Пономаренко Алексей, к/р.

№3

а) $T(n) = 25 \cdot T\left(\frac{n}{4}\right) + n^3 = 25 \cdot T\left(\frac{n}{4}\right) + \Theta(n^3)$

1) $d = 3$

2) $\log_b a = \log_4 25 \in (2; 3)$

По мастер-теореме $\Rightarrow d > \log_b a \Rightarrow \boxed{T(n) = \Theta(n^3)}$

б) $T(n) = 243 \cdot T\left(\frac{n}{3}\right) + 2n^5\left(\frac{\sqrt{n} + \log_2 n}{\sqrt{n}}\right)$

$T(n) = \sum_{k=0}^{h-1} \cancel{243^k} \cdot 2 \cdot \left(\frac{n}{\cancel{3^k}}\right)^5 \cdot \frac{\sqrt{\frac{n}{3^k}} + \log_2 \frac{n}{3^k}}{\sqrt{\frac{n}{3^k}}} = 2n^5 \cdot \sum_{k=0}^{h-1}\left(1 + \frac{\log_2 \frac{n}{3^k}}{\sqrt{\frac{n}{3^k}}}\right) \ominus$

$\ominus$ отдельно рассмотрим $\sum_{k=0}^{h-1} \frac{\log_2 \frac{n}{3^k}}{\sqrt{\frac{n}{3^k}}} = C \cdot \sum_{k=1}^{h-1} \frac{k}{3^{k/2}}$, где

$C = \frac{1}{\log_3 2}$, а также заметим, что ряд $\sum_{k=1}^{h-1} \frac{k}{3^{k/2}}$ сх-ся $\Rightarrow$

$\Rightarrow T(n) = 2n^5 \cdot \sum_{k=0}^{h-1}\left(1 + \frac{\log_2 \frac{n}{3^k}}{\sqrt{\frac{n}{3^k}}}\right) \leq C_0 \cdot n^5 = \boxed{\Theta(n^5)}$

№2

◁ Верно ли, что для ф-ий $\mathbb{N}_1 \to \mathbb{R}_{>0}$, что $f(n) = \underline{O}(g(n))$ (1) $\Leftrightarrow$

$\Leftrightarrow$ вып-но усл-е: $\exists C > 0: \lim_{n\to\infty} \frac{f(n)}{g(n)} \leq C$ (2)

▷ $\boxed{\Rightarrow}$ Докажем, что из (1) $\Rightarrow$ (2):

Распишем по опр-ю $f(n) = \underline{O}(g(n))$

$\exists N: \forall n \geq N \; \exists C_0: f(n) \leq C_0 \, g(n) \; \Big| : g(n)$ (т.к. $g(n) \in \mathbb{R}_{>0}$)

$\frac{f(n)}{g(n)} \leq C_0$ – далее делаем предельный

переход в неравенстве и получаем, что $\lim_{n\to\infty} \frac{f(n)}{g(n)} \leq \lim_{n\to\infty} C_0 = C_0 = C$

берем в качестве C конст. C_0

$\boxed{\Leftarrow}$ Теперь докажем в обр. сторону:

$\exists C > 0: \lim_{n\to\infty} \frac{f(n)}{g(n)} \leq C$

C – вещ. число $\Rightarrow$ предел ф-и $\frac{f(n)}{g(n)}$ ограничен.

$\exists A$ – предел ф-и $\frac{f(n)}{g(n)}$ $(A \leq C)$. Распишем по опр-ю предела:

$\forall \varepsilon > 0 \; \exists N(\varepsilon): \forall n \geq N \; \left|\frac{f(n)}{g(n)} - A\right| < \varepsilon \Rightarrow$ смотри продолжение на след. странице

Пономаренко Алексей, к/р

№2 (продолжение)

$\Rightarrow$ 1) $\begin{cases} \frac{f(n)}{g(n)} - A \geq 0 \\ \frac{f(n)}{g(n)} - A < \varepsilon \end{cases} \Rightarrow \frac{f(n)}{g(n)} < A + \varepsilon$

2) $\frac{f(n)}{g(n)} - A < 0 \Rightarrow \frac{f(n)}{g(n)} < A$

Таким образом, мы доказали утв-е в две стороны

$\boxed{\text{Ответ: верно}}$

№2.

Это утв-е неверно. Приведем контр-пример:

$\nexists$ $f(n) = \begin{cases} 1, & n\text{ - четное} \\ 2, & n\text{ - нечетное} \end{cases}$, $g(n) = 1$.

Тогда, $\frac{f(n)}{g(n)} \leq 2$ $\forall n \geq 1$, но предела не существует, а значит усл-е 2 не вып-но $\Rightarrow$ $\boxed{\text{утв-е неверно}}$

№1.

Рассмотрим всевозможные ситуации:

1) $q = p = 0 \Rightarrow$ заполняем массив b или 0, если $r = 0$, или r, если $r \neq 0$. Асимптотика $O(n)$ (просто заполняем массив)

2) $p = 0 \Rightarrow$ ф-я $qx + r$ линейна $\Rightarrow$ • если $q > 0$, то есть ф-я возрастает $\Rightarrow b_i = q \cdot a_i + r$, $i = \overline{0, \ldots, (n-1)}$

• если $q < 0$, то в массиве „b" наим. эл-том будет $(q \cdot a_{n-1} + r)$, в силу монот. убывания $f(x) = qx + r$. В данном случае в массив „b" помещаем $qx + r$, где на место x подставляем массив „a" в обратном порядке. Асимптотика $O(n)$, т.к. мы просто заполняем массив, без повторов

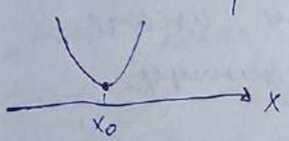
3) $p \neq 0 \Rightarrow$ ф-я $px^2 + qx + r$ – парабола. Это означает, что у неё есть область убывания и возрастания.

продолжение на след. листе

Пономаренко Алексей, к/р

№1 (продолжение)

• если $p > 0$, то ветви параболы напр-ны вверх



Найдем вершину $x_0 = -\frac{q}{2p}$. Чем ближе x к x_0, тем меньше $f(x)$ =>

=> первым делом пройдемся по массиву и найдем эл-т a_i - максимально близкий ($|a_i - x_0| = min$) к x_0. Помещаем его в массив „в" на позицию 0 как наименьший. Далее ставим два указателя слева (a_{i-1}) и справа (a_{i+1}) от нашего эл-та (если a_i - это левый или правый конец массива, то просто бежим до его другого конца). Далее сравниваем 2 числа, находящихся по указателям, и помещаем меньший из них в массив „в", сдвигая указатель от нашего наиболее близкого. Так действуем, пока оба указателя не укажут на концы массива. Этот алгоритм корректен, т.к. мы на каждом шаге выбираем наим. из эл-тов, при этом изначально идё от наименьшей точки параболы (x_0), а идем мы в стороны от вершины, т.е. по возрастанию.

Асимптотика: $\underbrace{O(n)}_{\text{поиск наиболее близкого эл-та}} + \underbrace{O(n)}_{\text{1 раз проходимся по всем эл-там „a"}} = O(n)$

4) $p < 0$. Аналогично пункту 3, только записывать в массив „в" будем с конца

Итого, мы построили корректный алгоритм сложности $O(n)$

№4.

1) Алгоритм: формируем массив „в" длины n, кот-й (изначально на все позиции массива „в" кладём 0) будем заполнять так: идем по массиву „а" в оперативн. памяти. Если $a_i \geq n$, то ничего не делаем, иначе добавляем в наш массив „в" на позицию a_i единицу. Пройдясь по всему массиву „а" 1 раз, идем по массиву „в". Если встречаем 0, то это и будет ответ, то есть, если $в_i == 0$, то i – ответ. Если мы прошлись по всему массиву, и в нём не оказалось ни одного нуля, ответ: n.

Корректность: заметим, что если между каким-либо двумя эл-тами массива „а" есть пробелы, то ответ будет или в этом пробеле, или раньше. Если все числа идут от 0 до $(n-1)$ подряд, то очевидно перв. натур. числом, отсутст. в массиве, будет число n. Если наим. эл-т массива > 0, то число 0 и будет нашим ответом. В противном случае, ответ будет номер ячейки массива „в", в кот-й лежит 0.

Сложность: $\underbrace{O(n)}_{\text{идем по массиву "а"}} + \underbrace{O(n)}_{\text{идем по массиву „в"}} = O(n)$

2) Алгоритм: смотрим первый эл-т – если он не 0, то ответ 0. Иначе бежим по массиву „а" до тех пор пока текущ. эл-т не будет отличаться от предыдущего более, чем на 1 =) ответ (текущ. эл-т – 1). Иначе ответ $(a_{n-1} + 1)$

Сложность: $O(n)$ – просто 1 раз проходимся по массиву

3) Корректность: используем метод противника.

Нам необходимо проверить с самого первого эл-та массива „а", и далее двигаться получая ответ на вопрос: $(a_i > a_{i-1} + 1 ?)$

Пономаренко Алексей, к/р

№4 (продолжение)

Нам, как противнику, выгоднее, если всегда будет ответ „нет", пока мы не пройдем все эл-ты массива. ~~Это~~ ~~так~~ То есть, менее, чем за $O(n)$ задачу не решить. Также это следует из того, что на n различ. позициях массива „а" могут стоять меньшее число эл-тов (из-за повторений) из-за чего всегда будет необх. проверять все эл-ты массива „а"

№5.

Алгоритм: я в своем решении считаю, что чем больше p_i, тем выше приоритет. Строим очередь с приоритетами. Храним суммарное время выполнения всех задач, и сколько длится текущая. При добавлении нового эл-та его приоритет меняем так: прибавляем к нему сумму всех времён выполн-я и вычитаем время вып-я текущей.

Сложность: Таким способом мы на каждом шаге ~~...~~ формируем очередь с приоритетом за $O(m)$, где m - текущ. суммарное кол-во запросов и извлекаем из дерева за $O(\log(m))$ задачу с наиб. приоритетом. То есть, суммарно получаем: $O(m \cdot \log m)$

Корректность: следует из свойств очереди с приоритетом, а также выполнения усл-й задачи

№6.

Алгоритм: строим по массиву „а" красно-черное дерево, в кот-е помещаем пару чисел $(i; a_i)$. Далее бежим по массиву и бин. поиском извлекаем из дерева нужный эл-т

Сложность: $O(n \cdot \log n)$: бежим по n эл-там массива, и за $O(\log n)$ достаем из дерева