AI Masters Algorithms **Name:** *Денис Грачев*

Homework 5

21 октября 2022 г.

Task 1

Отсортируем строки с помощью RadixSort считая что буквы-цифры. Тогда из доказанного на лекции сложность сортировки будет $O(nkl)$, где n - длина массива, k - длина строк, l - размер алфавита.

Task 2

Решим задачу рекурсивно. Пусть решаем задачу для отрезка l, r. Обозначим $m = \frac{l+r}{2}$. Проверим тогда значения a_{m-1}, a_m, a_{m+1}. Рассмотрим возможные варианты

- $\nearrow\searrow$ - тогда a_m - максимальный элемент, ответ получен.
- $\nearrow\nearrow$ - тогда максимальный элемент находится на отрезке $l, m-1$, запустим рекурсивно алгоритм для него.
- $\searrow\searrow$ - тогда максимальный элемент находится на отрезке $m+1, r$, запустим рекурсивно алгоритм для него.

Таким образом мы каждый раз уполовиниваем размер входа, и на каждом шаге рекурсии тратим $O(1)$ действий. Итоговая сложность $O(\log(n))$.

Быстрее невозможно, так как возможных ответов n и из доказанного на лекции чтобы с помощью бинарных вопросов найти в этом случае ответ, необходимо хотя бы $O(\log(n))$ вопросов.

Task 3

Решим задачу рекурсивно. Разделим на 3 равные кучки и взвесим первые две. Рассмотрим возможные варианты

- Первая кучка легче. Значит фальшивая монета в ней, решим задачу для нее.
- Вторая кучка легче. Значит фальшивая монета в ней, решим задачу для нее.
- Они равны. Значим фальшивая монетка в 3 кучке, решим задачу для нее.

Таким образом после каждого взвешивания подозрительная кучка уменьшается в 3 раза, следовательно количество взвешиваний будет $\log_3(n) + c$ из-за округлений.

Task 4

Рассмотрим дерево решений. У него должно быть хотя бы n листьев, так как возможно n различных ответов. Каждая вершина имеет 3 ребенка ($<, >, =$), таким образом количество листьев на слое h это 3^h. Следовательно, минимальная необходимая высота это $\log_3(n)$.

Task 5

Обозначим массивы l_1, l_2, искомую медиану m. Решим задачу рекурсивно. Тогда медиана l_1 это $l_1[n/2]$, медиана l_2 это $l_2[n/2]$.

- $l_1[n/2] < l_2[n/2]$, тогда отрежем половину у l_1 слева, а у l_2 спрва и решим задачу рекурсивно. Действительно, $l_1[n/2] \leq m \leq l_2[n/2]$, иначе с одной стороны от медианы будет больше n значений. Так же после отрезания половин, мы убрали $n/2$ чисел меньших m и $n/2$ чисел больших n, следовательно медиана осталась прежней.
- $l_1[n/2] > l_2[n/2]$, аналогично наоборот.

- $l_1[n/2] = l_2[n/2]$ мы нашли медиану, так как тогда одинаковое количество чисел меньше и больше $l_1[n/2] = l_2[n/2]$.

Таким образом мы найдем медиану. Каждый раз длина входа делится на 2, операции стоят $O(1)$ следовательно итоговая сложность $O(\log(n))$.

Task 6

Заметим, что функция $f(x) = \sum_{i=0}^{n} a_i x^i$ строго возрастающая. Так же, так как $f(x) \geq x$. Вычислить $f(x)$ смтоит $O(n)$ операций. Будем поддерживать x^k и $\sum_{k=0}^{k} a_i x^k$, переход к $k+1$ стоит $O(1)$ (добножить x^k на x, затем на a_0 и прибавить к сумме). Проверим, что $f(1) \leq y$, иначе решений нет.
Далее, пусть $l = 1, r = y$, тогда $f(1) \leq y \leq f(y)$.
Посчитаем $f\left(\frac{l+r}{2}\right)$.

- $f\left(\frac{l+r}{2}\right) = y$, тогда решение найдено
- $f\left(\frac{l+r}{2}\right) < y$, тогда по монотонности решений на $l, \frac{l+r}{2}$ нет. Обновим $l = \frac{l+r}{2}$.
- $f\left(\frac{l+r}{2}\right) > y$, тогда по монотонности решений на $\frac{l+r}{2}, r$ нет. Обновим $r = \frac{l+r}{2}$.
- $r == l$. Решений нет.

Так как мы каждый раз уполовиниваем отрезок поиска, то максимальная глубина будет $O(\log(y))$. Каждая итерация стоит $O(n)$. Итоговая сложность $O(n \log(y))$.