Алгоритм ннахождения медианы медиан - оптимизация выбора опорного элемента, которая дает возможность в худшем случае сортировать массив быстрой сортировкой за $O(NlogN)$ времени.

Алгоритм медианы медиан вычисляет приблизительную медиану, а именно точку, которая гарантированно находится между 30-м и 70-м процентилями. Таким образом, набор поиска уменьшается как минимум на 30 процентам. Проблема сводится к 70 процентам от первоначального размерае. Применение того же алгоритма к теперь меньшему набору рекурсивно, пока не останется только один или два элемента.

Алгоритм нахождения k порядковой статистики через медиану медиан:

1. Все n элементов входного массива разбиваются на $[n/5]$ групп по 5 элементов и одну группу, содержащую оставшиеся $nmod5$ элементов (эта группа может оказаться пустой).

2. Сначала методом сортировки вставкой сортируется каждая из $[n/5]$ групп (содержащих не более 5 элементов), а затем в каждом отсортированном списке выбирается медиана.

3. Путем рекурсивно определяется медиана x множества из $[n/5]$ медиан, найденных на шаге 2. (Если этих медиан окажется четное количество, то переменной x будет присвоено значение нижней медианы.)

4. С помощью *partition* делим массив относительно x.

5. Рекурсивно вызывается для подмассива, в котором все элементы меньше pivot, и для подмассива, в котором все элементы больше или равны pivot.

Лемма: Медиана медиан гарантированно делит массив в соотношении не хуже чем 3 : 7.

Как минимум половина медиан, найденных на шаге 2, больше или равны медиане медиану x . Таким образом, как минимум $n/5$ групп содержат по 3 элемента, превышающих величину x, за исключением одной группы, в которой меньше пяти элементов и еще одной группы, содержащей сам элемент x. Приходим к выводу, что количество элементов, величина которых превышает x, равно как минимум $\frac{3N}{10}$.

Рекуррентное соотношение для времени работы алгоритма медианы медиан в наихудшем случае. Для выполнения шагов 1, 2 и 4 требуется время $O(n)$ (шаг 2 состоит из $O(n)$ вызовов сортировки вставкой для множеств размером O(1)). Выполнение шага 3 занимает время $(n/5)$, а выполнение шага 5 — время не более $T(\frac{7n}{10})$.

$$T(n) <= T(\frac{n}{5}) + 7T(\frac{n}{10}) + O(n)$$

Покажем методом подстановки , что время работы линейно зависит от количества входных элементов. Покажем, что для некоторой небольшой константы c выполняется неравенство $T(n) <= nc$. Также выберем константу a, чтобы функция $O(n)$ была ограничена сверху an.

$$
\begin{aligned}
T(n) &\leq c \cdot \frac{n}{5} + c \cdot \left(\frac{7n}{10}\right) + an \\
&\leq \frac{cn}{5} + \frac{7cn}{10} + an \\
&= \frac{9cn}{10} + an \\
&= cn + \left(\frac{-cn}{10} + an\right)
\end{aligned}
$$

Следовательно, выличина $T(n) \leq cn$ верно, если взять $c >= 10a$.

Поскольку такое c можно выбрать, то алгоритм работает за линейное время.