

Лекция №7

Оценка сложности алгоритмов

Оценка сложности поиска в AVL-дереве

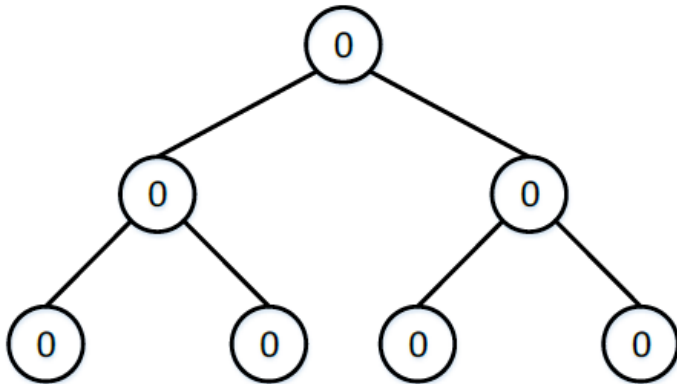
Два крайних случая AVL-деревьев это:

- (а) совершенное дерево – все узлы имеют показатель баланса 0;
- (б) дерево Фибоначчи – все узлы, кроме листовых, имеют показатель баланса +1, либо все узлы, кроме листовых, имеют показатель баланса -1.

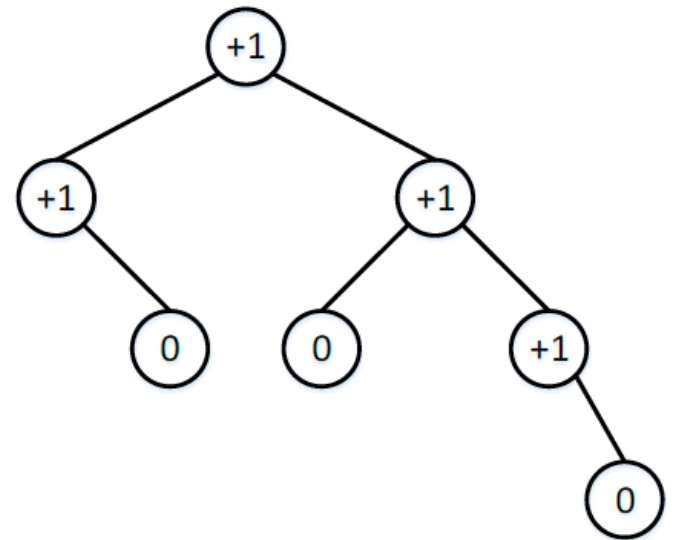
Крайние случаи AVL-деревьев:

(а) совершенное дерево;

(б) дерево Фибоначчи



(а)



(б)

Не для каждого набора ключей можно построить совершенное дерево, равно как и не для каждого набора ключей можно построить дерево Фибоначчи. Но эти деревья позволяют оценить диапазон возможных высот AVL-деревьев. Совершенное дерево является частным случаем идеально сбалансированного дерева, поэтому оно имеет минимально возможную высоту для данного количества узлов. Дерево Фибоначчи, напротив, имеет максимально возможную высоту для данного количества узлов, при условии, что сохраняются свойства AVL-дерева.

В совершенном дереве у каждого узла, кроме листовых, ровно два сына. Тогда количество узлов m равно $1 + 2 + 2^2 + \dots$

Количество таких слагаемых равно количеству уровней в дереве, т.е. на единицу больше высоты этого дерева. Если количество слагаемых равно $h + 1$, то степень двойки в последнем слагаемом будет равна h :

$$m = 1 + 2 + 2^2 + \dots + 2^h = 2^{h+1} - 1.$$

Поэтому высота совершенного дерева вычисляется как:

$$h = \log_2(m + 1) - 1$$

Покажем, что дерево Фибоначчи имеет минимально возможную высоту для заданного количества узлов среди всех АВЛ-деревьев. Если переформулировать свойство дерева Фибоначчи, то получится, что при заданной высоте оно имеет минимальное количество узлов из всех возможных АВЛ-деревьев.

Пусть W_h — множество АВЛ-деревьев заданной высоты h с минимально возможным количеством узлов, а w_h — количество узлов в каждом из этих деревьев. Тогда $w_0 = 1$ и $w_1 = 2$. Пусть T — некоторое дерево из множества W_h высоты $h \geq 2$, а T_L и T_R — его левое и правое поддеревья. Поскольку T имеет высоту h , то либо T_L , либо T_R имеет высоту $h - 1$. Не ограничивая общности рассуждений, предположим, что T_R имеет высоту $h - 1$. Поскольку T является АВЛ-деревом, и оба его поддерева T_L и T_R также являются АВЛ-деревьями, T_R является АВЛ-деревом высоты $h - 1$.

Более того, оно должно АВЛ-деревом высоты $h - 1$ с минимально возможным количеством узлов, потому как иначе оно могло бы быть заменено деревом той же высоты, но с меньшим количеством узлов, чтобы все дерево T имело минимально возможное количество узлов. Поэтому $T_R \in W_{h+1}$. Аналогично, поскольку T является АВЛ-деревом, разность высот его левого и правого поддеревьев не должна превышать 1 по модулю, а высота дерева равна h , высота дерева T_L может быть равна либо $h - 1$, либо $h - 2$.

- Но T должно иметь минимальное количество узлов, поэтому $T_L \in W_{h-2}$. Следовательно, $w_h = 1 + w_{h-2} + w_{h-1}$. Поскольку $w_0 = 1$ и $w_1 = 2$, первые несколько значений w_h будут равны 1, 2, 4, 7, 12, 20, Можно доказать по индукции, что $w_h = f_{h+3} - 1$.
- Для оценки высоты дерева Фибоначчи потребуются привести некоторые выкладки.
- Из определения дерева Фибоначчи следует, что для любого узла, кроме листовых, разность высот левого и правого поддеревьев равна 1 либо -1 . Не ограничивая общности рассуждений, будем считать, что эта разность равна 1. Таким образом, если высота дерева равна h , то высоты левого и правого поддеревьев равны $h - 2$ и $h - 1$ соответственно. Это свойство выполняется для любого поддерева дерева Фибоначчи.

Определение 3: Числа Фибоначчи – это элементы числовой последовательности, где первые два числа равны 1, а каждое последующее число равно сумме двух предыдущих:

$$f_1 = 1, f_2 = 1, f_n = f_{n-1} + f_{n-2}, n \geq 3, n \in \mathbb{Z}. \quad (2)$$

Теорема 1. Число узлов в дереве Фибоначчи F_h высоты h равно $m(h) = f_{h+2} - 1$.

Доказательство (по индукции)

Базис: $m(0) = f_2 - 1 = 0$, $m(1) = f_3 - 1 = 1$.

Шаг: по определению $m(h) = m(h-1) + m(h-2) + 1$.

Имеем $m(h) = (f_{h+1} - 1) + (f_h - 1) + 1 = f_{h+2} - 1$, т.к. $f_{h+1} + f_h = f_{h+2}$ (следует из Определения 1). Теорема доказана.

Теорема 2. Пусть C_1 и C_2 таковы, что уравнение

$$r^2 - C_1 r - C_2 = 0 \quad (3)$$

имеет два различных корня r_1 и r_2 , $r_1 \neq r_2$. Пусть последовательность a_n такова, что

$$a_n = \alpha_1 r_1^n + \alpha_2 r_2^n. \quad (4)$$

Тогда выполняется соотношение

$$a_n = C_1 a_{n-1} + C_2 a_{n-2}. \quad (5)$$

Доказательство. r_1 и r_2 – корни уравнения (3), тогда $r_1^2 = C_1 r_1 + C_2$
и $r_2^2 = C_1 r_2 + C_2$.

Имеем:

$$\begin{aligned} C_1 a_{n-1} + C_2 a_{n-2} &= C_1(\alpha_1 r_1^{n-1} + \alpha_2 r_2^{n-1}) + C_2(\alpha_1 r_1^{n-2} + \alpha_2 r_2^{n-2}) = \\ &= \alpha_1 r_1^{n-2}(C_1 r_1 + C_2) + \alpha_2 r_2^{n-2}(C_1 r_2 + C_2) = \\ &= \alpha_1 r_1^{n-2} r_1^2 + \alpha_2 r_2^{n-2} r_2^2 = \alpha_1 r_1^n + \alpha_2 r_2^n = a_n. \end{aligned} \quad (6)$$

Теорема доказана.

Теорема 3. Пусть C_1 и C_2 таковы, что уравнение (3) имеет два различных действительных корня r_1 и r_2 , $r_1 \neq r_2$.

Пусть последовательность a_n задана значениями своих первых членов a_0 и a_1 и рекуррентным соотношением (5). Тогда для некоторых α_1 и α_2 и любого $n = 1, 2, \dots$ выполняется соотношение (4).

Доказательство. Подберем такие α_1 и α_2 , чтобы выполнялось соотношение (4):

$$a_0 = \alpha_1 + \alpha_2, \text{ и} \quad (7)$$

$$a_1 = \alpha_1 r_1 + \alpha_2 r_2. \quad (8)$$

Для этого рассмотрим совокупность (7) и (8) как систему линейных уравнений, которую надо решить относительно α_1 и α_2 . Получим:

$$\alpha_1 = \frac{a_1 - a_0 r_2}{r_1 - r_2},$$

$$\alpha_2 = \frac{-a_1 + a_0 r_1}{r_1 - r_2}.$$

Соотношение (4) выполняется для a_0 и a_1 . Действительно, $a_0 = \alpha_1 r_1^0 + \alpha_2 r_2^0$ и $a_1 = \alpha_1 r_1^1 + \alpha_2 r_2^1$. Докажем по индукции, что если соотношение выполняется для a_{n-2} и a_{n-1} , то оно выполняется для a_n . Для этого повторим в обратном порядке вывод (5) из (4),₃

который проводился при доказательстве Теоремы 2, т.е. выведем (4) из (5). Для этого необходимо выписать цепочку (6) в обратном порядке:

$$\begin{aligned}\alpha_1 r_1^n + \alpha_2 r_2^n &= \alpha_1 r_1^{n-2} r_1^2 + \alpha_2 r_2^{n-2} r_2^2 = \\ &= \alpha_1 r_1^{n-2} (C_1 r_1 + C_2) + \alpha_2 r_2^{n-2} (C_1 r_2 + C_2) = \\ C_1 (\alpha_1 r_1^{n-1} + \alpha_2 r_2^{n-1}) + C_2 (\alpha_1 r_1^{n-2} + \alpha_2 r_2^{n-2}) &= C_1 a_{n-1} + C_2 a_{n-2} = a_n.\end{aligned}$$

Теорема доказана.

Применим доказанные теоремы к числам Фибоначчи. Соотношение (2) эквивалентно соотношению (5) при $C_1 = C_2 = 1$. Уравнение (3) при этом записывается как

$$r^2 - r - 1 = 0$$

и имеет корни $r_1 = \frac{1 + \sqrt{5}}{2}$ и $r_2 = \frac{1 - \sqrt{5}}{2}$. Следовательно, согласно

Теореме 3:

$$f_0 = \alpha_1 + \alpha_2 = 0,$$

$$f_1 = \alpha_1 \left(\frac{1+\sqrt{5}}{2} \right) + \alpha_2 \left(\frac{1-\sqrt{5}}{2} \right) = 1,$$

$$\alpha_1 = \frac{1}{\sqrt{5}}, \alpha_2 = -\frac{1}{\sqrt{5}},$$

$$f_n = \alpha_1 r_1^n + \alpha_2 r_2^n = \frac{1}{\sqrt{5}} \left(\frac{1+\sqrt{5}}{2} \right)^n - \frac{1}{\sqrt{5}} \left(\frac{1-\sqrt{5}}{2} \right)^n.$$

Согласно Теореме 1

$$m(h) = f_{h+2} - 1 = \frac{1}{\sqrt{5}} \left(\frac{1+\sqrt{5}}{2} \right)^{h+2} - \frac{1}{\sqrt{5}} \left(\frac{1-\sqrt{5}}{2} \right)^{h+2} - 1$$

$$\left| \frac{1}{\sqrt{5}} \left(\frac{1-\sqrt{5}}{2} \right)^{h+2} \right| < 1$$

$$m(h) + 1 > \frac{1}{\sqrt{5}} \left(\frac{1+\sqrt{5}}{2} \right)^{h+2}$$

Введем обозначение $\gamma = \frac{1+\sqrt{5}}{2}$. Тогда

$$m(h) + 1 > \frac{1}{\sqrt{5}} \gamma^{h+2} \quad (9)$$

Логарифмируя обе части (9), получаем

$$\log_2(m(h) + 1) > \log_2\left(\frac{1}{\sqrt{5}}\right) + (h + 2)\log_2 \gamma,$$

$$\log_2(m(h) + 1) > -\log_2(\sqrt{5}) + (h + 2)\log_2 \gamma,$$

$$h + 2 < \frac{\log_2(m + 1)}{\log_2 \gamma} + \frac{\log_2 \sqrt{5}}{\log_2 \gamma},$$

откуда

$$h < 1.44\log_2(m + 1) - 0.32. \quad (10)$$

Оценка сложности AVL-дерева

Таким образом, учитывая оценку высоты совершенного дерева (1) и оценку высоты дерева Фибоначчи (10) получаем, что высота h AVL-дерева из m узлов находится в диапазоне

$$\log_2(m+1) \leq h < 1.44 \log_2(m+1) - 0.32. \quad (11)$$

Это соотношение (11) задает оценку количества сравнений при поиске узла в AVL-дереве на пути от корня к этому узлу. Если, например, в AVL-дереве 10^6 вершин, то его высота, а, следовательно, и сложность поиска узла в нем, не превысит 29.

Оценка сложности поиска в КЧ-дереве

- **Лемма 1.** Красно-черное дерево с n внутренними узлами (без фик-тивных листьев) имеет высоту не более $2\log_2(n+1)$.

Доказательство.

1. Обозначим через $bh(x)$ черную высоту поддеревя с корнем в узле x . Покажем вначале, что оно содержит не менее $2^{bh(x)} - 1$ внутренних узлов.

- а. Индукция. Для листьев (не фиктивных) $bh(x) = 0$, то есть, $2^{bh(x)} - 1 = 2^0 - 1 = 0$.
- б. Пусть теперь x — не лист и имеет черную высоту $bh(x) = k$. Тогда каждый сын x имеет черную высоту не менее $k - 1$. Действительно, если узел красный, его сыновья могут быть только черными, и в этом случае черная высота сына x будет на 1 меньше, чем черная высота x , т.е. будет равна $k - 1$, потому как при расчете черной высоты узла сам узел в это число не включается. Если узел черный, то его сыновья

могут быть как черными, так и красными. Если сын черный, то аналогично предыдущему случаю его черная высота на 1 меньше, чем у x и равна $k - 1$. Если сын красный, то он имеет черную высоту такую же, как и x , т.е. k , т.к. количество черных узлов от x до листа, не включая x , такое же, как и количество черных узлов от его красного сына до листа.

- с. По предположению индукции каждый сын имеет черную высоту $bh \geq k - 1$, а, следовательно, не менее $2^{k-1} - 1$ узлов. Поэтому поддерево с корнем x имеет не менее $2^{k-1} - 1 + 2^{k-1} - 1 + 1 = 2^k - 1$ узлов.

2. Теперь пусть высота дерева равна h .

а. По свойству красно-черных деревьев, что если узел красный, то оба его сына черные, черные узлы составляют не меньше половины всех узлов на пути от корня к листу. Поэтому, черная высота дерева bh не меньше $h/2$.

б. Тогда $n \geq 2^{h/2} - 1$, откуда

$$h \leq 2\log_2(n + 1). \quad (12)$$

Лемма доказана.

Из Леммы 1 следует, что поиск по КЧ-дереву имеет сложность $O(\log_2 n)$.

Оценка сложности операций над самоперестраивающимся деревом

Во-первых, заметим, что все вышеперечисленные базовые операции (поиск, вставка, удаление) требуют выполнения $O(1)$ операций `splay` и $O(1)$ дополнительного времени. Действительно, при поиске и вставке требуется одна операция `splay`, а при удалении – две. Дополнительные действия – просмотр корня, удаление вершины и т.п. занимают фиксированное время, которое не зависит от высоты дерева. Сама операция `splay` занимает $O(n)$ времени, где n – количество узлов в дереве. Это происходит тогда, когда дерево совсем не сбалансировано. Но, в целом, эти деревья за счет того, что они самоперестраивающиеся, имеют тенденцию к сбалансированности.

Введем следующие понятия.

Определение 7: *Весом узла x называется количество узлов в поддереве T с корнем в x , включая сам узел: $|T(x)|$.*

Определение 8: Рангом узла называется двоичный логарифм его веса: $r(x) = \log_2 |T(x)|$.

Проведем усредненную оценку сложности операций с помощью так называемого *метода бухгалтерского учета*. Представим, что каждая операция с деревом стоит фиксированную сумму денег за единицу времени. В самом начале каждый узел содержит кредит — $r(x)$ рублей, т.е. сумму, равную его рангу, которая может частично или полностью использоваться для оплаты операций. Также можно инвестировать дополнительную сумму для оплаты операций, помимо кредита. Если после серии операций над деревом и перед началом новых операций суммарный кредит (сумма рангов всех вершин) будет не меньше, чем до них, то будем говорить о *сохранении денежного инварианта*. Тогда можно проводить эту серию операций любое количество раз. Докажем следующую лемму.

Доказательство:

Сложность будет оцениваться по количеству поворотов. Обозначим через $r(x)$ и $r'(x)$ значения ранга узла x до и после поворота (1-го типа – одинарного, 2-го типа – серии из двух одинарных или 3-го типа – двойного).

Рассчитаем, какую сумму потребуется инвестировать, чтобы ее хватило и на сохранение денежного инварианта, и непосредственно на операции.

Ниже будет показано, что для выполнения поворота любого типа и сохранения денежного инварианта требуется не более $3(r'(x) - r(x)) + 1$ рублей, причем 1 добавляется только при одинарном повороте. При выполнении операции $\text{splay}(T, x)$ производится последовательность из k поворотов 2-го и 3-го типов и поворота 1-го типа на заключительном этапе. После каждого поворота ранг узла x будет меняться. Пусть изначально ранг узла x состав-

ляет $r_0(x)$, а после i -го поворота $r_i(x)$. Тогда для выполнения последовательности из k поворотов потребуется сумма

$$1 + \sum_{i=1}^k 3(r_i(x) - r_{i-1}(x)) = 1 + 3(r_k(x) - r_0(x)) \quad (13)$$

Учитывая, что $r_k(x) = r(T)$, поскольку в конце последовательности поворотов узел оказывается в корне всего дерева, а $r_0(x) = r(x)$, получаем общее количество рублей, необходимое для выполнения операции $\text{splay}(x, T)$ и сохранения денежного инварианта, равное $3(r(T) - r(x)) + 1$ рублей. Учитывая, что минимальное значение $r(x)$ равно 0, а ранг дерева является логарифмом от количества его узлов, получаем верхнюю оценку: $3\log_2 n + 1$ рублей.

Теперь покажем, что для выполнения поворота и сохранения денежного инварианта требуется не более $3(r'(x) - r(x)) + 1$ рублей. Рассмотрим каждый тип поворота по отдельности.

1. Одинарный поворот

Обозначим корень за y , а его сына – искомый узел – за x . Для сохранения денежного инварианта потребуется сумма

$$\delta = r'(x) + r'(y) - r(x) - r(y).$$

После поворота x и y меняются ролями – x становится отцом, а y – сыном. Поэтому $r'(x) = r(y)$ и $r'(y) \leq r'(x)$. В остальных узлах значения r не меняются. Следовательно,

$$\delta = r'(y) - r(x) \leq r'(x) - r(x).$$

Для выполнения, собственно, вращения потребуется 1 операция, и, следовательно, 1 рубль. Поэтому общее количество рублей N , необходимое для выполнения поворота и сохранения денежного инварианта, удовлетворяет неравенству:

$$N \leq r'(x) - r(x) + 1 \leq 3(r'(x) - r(x)) + 1. \tag{14}$$

2. Два одинарных поворота

Обозначим искомый узел за x , его отца за y , а деда за z . Для сохранения денежного инварианта потребуется сумма

$$\delta = r'(x) + r'(y) + r'(z) - r(x) - r(y) - r(z).$$

После выполнения двух одинарных поворотов узел x становится «старшим» в этой комбинации, его отец y становится его сыном, а его дед z — его внуком. Поэтому $r'(x) = r(z)$, $r'(y) \leq r'(x)$ и $r(y) \geq r(x)$. В остальных узлах значения r не меняются. Следовательно, получим:

$$\delta = r'(y) + r'(z) - r(x) - r(y) \leq r'(x) + r'(z) - 2r(x).$$

Непосредственно два поворота составляют 2 операции, следовательно, требуют 2 рубля. Поэтому, для выполнения вращения и сохранения денежного инварианта потребуется сумма

$$N \leq r'(x) + r'(z) - 2r(x) + 2.$$

Докажем, что

$$r'(x) + r'(z) - 2r(x) + 2 \leq 3(r'(x) - r(x)). \quad (15)$$

Перепишем неравенство (15) в виде

$$r'(z) + r(x) - 2r'(x) \leq -2.$$

По определению ранга

$$r'(z) + r(x) - 2r'(x) = \log_2 |T'(z)| + \log_2 |T(x)| - 2\log_2 |T'(x)|, \quad (16)$$

где $T(x)$ ($T(z)$) – дерево с корнем в узле x (z) до поворота, а $T'(x)$ ($T'(z)$) – дерево с корнем в узле x (z) после поворота.

Преобразуем правую часть (16):

$$\begin{aligned}
& \log_2 |T'(z)| + \log_2 |T(x)| - 2\log_2 |T'(x)| = \\
& = (\log_2 |T'(z)| - \log_2 |T'(x)|) + (\log_2 |T(x)| - \log_2 |T'(x)|) = \\
& = \log_2 \frac{|T'(z)|}{|T'(x)|} + \log_2 \frac{|T(x)|}{|T'(x)|}.
\end{aligned}$$

Зная, как будет перестраиваться дерево при повороте, можно заметить, что $|T'(x)| \geq |T'(z)| + |T(x)|$.

Следовательно,

$$\frac{|T'(z)|}{|T'(x)|} + \frac{|T(x)|}{|T'(x)|} = \frac{|T'(z)| + |T(x)|}{|T'(x)|} \leq 1.$$

Выпуклость функции логарифма предполагает, что

$$\log_2 x + \log_2 y = \log_2 xy \text{ для } x, y > 0, x + y \leq 1$$

принимает максимальное значение -2 при

$$x = y = \frac{1}{2}.$$

Поэтому

$$\log_2 \frac{|T'(z)|}{|T'(x)|} + \log_2 \frac{|T(x)|}{|T'(x)|} \leq -2,$$

следовательно,

$$\log_2 |T'(z)| + \log_2 |T(x)| - 2\log_2 |T'(x)| \leq -2.$$

Таким образом, мы получили, что для выполнения двух одинарных поворотов и сохранения денежного инварианта требуется сумма

$$N \leq 3(r'(x) - r(x)). \quad (17)$$

Двойной поворот

После выполнения двойного поворота узел x становится «старшим», а его отец y и дед z становятся его сыновьями. Так же, как и в предыдущем случае, имеют место соотношения $r'(x) = r(z)$ и $r(y) \geq r(x)$. Следовательно, для сохранения денежного инварианта потребуются сумма

$$\delta = r'(x) + r'(y) + r'(z) - r(x) - r(y) - r(z) \leq r'(y) + r'(z) - 2r(x).$$

Для выполнения двойного поворота и сохранения денежного инварианта потребуется сумма $N = \delta + 2$, поэтому

$$N \leq r'(y) + r'(z) - 2r(x) + 2.$$

Докажем, что

$$r'(y) + r'(z) - 2r(x) + 2 \leq 2(r'(x) - r(x)). \quad (18)$$

Перепишем неравенство (18) в виде

$$r'(y) + r'(z) - 2r'(x) \leq -2.$$

Это неравенство доказывается аналогично предыдущему случаю, только, учитывая, что y и z стали сыновьями x , используем неравенство

$$|T'(x)| \geq |T'(y)| + |T'(z)|.$$

Получаем, что для выполнения двойного поворота и сохранения денежного инварианта требуется сумма

$$N \leq 2(r'(x) - r(x)) \leq 3(r'(x) - r(x)). \quad (19)$$

Учитывая неравенства (14), (17) и (19), получаем, что для выполнения любого из трех типов поворотов и сохранения денежного инварианта требуется не более

$$3(r'(x) - r(x)) + 1 \text{ рублей.}$$

Лемма доказана.

Теорема 4: Любая последовательность из m словарных операций на самоперестраивающемся дереве, которое было изначально пусто и на каждом шаге содержало не более n узлов, занимает не более $O(m \log n)$ времени.

Доказательство:

Из Леммы 2 следует, что операция $\text{splay}(T, x)$ требует инвестирования не более $3 \log_2 n + 1$ рублей. Вспоминая, что каждая из словарных операций требует $O(1)$ операций splay и $O(1)$ дополнительного времени, получаем, что для выполнения последовательности из m операций дополнительно требуется не более $O(m(3 \log_2 n + 1))$ инвестиций (при удалении операция splay производится два раза) и при этом можно использовать деньги узла. Сначала дерево содержит 0 рублей, в конце ≥ 0 рублей. Следовательно, $O(m \log n)$ рублей хватает на все операции. Теорема доказана.