Задание составлено на основе материалов Алексея Балицкого, Рената Гимадеева и Ильи Козлова.

Рекомендации к чтению

- Кормен, пункт 9.3 (линейный в худшем случае алгоритм поиска порядковой статистики). В этом задании уже обязательно нужно ознакомиться с этим алгоритмом.
- Рекомендация от одного авторитетного человека: книга блестящих математиков Гача и Ловаса по вычислительной сложности:

  http://www.cs.elte.hu/~lovasz/complexity.pdf.

  В ней можно прочитать то, что я рассказывал на семинаре про класс P и алгоритм Евклида, а так же про Гауссово исключение, которое есть в вашем каноническом задании. Можете почитать и другие полиномиальные темы.
- Кто не хочет читать на английском – можете прочитать про класс P в Мусатове, там все кратко и понятно. Кто любит читать много буков, то можете почитать книгу Хопкрофта, Мотвани и Ульмана начало 10 главы.
- Рекомендую (но это необязательно и в этом задании не участвует) в Кормене прочитать пункт 9.2 (линейный в среднем алгоритм поиска порядковой статистики), чтобы уже сейчас получить первое представление о вероятностных алгоритмах. О вероятности можно думать пока что примитивно, на физическом уровне строгости: вероятность это что-то типа отношения числа удачных исходов к числу всех опытов при многократном повторении опыта. О вероятностных алгоритмах мы будем говорить позднее.

Ключевые понятия: полиномиальный алгоритм, класс P, линейный алгоритм поиска порядковой статистики, алгоритм Евклида (если я не указываю источник по какой-то теме задания, то источником считаются лекции, семинары и википедия).

Обязательные задачи

Задача 1 $(1 + 1)$ Есть три типа табличек: черный квадрат 2×2, белый квадрат 2×2 и серый прямоугольник 2×1 (его можно поворачивать на $90°$).

(a) Найдите число способов замостить этими табличками полосу $2 \times n$.

(b) Предложите эффективный алгоритм нахождения этого числа.

Задача 2 (2) Предположим у нас есть некая программа (black box), которая эффективно находит медиану массива длины n за $C \cdot n$ сравнений, где C – маленькая константа (маленькая в смысле меньше констант, которые получаются в известных вам алгоритмах поиска медианы). Придумайте как можно более эффективный (**внимание**, в этой задаче константа в асимптотике имеет значение), который находит k-ую порядковую статистику массива. Докажите его корректность и оцените требуемое количество сравнений в худшем случае (оно будет зависеть от C).

Задача 3 (2) Дано n точек в трёхмерном пространстве $\mathbb{R}^3$ (точки заданы на входе своими координатами). Предложите как можно более быструю процедуру нахождения куба минимального размера, параллельного осям координат, с центром в начале координат, содержащего не менее трети всех точек.

В следующих задачах нужно привести строгое доказательство, которое использует какое-нибудь формальное определение класса P, например такое:

Определение. *$L \in P$ т. и т. т. когда существует МТ M и полином $p(n)$, такие что L распознается машиной M и на любом слове $x \in \{0,1\}^*$ (мы всегда без ограничения общности работаем в двоичном алфавите) M делает не больше $p(|x|)$ переходов.*

При этом с машинами Тьюринга можно обращаться довольно свободно. Приведу пример корректного доказательства. Пусть языки $L_1 \in P$ и $L_2 \in P$, докажем, что $L_1 \cup L_2 \in P$. Так как $L_1 \in P$ и $L_2 \in P$, существуют машины Тьюринга M_1 и M_2 и полиномы $p_1(n)$ и $p_2(n)$, т. ч. $L_1 = L(M_1)$, $L_2 = L(M_2)$ и на любом слове $x \in \{0,1\}^*$ M_1 и M_2 делают не больше $p_1(|x|)$ и $p_2(|x|)$ переходов соответственно. Рассмотрим машину Тьюринга M, которая на любом слове x будет сначала симулировать работу машины M_1. Если M_1 принимает x, то M принимает x. Если M_1 не принимает x, то M симулирует работу M_2 на x. Если M_2 принимает x, то M принимает x. В противном случае M отвергает x. На любом слове x M сделает не больше $p(|x|) = p_1(|x|)+p_2(|x|)$ переходов. Если $x \in L_1 \cup L_2$, то $x \in L_1$ или $x \in L_2$ и в обоих случаях M примет x. Если $x \notin L_1 \cup L_2$, то $x \notin L_1$ и $x \notin L_2$, поэтому M не примет x. Таким образом $L_1 \cup L_2 \in P$ по определению (для него существует распознающая его машина M, на любом слове делающая не больше $p_1(|x|) + p_2(|x|)$ переходов).

Вы так же можете ссылаться на то, что на машине Тьюринга можно за полиномиальное время выполнять со словами любые действия, которые можно выполнить полиномиально на реальном компьютере.

Задача 4 $\left(\frac{2}{3} + \frac{2}{3} + \frac{2}{3} + 3\right)$ Докажите, что если $L_1 \in P$ и $L_2 \in P$, то

(a) $L_1 \cap L_2 \in P$.

(b) $\overline{L_1} \in P$

(c) $L_1 L_2 \in P$.

(d) $L_1^* \in P$ (этот пункт сложный, его можно сдавать вместе со следующей домашкой, я его там продублирую).

Если какие-то обозначения вам неизвестны, спросите меня.

Задача 5 $\left(\frac{2}{3}+\frac{2}{3}+\frac{2}{3}+1\right)$ Докажите, что следующие языки принадлежат классу P. Можно считать, что графы кодируются соответствующими матрицами смежности.

(a) Язык двудольных графов, содержащих не менее 2016 треугольников (троек попарно смежных вершин).

(b) Язык несвязных графов без циклов (надеюсь, что после курса основных алгоритмов вы умеете решать такие задачи, а те кто не ходил и не может решить, спросите ребят, которые ходили)

(c) Язык квадратных $\{0,1\}$-матриц (т.е. элементами могут быть только нули и единицы) порядка $n \geqslant 3000$, в которых есть квадратная подматрица порядка $(n-2017)$, заполненная одними единицами.

(d) Пусть $q(t) = t^{2016} \in \mathbb{Z}[t]$ и $a, m \in \mathbb{N}$. Язык $L_{a,m} \subseteq \mathbb{N}$ определяется правилами $x_0 = a \pmod{m}$, $x_{i+1} = q(x_i) \pmod{m}$ (все члены этой рекуррентной последовательности лежат в языке, и только они).

Задача 6 (3) Дано описание программы.

ВХОД(x, y — натуральные числа) {
  Пусть 2^{d_x} — максимальная степень 2, делящая x; d_y определяется аналогично
  Положим $a = x2^{-d_x}$, $b = y2^{-d_y}$
  Поменяем местами b и a, если $b > a$
  ПОКА $b > 0$ ВЫПОЛНИТЬ {
    Положим r равным тому из чисел $a+b$, $a-b$, которое делится на 4
    ЕСЛИ $r = 0$
      Положим $a = b$, $b = 0$
    ИНАЧЕ
      Положим $a = \max(b, r2^{-d_r})$, $b = \min(b, r2^{-d_r})$,
      где 2^{d_r} — максимальная степень 2, делящая r
  }
  ВЫХОД($a2^{\min(d_x,d_y)}$)
}

(a) Что вычисляет программа?

(b) Оцените число производимых битовых операций.

Дополнительные задачи (можно сдавать в течение семестра)

Задача 7 (4) Дан массив из n элементов, на которых определено отношение равенства (например, речь может идти о массиве картинок или музыкальных записей). Постройте как можно более быстрый алгоритм, который определяет, есть ли в массиве элемент, повторяющийся не менее $\frac{n}{2}$ раз.

Задача 8 (3) Вдоль дороги (представляющей целочисленную координатную ось) стоят $N \leqslant 10^7$ деревень, они точечные и заданы координатой. Формально, даны целозначные координаты деревень $x_1, x_2, \ldots, x_N, |x_i| < 10^9$. Координаты вводятся в несортированном порядке. Также даны количества жителей $0 < m_1, m_2, \ldots, m_N < 10^4$. Приведите эффективный алгоритм (докажите его корректность и оцените сложность), который находит оптимальное местоположение для колодца, которое минимизирует взвешенную сумму расстояний от него до всех деревень отелей (веса – количества жителей).

Числа в этой задаче даны, чтобы вы могли прикинуть приемлемую сложность алгоритма. Представьте, что вам нужно запрограммировать алгоритм на языке высокого уровня, чтобы он работал в пределах пары секунд на любом входе, используя вычислительные мощности слабого персонального компьютера.