Лабораторная №4

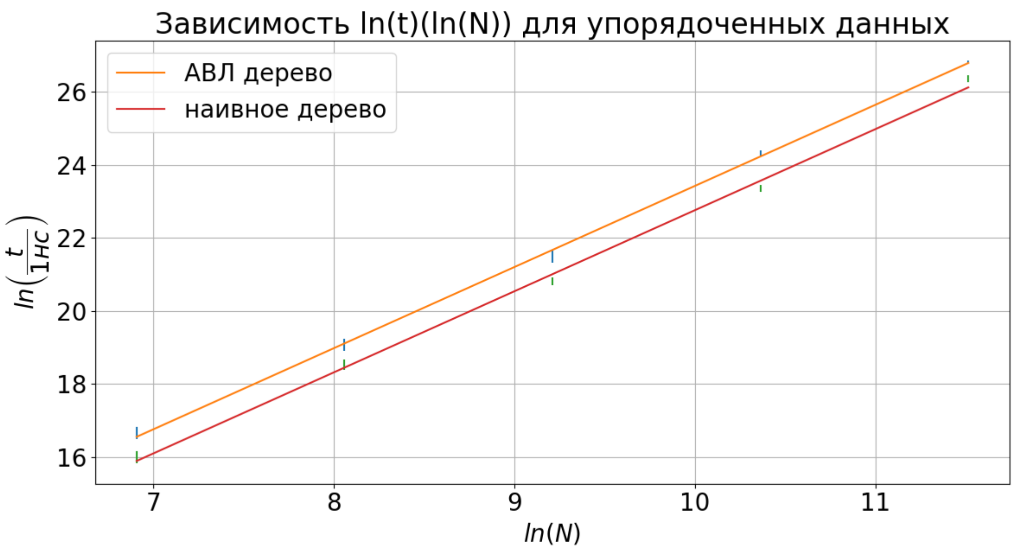
В данной лабораторной работе реализованы бинарные деревья поиска: наивное дерево и АВЛ дерево. Исследуем сложности поиска элементов и заполнения деревьев отсортированными и случайными данными.

1 Наполнение

1.1 Наполнение отсортированными данными

Сначала проведем измерения для наполнения деревьев строго отсортированными данными. Так как при построении АВЛ дерева происходит балансировка, предполагается, что наполняться оно будет медленнее. Проведем измерения:

Построим графики в логарифмических осях.



Как видно, графики хорошо аппроксимируется прямыми $ln\left(\frac{t}{1\text{нс}}\right) = kln(N) + b$

Тогда $t = t_0 e^b N^k$, где $t_0 = 1$(нс)

Для АВЛ дерева из графика получаем

$$k = (2.22 \pm 0.04)$$

$$b = (1.2 \pm 0.4)$$

Тогда можем оценить характерное время добавления одного элемента τ, приняв $N = 1$:

$$\tau = t_0 e^b = (3.3 \pm 1.3)\text{нс}$$

Для наивного дерева получаем

$$k = (2.22 \pm 0.06)$$

$$b = (0.5 \pm 0.5)$$

Тогда можем оценить характерное время добавления одного элемента τ, приняв $N = 1$:

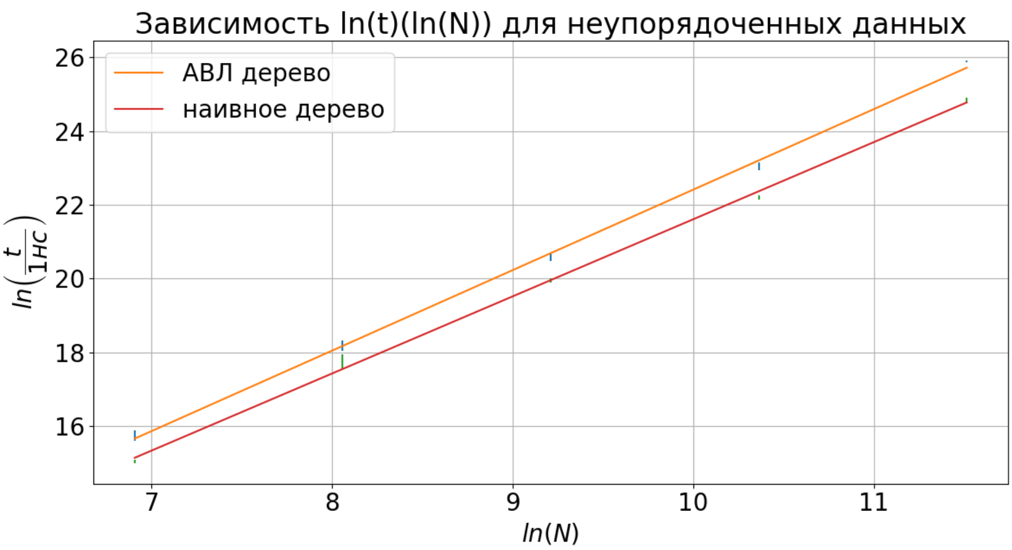
$$\tau = t_0 e^b = (1.6 \pm 0.8)\text{нс}$$

Заметим, что в пределах погрешности k совпадают для обеих реализаций, то есть вид зависимости $t(N)$ одинаков, но характерные времена добавления одного элемента различаются примерно в 2 раза.

1.2 Наполнение случайными данными

Теперь проведем измерения для случайных данных:

Построим графики в логарифмических осях.



Как видно, графики хорошо аппроксимируется прямыми $ln\left(\frac{t}{1\text{нс}}\right) = kln(N) + b$

Тогда $t = t_0 e^b N^k$, где $t_0 = 1(\text{нс})$

Для АВЛ дерева из графика получаем

$$k = (2.09 \pm 0.05)$$

$$b = (0.7 \pm 0.4)$$

Тогда можем оценить характерное время добавления одного элементаτ, приняв $N = 1$:

$$\tau = t_0 e^b = (2.0 \pm 0.8)\text{нс}$$

Для наивного дерева получаем

$$k = (2.18 \pm 0.04)$$

$$b = (0.6 \pm 0.4)$$

Тогда можем оценить характерное время добавления одного элементаτ, приняв $N = 1$:

$$\tau = t_0 e^b = (1.8 \pm 0.7)\text{нс}$$

Заметим, что в пределах погрешности k совпадают для обеих реализаций, то есть вид зависимости $t(N)$ одинаков, характерные времена добавления тоже равны в пределах погрешности. Это может быть связано с тем, что для отсортированных данных необходимо было большее количество раз перестраивать АВЛ дерево.

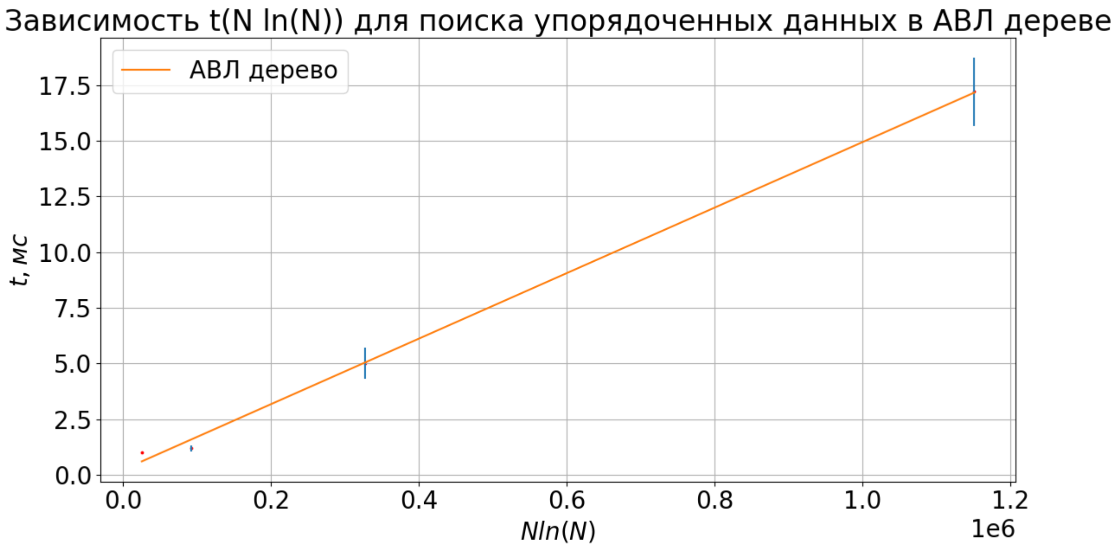
Также стоит отметить, что заполнение происходит со сложностью примерно $O(N^2)$. Значит, добавление одного элемента имеет сложность $O(N)$, так как при заполнении было добавлено N элементов, а не один.

2 Поиск элементов, которые содержатся в дереве

2.1 Поиск при заполнении отсортированными данными

Проведем измерения для наполнения деревьев отсортированными данными. Так как наивное дерево эквивалентно списку, то ожидается, что время поиска в АВЛ дереве будет меньше. Теоретически сложность поиска одного элемента в АВЛ дереве $O(ln(N))$, а в наивном $O(N)$. Так как поиск был проведен N раз, то предполагается зависимость $t \sim Nln(N)$ для АВЛ дерева и $t \sim N^2$ для наивного дерева.

Построим график $t(Nln(N))$ для АВЛ дерева.



Как видно, график хорошо аппроксимируется прямой $t = t_0(kNln(N)+b)$, где $t_0 = 1$(нс)

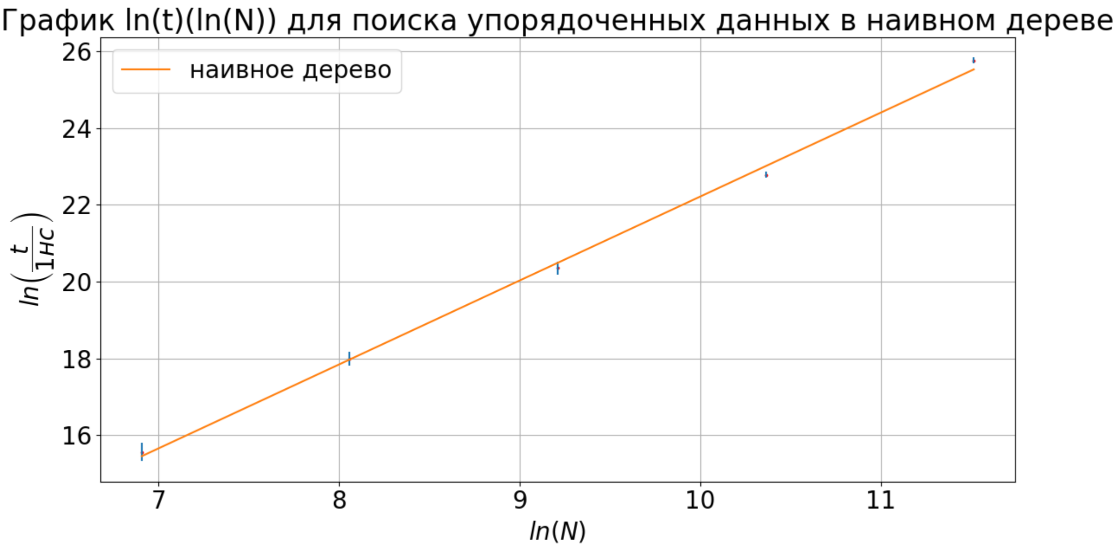
Из графика находим

$$k = (14.7 \pm 0.4)$$

$$b = (2.3 \pm 2.3) \cdot 10^5$$

Так как поправка $t_0 b$ порядка 10^{-4} секунды, а характерное время t порядка 10^{-2} секунды, то можем считать, что теоретическая зависимость выполняется. Тогда сложность поиска одного элемента в АВЛ дереве действительно $O(ln(N))$

Построим график зависимости t от N для наивного дерева в логарифмических осях.



Как видно, график хорошо аппроксимируется прямой $ln\left(\frac{t}{1\text{нс}}\right) = kln(N)+b$

Из графика находим:

$$k = (2.19 \pm 0.06)$$

$$b = (0.3 \pm 0.5)$$

Значит, $t \sim N^2$, то есть сложность поиска одного элемента в наивном дереве $O(N)$.

Итак, теоретические зависимости подтверждаются.

2.2 Поиск при заполнении случайными данными

Теперь проведем измерения для случайных данных. Для АВЛ дерева по прежнему ожидается зависимость вида $t \sim Nln(N)$. Для наивного дерева она может оказаться такой же, так как данные случайны и структура наивного дерева "близка"к структуре АВЛ дерева.

Построим график $t(Nln(N))$ для АВЛ дерева.

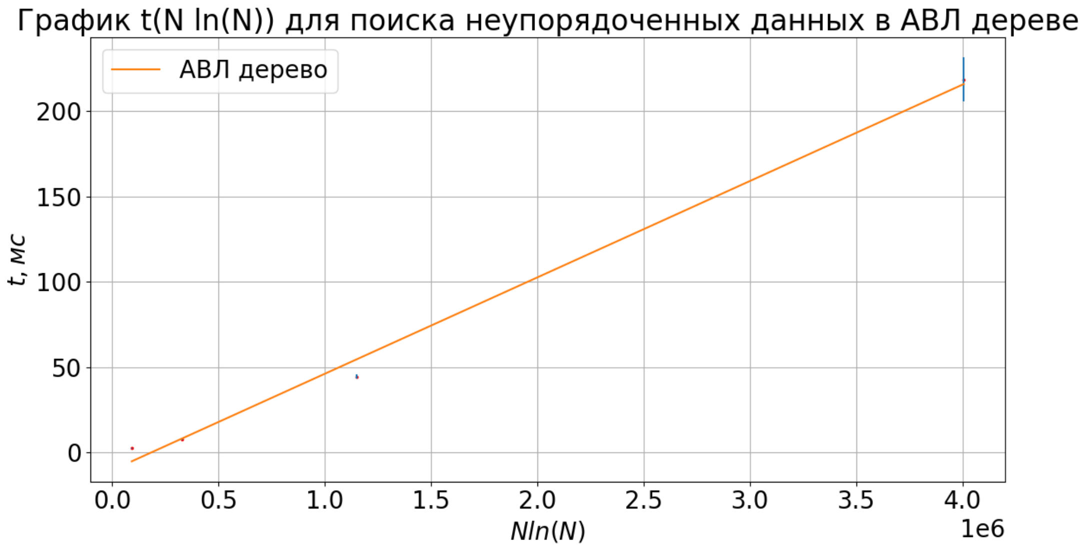


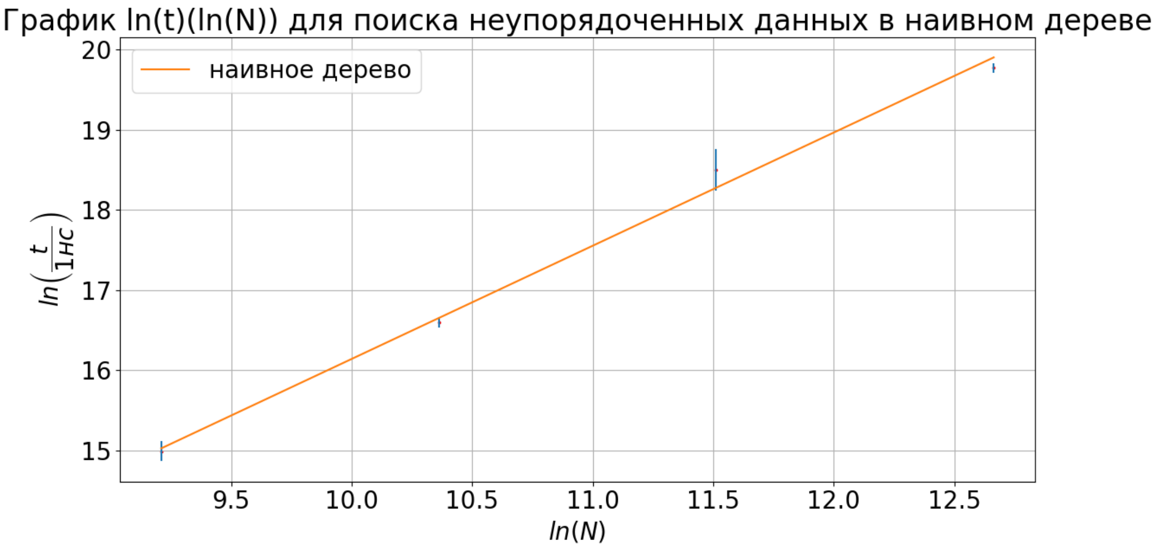
График действительно линейный, $t = t_0(kNln(N) + b)$, где $t_0 = 1$(нс)
Из графика находим

$$k = (56 \pm 3)$$

$$b = (-10 \pm 6) \cdot 10^6$$

Так как поправка $t_0 b$ порядка 10^{-2} секунды, а характерное время t порядка 10^1 секунды, то можем считать, что теоретическая зависимость выполняется. Тогда сложность поиска одного элемента в АВЛ дереве действительно $O(ln(N))$

Теперь исследуем наивное дерево. Построим график t от N в логарифмических координатах.



Как видно, график аппроксимируется прямой $ln\left(\frac{t}{1\text{нс}}\right) = kln(N) + b$
Из графика находим:

$$k = (1.41 \pm 0.07)$$

$$b = (2.1 \pm 0.8)$$

В пределах погрешности верно, что для наивного дерева сложность $O(N^{0.4})$

Построим график $t(Nln(N))$ для наивного дерева.

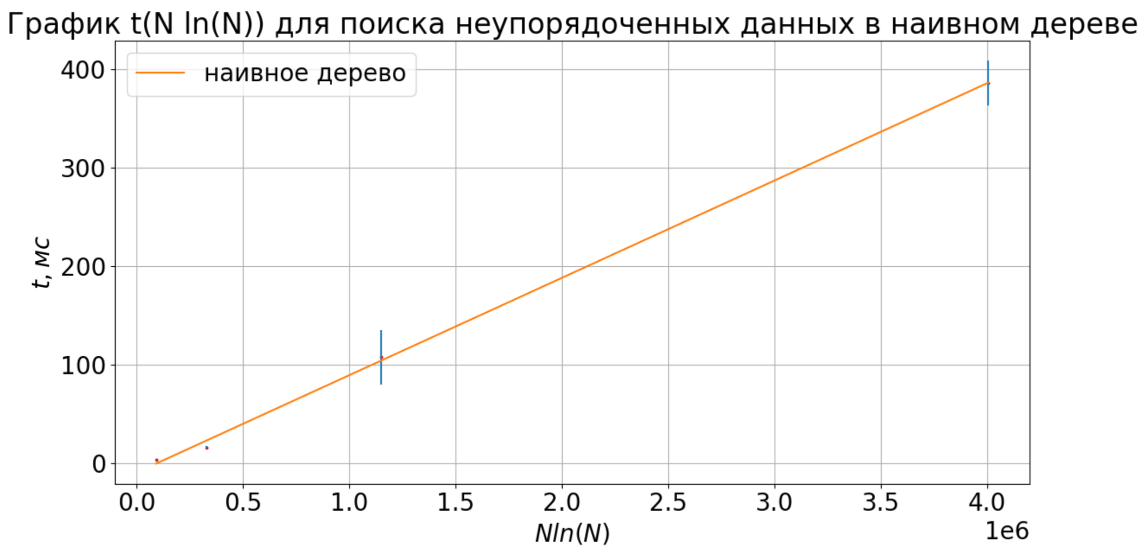


График аппроксимируется прямой вида $t = t_0(kNln(N) + b)$, где $t_0 =$ 1(нс)

Из графика находим

$$k = (98.9 \pm 1.9)$$

$$b = (-10 \pm 6) \cdot 10^6$$

Аналогично поправка $t_0 b$ мала, поэтому можем считать, что в пределах погрешности $t \sim Nln(N)$.

Так как в данном случае рассматривается одна и та же модель для обоих деревьев, то можем сравнить их характерные времена поиска одного элемента $\approx kt_0$. Для АВЛ дерева это время примерно в 2 раза меньше.

Итак, теоретическая зависимость для АВЛ дерева подтверждается, а для наивного дерева в пределах погрешности применимы две модели: сложность поиска одного элемента либо $O(N^{0.4})$, либо $O(ln(N))$. Во втором случае относительная погрешность коэффициентов меньше, поэтому эта модель точнее.

3 Поиск элементов, которых нет в дереве

Для того, чтобы поиск не происходил все время по одному пути, в деревья добавлялись четные элементы, а искались нечетные, отличающиеся от тех, которые есть в дереве, на единицу.

3.1 Поиск при заполнении отсортированными данными

Поиск элемента, которого нет, эквивалентен поиску элемента, у которого нет потомков, то есть который находится "внизу"в дереве. Для АВЛ дерева поиск такого элемента имеет сложность $O(ln(N))$, а для наивного дерева, заполненного отсортированными данными $O(N)$, так как дерево эквивалентно списку и при поиске перебираются все элементы, не превосходящие данный.

Построим график $t(Nln(N))$ для АВЛ дерева.

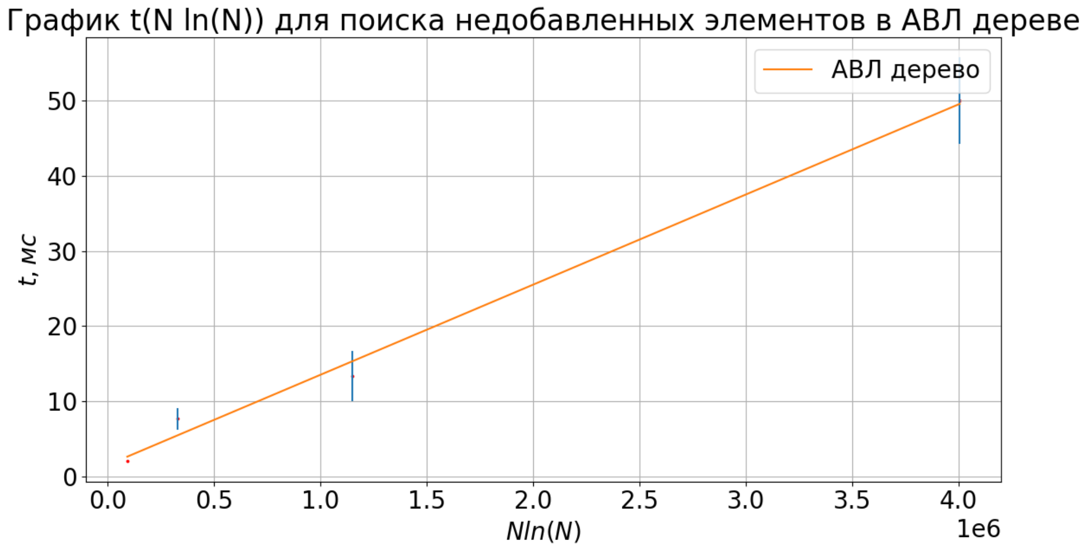


График действительно линейный, $t = t_0(kNln(N) + b)$, где $t_0 = 1$(нс)
Из графика находим

$$k = (12.0 \pm 0.7)$$

$$b = (1.5 \pm 1.5) \cdot 10^6$$

Итак, сложность поиска действительно $O(ln(N))$.

Теперь построим график t от N в логарифмических осях для наивного дерева.

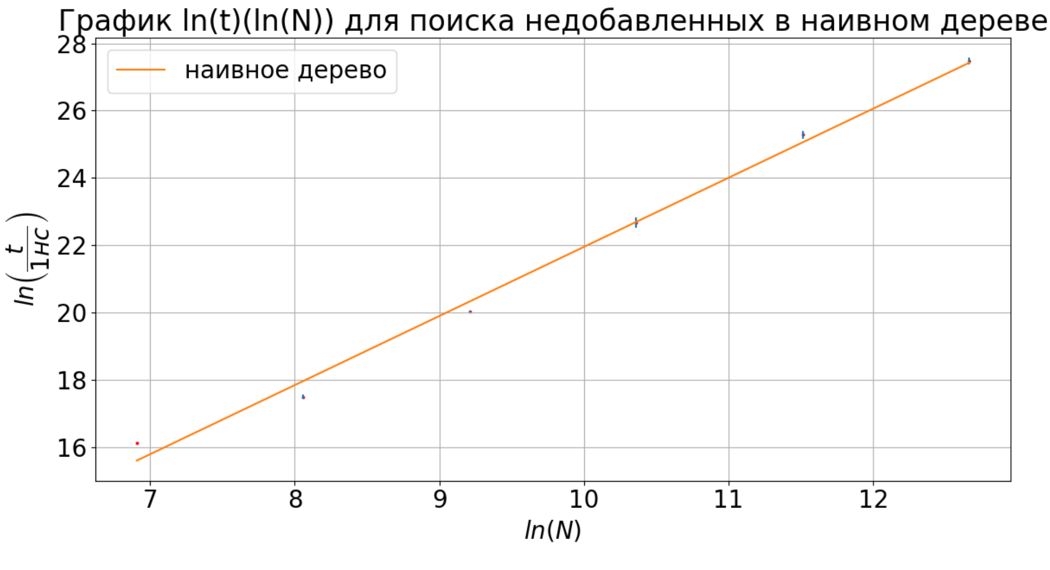


График действительно линейный, $ln\left(\frac{t}{1\text{нс}}\right) = kln(N) + b$
Из графика находим:

$$k = (2.05 \pm 0.08)$$

$$b = (1.4 \pm 0.8)$$

Тогда поиск недобавленного элемента действительно имеет сложность $O(N)$.

3.2 Поиск при заполнении случайными данными

Для АВЛ дерева все так же, как для отсортированных данных. Для наивного дерева зависимость t от N может оказаться такой же, как для АВЛ дерева, так как данные случайны и структура наивного дерева "близка"к структуре АВЛ дерева.

Построим график $t(Nln(N))$ для АВЛ дерева.

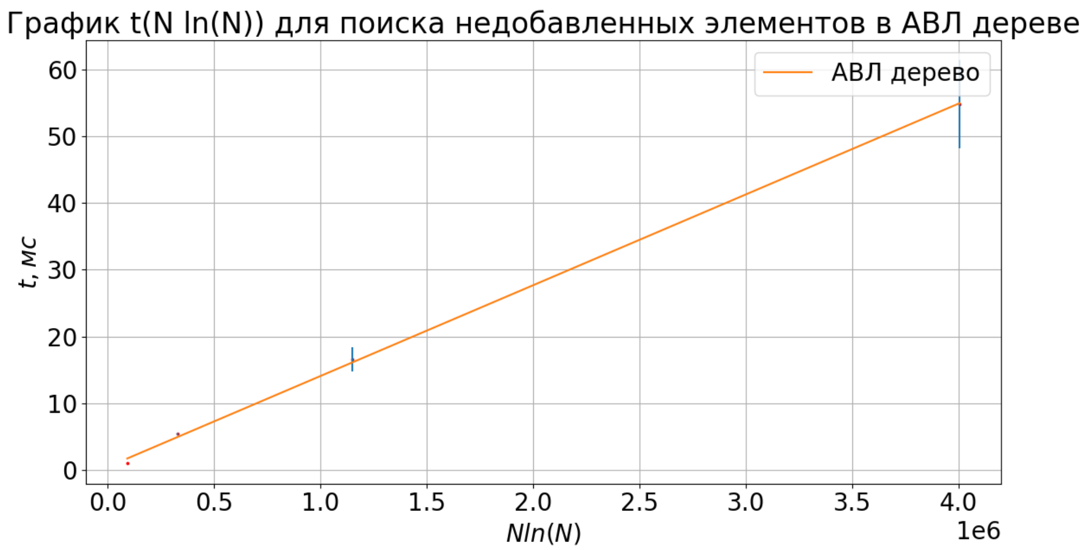


График действительно линейный, $t = t_0(kNln(N) + b)$, где $t_0 = 1(\text{нс})$
Из графика находим

$$k = (13.6 \pm 0.2)$$

$$b = (5 \pm 5) \cdot 10^5$$

Итак, сложность поиска действительно $O(ln(N))$.

Теперь построим график t от N в логарифмических осях для наивного дерева.

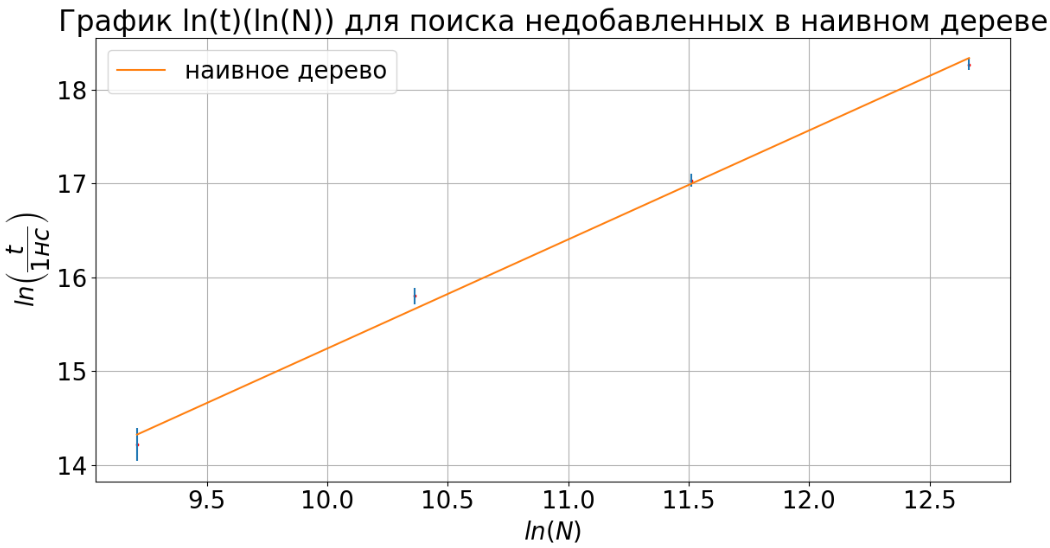


График хорошо аппроксимируется прямой вида $ln\left(\frac{t}{1\text{нс}}\right) = kln(N) + b$. Из графика находим:

$$k = (1.16 \pm 0.05)$$

$$b = (3.6 \pm 0.6)$$

То есть в предположении, что $t(N)$ - степенная, сложность поиска элемента, которого нет в дереве $O(N^{0.16})$

Построим график $t(Nln(N))$ для наивного дерева.

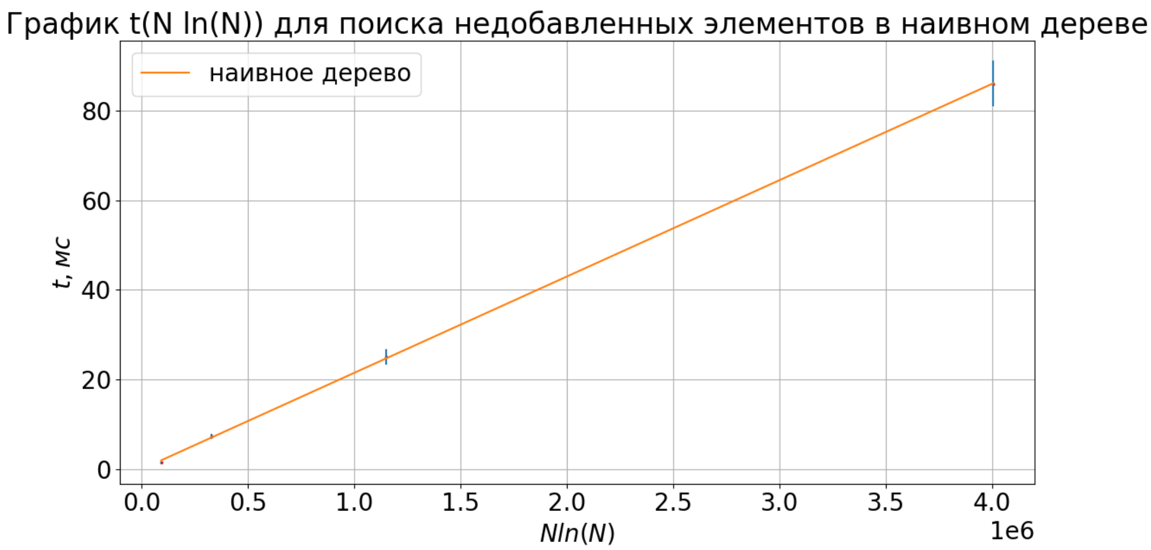


График аппроксимируется прямой вида $t = t_0(kNln(N) + b)$, где $t_0 =$ 1(нс)

Из графика находим

$$k = (21.50 \pm 0.14)$$

$$b = (0 \pm 3) \cdot 10^5$$

То есть данная модель тоже хорошо описывает поиск элемента в наивном дереве, заполненном случайными данными.

Так как в данном случае рассматривается одна и та же модель для обоих деревьев, то можем сравнить их характерные времена поиска одного элемента $\approx kt_0$. Для АВЛ дерева это время примерно в 1.6 раз меньше.

Итак, теоретическая зависимость для АВЛ дерева подтверждается, а для наивного дерева в пределах погрешности применимы две модели: сложность поиска одного элемента либо $O(N^{0.16})$, либо $O(ln(N))$. Во втором случае относительная погрешность коэффициентов меньше, поэтому эта модель точнее.

4 Вывод

Были проведены исследования для наивного и АВЛ деревьев. Сложность заполнения обоих деревьев получилась $O(N)$ для всех типов данных, причем наивное дерево заполняется быстрее, так как в нем нет самобалансировки.

Сложность поиска элемента в АВЛ дереве $O(ln(N))$ для всех типов данных и даже для элементов, которых нет в дереве.

Для наивного дерева, заполненного отсортированными данными, сложность поиска $O(N)$, так как оно эквивалентно списку. Для заполненного случайными данными большую точность дает модель, в которой поиск имеет сложность $O(ln(N))$