных по неубыванию массива a и b . Определите, есть ли в них одинаковые числа. Время $O(n)$.
- 1.2. Даны два отсортированных по неубыванию массива a и b . Найдите такие i и j , что разница $|a_i - b_j|$ минимальна. Время $O(n)$.
- 1.3. Даны два отсортированных по неубыванию массива a и b и число S . Найдите такие i и j , что сумма $a_i + b_j = S$. Время $O(n)$.
- 1.4. Даны два отсортированных по неубыванию массива a и b . Найдите число пар (i, j) , таких, что $a_i = b_j$. Время $O(n)$.
- 1.5. Даны два отсортированных по неубыванию массива a и b . Найдите число пар (i, j) , таких, что $a_i > b_j$. Время $O(n)$.
- 1.6. Дан массива a . Пара (i, j) , такая, что $i < j$ и $a_i > a_j$ называется инверсией. Пусть в массиве длины n ровно k инверсий. Докажите, что сортировка вставками работает за $O(n + k)$.
- 1.7. Дан массива a . Найдите число инверсий в нем. Время $O(n \log n)$
- 1.8. Покажите, что сортировка слиянием является устойчивой (то есть, не меняет порядок равных элементов).
- 1.9. Покажите, как сделать сортировку слиянием снизу вверх, без рекурсии.

2 Рекуррентные соотношения

- 2.1. Докажите по индукции, что если $T(n) = 2T(n/2 + 20) + n$, то $T(n) = O(n \log n)$.
- 2.2. Докажите по индукции, что если $T(n) = 2T(n/2 + \log n) + n$, то $T(n) = O(n \log n)$.
- 2.3. Докажите по индукции, что если $T(n) = \log n \cdot T(n/\log n) + n$, то $T(n) = O(n \log n)$.
- 2.4. Докажите по индукции, что если $T(n) = 2T(n/2) + n$, то $T(n) = \Omega(n \log n)$ (оценка снизу).
- 2.5. Докажите по индукции, что если $T(n) = 2T(\sqrt{n}) + 1$, то $T(n) = O(\log n)$.
- 2.6. Найдите асимптотическую оценку на $T(n)$, если $T(n) = T(n - a) + T(a) + n$ (a — константа).
- 2.7. Для каждой из приведенных программ и функций оцените время ее работы

a)

```
for i in range(n):
    j = 0
    while j * j < i:
        j += 1
```

b)

```
for i in range(n):
    j = i
    while j > 0:
        j = j // 2
```

c)

```
def f(n):
    if n == 0:
        return 1
    else
        return 5 * f(n // 3)
```

d)

```
def f(n):
    if n == 0:
        return 1
    else
        return f(n // 3) + f(n // 3)
```

3 Куча, сортировка кучей

- 3.1. Проиллюстрируйте работу алгоритма сортировки кучей на примере массива [5, 2, 7, 3, 4, 2, 8].
- 3.2. Пусть в куче лежат числа от 1 до 255, по одному разу каждое. Какое минимальное число может лежать в куче на самом нижнем уровне?
- 3.3. Пусть в куче лежат числа от 1 до n , по одному разу каждое. В каком случае операция `removeMin` будет работать минимальное, а в каком — максимальное время?
- 3.4. Пусть дерево кучи организовано таким образом, что у каждой вершины (кроме нижнего слоя) не два ребенка, а три. Какие номера будут у детей вершины i в этом случае?
- 3.5. Пусть дерево кучи организовано таким образом, что у каждой вершины (кроме нижнего слоя) d детей (при этом d — не константа, а параметр). За какое время работают основные операции над кучей в этом случае? Приведите оценку зависимости от n и d .
- 3.6. Как из двух куч сделать структуру данных, которая одновременно может искать и удалять как максимум, так и минимум.
- 3.7. Покажите, что если делать кучу из массива, просто добавляя элементы один за другим в кучу, то время работы в худшем случае будет $\Omega(n \log n)$ (внимание, оценка снизу).
- 3.8. Есть k отсортированных массивов, содержащих в сумме n элементов. Слейте их в один отсортированный массив за время $O(n \log k)$.
- 3.9. Приведите пример массива, на котором сортировка кучей работает за $O(n)$.

4 Быстрая сортировка

- 4.1. Приведите пример, когда быстрая сортировка работает за $\Omega(n^2)$, если в качестве разделителя выбирается: а) самый левый элемент отрезка, б) самый правый элемент отрезка, в) центральный элемент отрезка ($a[(1 + r) / 2]$).
- 4.2. То же самое, но в качестве разделителя выбирается медиана из этих трех элементов.
- 4.3. Предложите способ, который бы защищал быструю сортировку от плохих тестов. То есть, пусть быстрая сортировка работает за время $A \cdot n \log n$. Предложите модификацию, которая в среднем тоже работает примерно за $A \cdot n \log n$, а в худшем — за $B \cdot n \log n$.
- 4.4. Как модифицировать быструю сортировку, чтобы даже в худшем случае требовалось $O(\log n)$ дополнительной памяти?
- 4.5. У вас есть n болтов и n подходящих к ним гаек, все болты (и, соответственно, все гайки) разного диаметра. Посмотрев на два болта (или две гайки), сложно понять, какой больше, а какой меньше, поэтому единственная операция, которая у вас есть — взяв какой-то болт и какую-то гайку, сравнить их диаметры (если не пролезает — значит болт больше, если болтается — значит меньше). Найдите для каждого болта подходящую гайку за $O(n \log n)$.

5 Разные задачи про сортировки и не только

- 5.1. В отсортированном массиве размера n изменили k элементов (неизвестно, каких именно). Отсортируйте полученный массив за $O(n + k \log k)$.
- 5.2. Как с помощью генератора случайных чисел получить случайную перестановку? Нужно, чтобы каждая перестановка появлялась с вероятностью $1/n!$.

- 5.3. Дан массив из n чисел. Необходимо для каждого элемента найти число элементов, которые меньше его. Время работы $O(n \log n)$.
- 5.4. Дано два массива: a и b . Найдите такие i и j , что $a_i < a_j$ и $b_i < b_j$, или скажите, что найти невозможно. Время работы $O(n \log n)$.
- 5.5. В ряд стоят n ящиков. Необходимо отсортировать их по номерам. У вас есть кран, который умеет делать одну команду: `swap(i, j)`, которая меняет местами ящики i и j . Отсортируйте ящики, совершив минимальное число операций. Время работы $O(n \log n)$.
- 5.6. Покажите, что любую сортировку можно сделать устойчивой, не поменяв асимптотику работы и потратив $O(n)$ дополнительной памяти.
- 5.7. На прямой живут n друзей, i -й друг живет в точке x_i . Они хотят встретиться в одной точке. Помогите им найти такую точку, чтобы суммарное расстояние, которое они пройдут, было бы минимально.
- 5.8. Дан массив, а) проверить что все элементы в нем различны. б) проверить что все элементы в нем равны.
- 5.9. Даны два массива a и b , состоящих из целых чисел. Проверьте, можно ли в массиве a выбрать k чисел, а в массиве b — m чисел так, что любое число, выбранное в первом массиве, строго меньше любого числа, выбранного во втором массиве.

6 К-я порядковая статистика

- 6.1. Нужно получить первые k элементов массива в отсортированном порядке. За какое минимальное время это можно сделать?
- 6.2. Что будет, если в алгоритме Блюма, Флойда, Пратта, Ривеста и Тарьяна заменить константу 5 на 3 или 7?

7 Цифровая сортировка

- 7.1. Дан массив из n чисел от 1 до k , разработайте структуру данных, которая за $O(1)$ отвечает на запросы вида «Сколько в массиве элементов от a до b ?». Время на предподсчет $O(n + k)$.
- 7.2. Как с помощью цифровой сортировки сортировать строки (например, состоящие только из латинских букв) в лексикографическом порядке? За какое время будет работать такая сортировка?
- 7.3. Возьмем массив a из n элементов, каждый из которых — это число 1 до n . Циклический сдвиг номер i — это последовательность $a_i, a_{i+1}, \dots, a_{n-1}, a_0, a_1, \dots, a_{i-1}$. Предложите алгоритм сортировки циклических сдвигов в лексикографическом порядке за время $O(n^2)$.
- 7.4. Предложите алгоритм сортировки циклических сдвигов за $O(n \log n)$. Указание: зная порядок на подстроках длины L порядок на подстроках длины $2L$ можно восстановить за $O(n)$.
- 7.5. Пусть известно, что массив длины n из чисел от 1 до n получен с помощью генератора случайных чисел, каждое число независимо получено с помощью равномерного распределения. Предложите модификацию алгоритма сортировки подсчетом, который сортирует данный массив за $O(n)$ используя лишь $O(\sqrt{n})$ дополнительной памяти (обе оценки должны выполняться в среднем).

8 Двоичный поиск

- 8.1. Дан отсортированный массив, отвечать на запросы: «Сколько элементов равны X ?».
- 8.2. Дан массив положительных чисел, отвечать на запросы: «Какое максимальное число элементов из начала массива можно взять, чтобы их сумма была не больше X ?».
- 8.3. Задан массив, полученный циклическим сдвигом из отсортированного по возрастанию. Все элементы массива различны. Требуется за $O(\log n)$ найти в нем заданный элемент.
- 8.4. Задан массив, полученный приписыванием одного отсортированного по возрастанию массива в конец другому отсортированному по возрастанию. Все элементы массива различны. Требуется за $O(\log n)$ найти в нем заданный элемент.
- 8.5. Задан массив, полученный приписыванием отсортированного по убыванию массива в конец отсортированному по возрастанию. Все элементы массива различны. Требуется за $O(\log n)$ найти в нем заданный элемент.
- 8.6. Задан массив, полученный приписыванием отсортированного по убыванию массива в конец отсортированному по возрастанию и затем циклическим сдвигом получившегося массива. Все элементы массива различны. Требуется за $O(\log n)$ найти в нем заданный элемент.
- 8.7. Пусть выполняется целочисленный двоичный поиск с начальными значениями L, R . Преложите алгоритм определения за $O(\log n)$ по заданным значениям l и r , могут ли они возникнуть в процессе двоичного поиска.
- 8.8. В игре есть n типов ресурсов, для постройки одного юнита требуется a_i единиц ресурса i для всех i от 1 до n . У Пети есть b_i единиц ресурса i и еще d единиц золота. Одну единицу золота можно обменять на d_i единиц ресурса i . Сколько юнитов может построить Петя?
- 8.9. Покажите, что если в условии на функцию в троичном поиске заменить строгое убывание и возрастание на нестрогое, то найти минимум быстрее, чем за $O(n)$ нельзя.
- 8.10. Даны n кенгуру с сумками. У каждого кенгуру есть размер (целое число). Кенгуру может поместиться в сумке другого кенгуру тогда и только тогда, когда размер кенгуру-носителя как минимум в два раза больше размера кенгуру-пассажира. Каждый кенгуру может нести не более одного кенгуру, а кенгуру-пассажир не может носить никаких кенгуру. Кенгуру-пассажира не видно, когда он в сумке кенгуру-носителя. Пожалуйста, разработайте такой план рассадки кенгуру, чтобы было видно как можно меньше кенгуру.

9 Связные списки

- 9.1. Придумайте, как развернуть связный список за время $O(n)$ с $O(1)$ дополнительной памяти.
- 9.2. Для экономии памяти в двусвязных списках иногда вместо двух ссылок хранят только их битовый XOR. (например, если $\text{prev}[i] = 5$, $\text{next}[i] = 3$, то вместо них храним $\text{prevnext}[i] = 6$). Как работать с таким списком?
- 9.3. В функциональных языках программирования нет переменных, то есть когда вы, например, создаете узел списка, поменять в нем ссылку на следующий или предыдущий элемент уже не получится. Как сделать функциональный стек?
- 9.4. Дан набор из n элементов, в каждом есть ссылка на следующий. Проверьте, правда ли эти элементы образуют линейный список (менять ссылки нельзя). (Время $O(n)$, память $O(1)$).
- 9.5. Дан набор из n элементов, в каждом есть ссылка на следующий. Проверьте, правда ли эти элементы образуют кольцевой список (менять ссылки нельзя). (Время $O(n)$, память $O(1)$)

- 9.6. Дан набор из n элементов, в каждом есть ссылка на следующий и предыдущий. Проверьте, правда ли эти элементы образуют несколько кольцевых списков (менять ссылки нельзя). (Время $O(n)$, память $O(1)$)
- 9.7. Дан набор из n элементов, в каждом есть ссылка на следующий и предыдущий. Проверьте, правда ли эти элементы образуют несколько связанных списков и, если да, объедините их в один (в любом порядке). (Время $O(n)$, память $O(1)$)
- 9.8. Слить два отсортированных односвязных списка в один за время $O(n)$ с $O(1)$ дополнительной памяти.
- 9.9. Придумайте, как отсортировать связный список за время $O(n \log n)$ с $O(1)$ дополнительной памяти.
- 9.10. Придумайте, как хранить связный список, чтобы его можно было развернуть за $O(1)$ (с сохранением всех остальных операций).
- 9.11. Есть таблица $n \times n$. С ней проводят m операций следующего вида: вырезать прямоугольный кусок, развернуть его на 180° и вклеить на то же место. Выведите состояние таблицы после всех операций. Время $O(nm)$.
- 9.12. Есть полоска бумаги, разбитая на n одинаковых квадратных клеток. На клетках написаны числа от 1 до n . С полоской совершают m действий следующего вида: отсчитывают x_i клеток от левого конца полоски и сгибают полоску в этом месте, кладя правую половину поверх левой. Выведите финальное состояние полоски после всех сгибов. Время $O(m + n)$.

10 Стеки и очереди

- 10.1. Дек — это структура данных, объединяющая стек и очередь, в ней можно добавлять и удалять элементы с обоих концов. Реализуйте дек на базе массива. Время работы $O(1)$.
- 10.2. Добавьте в стек и очередь операцию `getSum()`, возвращающую сумму элементов в стеке/очереди. Время работы $O(1)$. Дополнительная память $O(1)$.
- 10.3. Добавьте в стек операцию `getMin()`, возвращающую минимальный элемент в стеке. Время работы $O(1)$. Дополнительная память $O(size)$ ($size$ — число элементов в стеке).
- 10.4. Научитесь вычислять выражения в префиксной записи (это как постфиксная, только наоборот, например, выражение $4 - ((1 + 2) * 3)$ в префиксной записи выглядит так: $- 4 * + 1 2 3$). При этом читать строку с выражением нужно слева направо (так бывает нужно делать, например, если строку вам передают по сети и у вас нет памяти, чтобы хранить ее целиком).
- 10.5. Научитесь вычислять выражения в инфиксной записи со скобками (обычные выражения). Для простоты можно считать, что в выражении проставлены все скобки (то есть внутри скобок вычисляется только один оператор, например так можно: $(4 - ((1 + 2) * 3))$, а так — нет: $(1 + 2 + 3)$).
- 10.6. Научитесь по выражению в префиксной записи строить выражение в постфиксной записи.
- 10.7. Научитесь по выражению в инфиксной записи строить выражение в постфиксной записи.
- 10.8. Был массив чисел, все соседние числа были различны. С этим массивом несколько раз проделали операцию: вставить в массив два одинаковых числа рядом. По конечному состоянию массива восстановите его исходное состояние.
- 10.9. Дан массив из целых чисел. Для каждого элемента найдите ближайший элемент слева, меньший его.

10.10. Дано начальное и конечное состояние стека. Какое минимальное число операций могло произойти?

10.11. (Codeforces 358C) Скоро у Димы день рождения! Большой праздник! Сережа подарит Диме свое отсутствие в комнате и не будет мешать Диме и Инне праздновать день рождения. Инна подарит Диме стек, очередь и дек.

Своим подарком Инна хочет проверить, насколько Дима хороший программист. Для этого она будет поочередно давать Диме команды. Команды бывают двух типов:

- Добавить заданное число в один из контейнеров. Причем в очередь и стек можно добавлять элементы только в конец, в дек можно добавлять элементы как в начало, так и в конец.
- Извлечь из не более чем трех различных контейнеров по числу. Все извлеченные числа сказать Инне, после чего опустошить все контейнеры. Причем в контейнере очередь можно извлекать числа только из начала, в стеке — только из конца, в деке — из начала и из конца. Нельзя извлекать числа из пустых контейнеров.

Каждый раз, когда Дима выполняет команду второго типа, Инна целует Диму некоторое число (возможно, ноль) раз. Дима знает Инну как облупленную, он уверен, что это число равно сумме извлеченных из контейнеров чисел на данной операции.

Как уже было сказано, Дима знает Инну как облупленную, ему известно, какие команды Инна будет давать Диме и в каком порядке. Помогите Диме найти стратегию, используя которую, он получит как можно больше поцелуев на свой день рождения!

11 Амортизационный анализ

11.1. Проанализировать саморасширяющийся массив, если расширение происходит в A раз ($1 < A$). Какое значение A лучше выбирать на практике?

11.2. Пусть выделение массива памяти любого размера происходит за время $O(1)$. Разработайте вектор с истинной (не амортизированной) стоимостью всех операций $O(1)$ и памятью $O(n)$. (Подсказка: надо распределить операцию копирования массива на несколько операций).

11.3. Добавьте в очередь операцию `getMin()`, возвращающую минимальный элемент в очереди. Амортизированное время работы $O(1)$.

11.4. Реализуйте дек на трех стеках с амортизированным временем работы всех операций $O(1)$.

11.5. Мультистек — это бесконечная последовательность стеков, максимальный размер i -го стека равен 3^i . Изначально все стеки пустые. Новый элемент добавляется в первый стек. Если он уже заполнен, то все его элементы перекладываются во второй, если он также заполнен, то его элементы перекладываются в третий, и т. д. Покажите, что амортизированное время добавления элемента в мультистек $O(\log n)$.

11.6. Покажите, как сделать, чтобы в мультистеке стеки были всегда отсортированы (при этом амортизированное время добавления элемента все еще $O(\log n)$).

11.7. Покажите, как в полученной структуре найти и удалить минимальный элемент за $O(\log n)$. (В результате должна получиться структура данных на базе стеков, в которую можно добавлять и удалять минимум за $O(\log n)$).

11.8. Добавьте битовому счетчику операцию `clear`, сбрасывающую его в 0. Амортизационная стоимость операции должна быть $O(1)$.

- 11.9. Пусть битовый счетчик хранит числа в фибоначчиевой системе счисления (это когда стоимость i -го разряда равна i -му числу Фибоначчи: 1, 1, 2, 3, 5, ...). Покажите, как в таком счетчике сделать операции увеличения на 1 и уменьшения на 1.
- 11.10. Структура данных «буферное окно» используется в текстовых редакторах и позволяет перемещать курсор по тексту и добавлять/удалять символы в позиции курсора. Для этого текст хранится в виде строки, в которой на позиции курсора есть окно из незаполненных элементов. Когда пользователь вводит символ, он добавляется на первое незаполненное место и размер буфера уменьшается на 1, когда удаляется — наоборот. Как нужно работать с такой структурой, чтобы амортизированное время добавления/удаления одной буквы было $O(1)$? Чем эта структура лучше/хуже, чем связный список?

12 Система непересекающихся множеств

- 12.1. Добавьте в СНМ операции $\text{getMin}(x)$, $\text{getMax}(x)$, $\text{getSum}(x)$, возвращающие минимум, максимум и число элементов в множестве x .
- 12.2. Дан массив a длины n , заполненный нулями, Делают два вида запросов: 1) $a_i := 1$ 2) найти число непрерывных отрезков из единиц.
- 12.3. Дан массив a длины n , заполненный нулями, Делают два вида запросов: 1) $a_i := 1$ 2) найти ближайший к i ноль.
- 12.4. Дан массив a из n положительных чисел. Найдите отрезок с максимальным значением произведения (сумма \times минимум).
- 12.5. Дан массив a из n положительных чисел. Для каждого числа найдите, для сколько разных отрезков оно является минимумом.
- В следующих задачах используются термины из теории графов, которые мы пока не проходили. Но вы не пугайтесь, они все несложные, если не знаете какой-то из них — загуглите.*
- 12.6. В изначально пустой граф из n вершин добавляются одно за другим m ребер. После каждого добавления найдите размер самой большой компоненты связности (по числу вершин).
- 12.7. Есть пустой граф из n вершин. Делают два вида запросов: 1) добавить ребро 2) найти число ребер в компоненте связности x .
- 12.8. Есть пустой граф из n вершин. Делают два вида запросов: 1) добавить ребро 2) найти число компонент связности, являющихся деревьями.
- 12.9. Есть пустой граф из n вершин. Делают два вида запросов: 1) добавить ребро 2) найти число компонент связности, являющихся циклами.
- 12.10. Есть пустой граф из n вершин. Делают два вида запросов: 1) добавить ребро 2) найти число компонент связности, являющихся двудольными графами.
- 12.11. Докажите амортизированную оценку $O(\log n)$ для эвристики сжатия путей (без других эвристик).
- 12.12. Для кластеризации изображений решают следующую задачу. Есть картинка $n \times m$ пикселей (для простоты будем считать, что каждый пиксель задается одним числом). Мы хотим выделить на ней части примерно одинакового цвета. Для этого нужно разбить картинку на k связных областей так, чтобы минимальная разность значений пикселей на границе областей была максимально возможной.

13 Динамическое программирование

- 13.1. В одном здании крышу поддерживают n металлических балок, стоящих в ряд. Сторож Петрович давно хочет сдать их в металлолом, но он опасается, что если спилить две подряд стоящие балки, то что-нибудь сломается. Вес i -й балки равен a_i . Сколько металлолома может безопасно сдать Петрович? Время $O(n)$.
- 13.2. Задано уравнение вида $A + B = C$, где A , B и C — неотрицательные целые числа длины n , в десятичной записи которых некоторые цифры заменены знаками вопроса. Например: $?2 + 34 = 4?$. Требуется подставить вместо знаков вопроса цифры, чтобы это равенство стало верным. Время $O(n)$.
- 13.3. У Васи есть калькулятор, который умеет выполнять три операции: прибавить 1, умножить на 2 и умножить на 3. Какое наименьшее число операций необходимо для того, чтобы получить из числа 1 число n . Время $O(n)$.
- 13.4. Коля сочиняет специфическую музыку. В работе он использует только m нот, причем две последовательные ноты отличаются ровно на 1. Сколько различных композиций из n нот он может сочинить? Время $O(nm)$.
- 13.5. Вася любит ходить в кино и решать домашку. Еще Вася любит разнообразие, поэтому он никогда не делает одно и то же два дня подряд. Для n последовательных дней известно, есть ли в этот день интересное кино и есть ли в этот день интересная домашка. Какое минимальное число дней Вася будет сидеть без дела? Время $O(n)$.
- 13.6. Футболист снова бежит из клетки $(0, 0)$ в клетку (n, m) . На этот раз он не очень торопится, поэтому ему не обязательно идти кратчайшим путем, но при этом он хочет сделать не более k шагов. Найдите число способов сделать это. Время $O(nmk)$.
- 13.7. В очереди за концерт Ромы Желудя стоит n школьников. Чтобы увеличить скорость движения очереди, две подряд идущие школьницы могут договориться, и та, что идет раньше в очереди купит два билета: на себя и на следующую за ней. Школьница i тратит a_i секунд на покупку одного билета и b_i секунд на покупку двух билетов. За какое минимальное время все смогут купить билеты? Время $O(n)$.
- 13.8. Дана последовательность, найдите максимальную по длине подпоследовательность из последовательных чисел (например, 5, 6, 7, 8). Время $O(n)$.
- 13.9. Даны две **перестановки** чисел от 1 до n . Найдите их наибольшую общую возрастающую подпоследовательность (да, одновременно и общую, и возрастающую). Время $O(n^2)$.
- 13.10. Депутаты приняли закон, по которому все имена, которые даются детям, должны соответствовать шаблону, соответствующему региону. Шаблон состоит из букв, а так же символов «?» и «*». Символ «?» можно заменить на одну любую букву, символ «*» — на любую строку (в том числе пустую). Проверьте, что строка соответствует шаблону. Время $O(nm)$ (n и m — длина строки и шаблона).
- 13.11. Тяжело, когда ~~законы принимают идиоты~~ родители ребенка прописаны в разных регионах, ведь тогда имя ребенка должно соответствовать сразу двум шаблонам. Найдите самое короткое подходящее имя или скажите, что такого нет. Время $O(nm)$ (n и m — длины шаблонов).
- 13.12. Как изменится решение задачи о редакционном расстоянии если операции добавления, удаления и изменения имеют разные стоимости?
- 13.13. Петя и Вася выиграли на олимпиаде n призов, суммарной стоимостью r . Они хотят разделить их между собой так, чтобы разница в суммарной стоимости была минимальна. Время $O(nr)$.

- 13.14. Петя и Вася опять выиграли на олимпиаде, на этот раз $2n$ призов, суммарной стоимостью r . Они хотят разделить их между собой так, чтобы каждому досталось по n призов, а разница в суммарной стоимости была минимальна. Время $O(n^2r)$.
- 13.15. Петя и Вася позвали Колю и выиграли еще одну олимпиаду, опять получив n призов суммарной стоимостью r . Они хотят разделить их между собой так, чтобы разница между максимальной и минимальной долей была минимальна. Время $O(nr^2)$.
- 13.16. У мага есть n заклинаний и m единиц маны. Заклинание i тратит c_i маны и наносит врагу урон d_i . Убейте монстра с h здоровьем, потратив как можно меньше маны. Время $O(nm)$.
- 13.17. У Пети есть n книг, отсортированных по названию. Книга i имеет высоту h_i . Петя хочет сложить книги в шкаф в отсортированном порядке: первые несколько книг стоят на первой полке, следующие несколько на второй, и т.д. На каждую полку помещается не более m книг. Высота полки равна высоте максимальной книге на полке. Помогите Пете распределить книги по полкам так, чтобы суммарная высота полок была минимальной. Время $O(nm)$.
- 13.18. Деревня Гадюкино представляет собой n домов, расположенных вдоль прямой дороги. Координата i -го дома a_i . Компания «интернет в каждый дом» хочет поставить в деревне несколько вайфай-точек, покрывающих все дома. Точка с радиусом покрытия r стоит $A + Br^2$ рублей. Найдите минимальную стоимость необходимых точек. Время $O(n^2)$.
- 13.19. Есть n коров на прямой дороге, известны их координаты. Идет дождь, и нужно укрыть их зонтами. Есть m типов зонтов, i -й зонт стоит c_i рублей и может накрыть коров в радиусе r_i . Сколько денег нужно потратить чтобы укрыть всех коров? Время $O(nm)$.
- 13.20. Петя и Вася играют в игру: в ряд выложены n карточек, на i -й карточке написано число a_i . За один ход можно взять одну, две или три карты с конца ряда. Игра заканчивается, когда нет больше карт. Выигрывает тот, у кого в конце игры сумма чисел на картах максимальна. Кто выиграет при правильной игре? Время $O(n)$.
- 13.21. Компания «Кака-кола» устроила акцию: нужно собрать крышечки с p звездами в сумме, чтобы получить приз. У Пети есть m крышечек с одной звездой, n с двумя и k с тремя. Сколько призов сможет получить Петя? Время $O(mnk)$.
- 13.22. Есть строка. Удалить минимальное число букв, чтобы получился палиндром. Время $O(n^3)$.
- 13.23. Пете нужно посчитать произведение n матриц. Матрица i имеет размер $a_i \times a_{i+1}$. Как вы знаете, чтобы посчитать произведение матриц $a \times b$ и $b \times c$, нужно сделать примерно abc действий. Помогите Пете выполнить все умножения в таком порядке, чтобы суммарное число действий было минимально. Время $O(n^3)$.
- 13.24. Вам нужно распилить бревно длины L на несколько частей. Точки распила уже отмечены, всего есть n точек, расстояние от левого конца до i -й точки распила a_i . Лесопилка за один раз может сделать один распил, при этом стоимость равна длине бревна, которое нужно распилить. В каком порядке нужно делать распилы, чтобы суммарная стоимость была минимальна? Время $O(n^3)$.
- 13.25. Чтобы сжать длинную строку, используется следующий формат: строка $D(X)$ означает, что строку X надо повторить D раз. Такие конструкции могут вкладываться друг в друга, например «3(2(A)2(B))C» разжимается в «AABVAABVAABVBC». Для данной строки длины n найдите самое короткое представление в таком виде. Время $O(n^3)$.
- 13.26. Деревня Гадюкино представляет собой n домов, расположенных вдоль прямой дороги. Координата i -го дома a_i . В деревне планируется построить k бомбоубежищ. Расставьте их так, чтобы суммарное расстояние от каждого из домов до ближайшего к нему бомбоубежища было минимально. Время $O(n^2k)$.

В следующих заданиях время работы не указано специально, придумайте сами. Чем быстрее получится, тем лучше.

- 13.27. В одной игре есть n героев, у каждого есть три параметра: сила, ловкость и интеллект, выражающиеся целыми числами от 1 до m . Пете нужно выбрать команду из k героев. Он считает, что для того, чтобы команда была более сбалансированной, нужно, чтобы произведение суммарной силы, суммарной ловкости и суммарного интеллекта было максимально. Помогите Пете составить команду.
- 13.28. Есть набор из n чисел. Разбить его на максимальное число наборов, в каждом из которых сумма всех чисел больше или равна k .
- 13.29. Есть набор из n чисел. Разбить его на минимальное число наборов, в каждом из которых НОД всех чисел больше 1.
- 13.30. У доктора Хауса есть n вариантов, чем может болеть пациент. Он может сделать m разных анализов, анализ i дает положительный результат, если $a_{ij} = 1$, где j — номер болезни пациента. Хаус хочет сделать минимальное число анализов, так чтобы гарантированно поставить диагноз.
- 13.31. Коля решает задачу о рюкзаке без стоимостей (нужно набрать максимальный вес не больший S) жадным алгоритмом, он работает так. Пусть сейчас уже взяты некоторые предметы, и их суммарный вес равен T , тогда на следующем шаге алгоритма Коля возьмет предмет максимального веса среди тех, для которых $T + w_i \leq S$. Если таких нет, то алгоритм заканчивает работу. Пусть этот алгоритм получил набор с суммой ANS , а оптимальный ответ равен OPT . На каком тесте отношение ANS/OPT будет минимально?
- 13.32. Как изменится задача о замощении доминошками, если доски имеют размер 3×1 ?
- 13.33. Найти число замощений прямоугольника $n \times m$ уголками 2×2 (квадрат без одной клетки).
- 13.34. Найти число замощений правильного шестиугольника со стороной n ромбами со стороной 1.
- 13.35. Найти число способов покрасить клетки прямоугольника $n \times m$ в черный и белый цвета так, чтобы не было квадрата 2×2 , покрашенного в один цвет.
- 13.36. Найти число способов обойти все клетки прямоугольника $n \times m$, посетив каждую клетку ровно один раз и оказавшись в конце в той же клетке, где начал. Перемещаться можно только в соседние клетки по стороне.
- 13.37. Дана шахматная доска $n \times m$, некоторые клетки которой удалены. Поставить на доску максимальное число королей так, чтобы они не били друг друга.
- 13.38. Та же задача, но про коней.
- 13.39. Та же задача, но про слонов.
- 13.40. Та же задача, но про ладей.
- 13.41. Та же задача, но про ферзей.

14 **NEW!** Хеш-таблицы

- 14.1. Пусть число возможных ключей $U > nm$, где n — число элементов в хеш-таблице, а m — ее размер. Покажите, что для любой хеш-функции в худшем случае возможно попадание всех n элементов в одну ячейку хеш-таблицы.

- 14.2. Дан массив a из n чисел. Посчитайте число отрезков, на которых битовый `xor` всех чисел равен x . Время $O(n)$.
- 14.3. Добавьте в хеш-таблицу возможность пробежаться по всем ее элементам в том порядке, в котором они добавлялись, за $O(n)$.
- 14.4. Как сделать хорошую хеш-функцию для массива?
- 14.5. Как сделать хорошую хеш-функцию для двумерного массива?
- 14.6. Как сделать хорошую хеш-функцию для мультимножества?
- 14.7. Как сделать хорошую хеш-функцию для дерева?
- 14.8. Сливаемые множества. Добавьте в хеш-таблицу операцию `merge`, которая объединяет два множества в одно. Амортизированное время работы $O(\log n)$.
- 14.9. Добавьте в очередь (ну или сразу в дек) операцию `countUnique`, которая возвращает число различных элементов в очереди. Время $O(1)$.
- 14.10. Есть множество вещественных чисел. Нужно делать две операции за $O(1)$: 1) добавить число, 2) найти какое-нибудь число в диапазоне $(x - \varepsilon, x + \varepsilon)$ (или сказать, что ни одного числа нет) ε одинаковый для всех запросов.
- 14.11. Посмотрите, как устроен метод `Long.hashCode()` в Java (ссылку не дам, сами загуглите). Как быстро сгенерировать много `Long`ов с одинаковым хешом? Проверьте, что если положить их в `HashSet<Long>`, он будет работать долго.
- 14.12. Посмотрите, как устроен метод `String.hashCode()` в Java. Как быстро сгенерировать много строк с одинаковым хешом? Проверьте, что если положить их в `HashSet<String>`, он будет работать долго.
- 14.13. **NEW!** Реализуйте хеш-таблицу со списками, на n ячеек, и положите в нее n элементов. Запустите 100 раз с разными хеш-функциями и найдите среднюю длину самого длинного списка. Прodelайте эти операции для $n = 100, 1000, 10000, 100000, 1000000, 10000000$.
- 14.14. **NEW!** Реализуйте хеш-таблицу с открытой адресацией, на $2n$ ячеек, и положите в нее n элементов. Запустите 100 раз с разными хеш-функциями и найдите среднюю длину самого большого блока из единиц. Прodelайте эти операции для $n = 100, 1000, 10000, 100000, 1000000, 10000000$.
- 14.15. **NEW!** Реализуйте хеш-таблицу с хешированием кукушки, на $2n$ ячеек, и положите в нее n элементов. Запустите 100 раз с разными хеш-функциями и найдите среднее максимальное число переносов при добавлении элемента. Прodelайте эти операции для $n = 100, 1000, 10000, 100000, 1000000, 10000000$.
- 14.16. **NEW!** В хеш-таблице с открытой адресацией удаление элемента производят, помечая удаленный элемент специальным значением `DEL`, которое не равно `null`, поэтому поиск на нем не останавливается. Примерный псевдокод (без расширения и перехеширования при заполнении) приведен ниже.
- В этом коде есть проблема, из-за которой некоторая последовательность действий будет работать долго. Найдите ее.

```

void insert(k):
    x = hash(k)
    while a[x] != null:
        if a[x] == k:
            return
        x = (x + 1) mod m
    a[x] = k

void remove(k):
    x = hash(k)
    while a[x] != null:
        if a[x] == k:
            a[x] = DEL
            return
        x = (x + 1) mod m

bool contains(k):
    x = hash(k)
    while a[x] != null:
        if a[x] == k:
            return True
        x = (x + 1) mod m
    return False
    
```

- 14.17. **NEW!** Как можно легко исправить проблему из предыдущего задания?
- 14.18. **NEW!** Для того, чтобы избежать коллизий, Петя взял массив очень большого размера. Проблема только в том, что массив заполнен «мусором» — данными, оставленными от прошлого использования. Обнулить все эти данные Петя не может, потому что массив реально огромный. Помогите Пете сделать на базе этого огромного массива хеш-таблицу на n элементов, потратив $O(n)$ дополнительной памяти.
- 14.19. **NEW!** В хеш-таблице со списками будем поддерживать все списки в отсортированном порядке. Как это изменит время работы всех операций в среднем и в худшем случае?
- 14.20. **NEW!** В хеш-таблице с открытой адресацией при добавлении и поиске элемента x будем изучать элементы начиная с $h_1(x)$ с шагом $h_2(x)$ (то есть значения $(h_1(x) + i \cdot h_2(x)) \bmod M$ для $i = 0, 1, \dots$) для некоторых двух хеш-функций h_1 и h_2 . Через сколько шагов такая последовательность заикнется? Как это числ зависит от $h_1(x)$, $h_2(x)$ и M ?
- 14.21. **NEW!** Пусть процессор работает с w -битными числами и поддерживает все стандартные арифметические и битовые операции. Научитесь после подсчета за $O(w)$ и с $O(w)$ памяти, находить за $O(1)$ времени максимальную степень двойки, на которую делится заданное число.