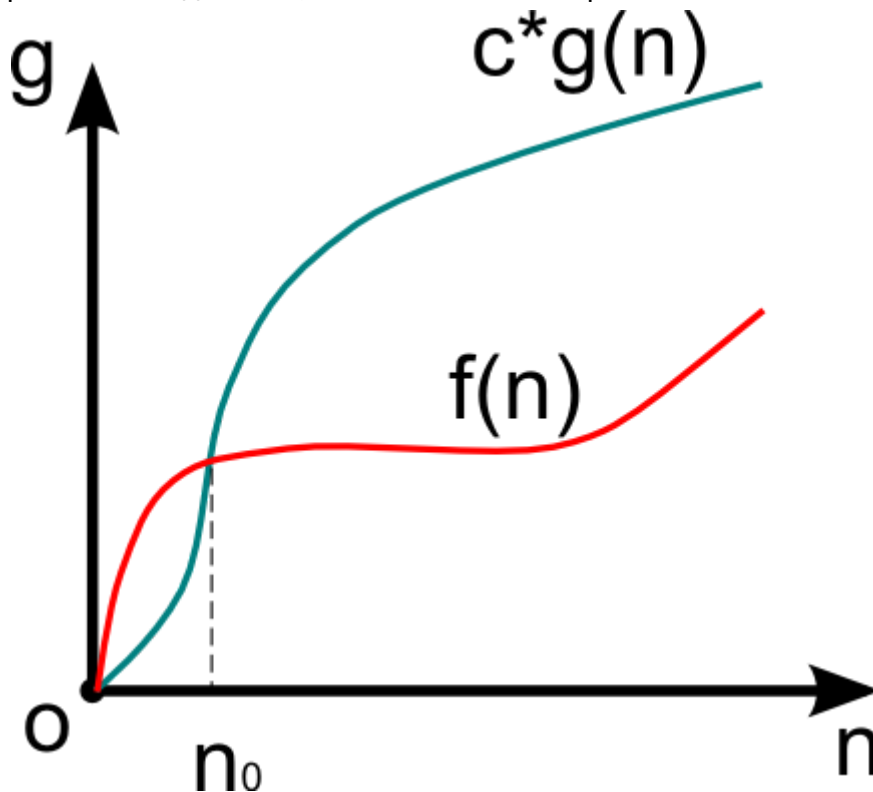


Формально, чисто математически, верхней оценкой называется функция, которая растет не медленнее, чем сложность алгоритма.



Источник: <https://habr.com/ru/post/173821/>

Иными словами, верхняя оценка - это функция $g(n)$, которая начиная с какого-то момента всегда выше, чем $f(n)$ - сложность алгоритма, с точностью до произвольного множителя c . Если говорить еще проще - верхняя оценка - это потолок (по времени), выше которого мы точно не залезем. Записывается это как

$$f(n) = O(g(n))$$

Что означает, что функция $f(n)$ растет не быстрее, чем функция $g(n)$, или, если хотите по-научному, что $g(n)$ - асимптотическая оценка сложности работы алгоритма.

Правила работы с O-нотацией

Существует несколько правил, облегчающих анализ алгоритмов при помощи O-нотации.

- **Сумма.** "O" большое суммы функций является эквивалентным "O" большому от наиболее быстрой функции. Пример:
 $O(n^2 + n^3) = O(n^3)$
- **Произведение.** Произведение "O" большого от функций равно "O" большому от произведения этих функций. Пример:
 $O(n) * O(\log(n)) = O(n * \log(n))$
- **Умножение на число.** Числовой множитель не влияет на асимптотическую сложность
 $5 * O(n) = O(5n) = O(n)$