**36. Асимптотическая оценка сложности алгоритмов. Верхняя, средняя и нижняя оценки.**

Зачем анализировать: чтобы оценить качественные и количественные свойства алгоритма, что позволит сравнивать алгоритмы между собой.

Критерии:

* Кол-во переданных по сети данных
* Требования к аппаратному обеспечению
* *Временные затраты//базовый*
* *Затраты памяти// базовый*
* Объём вспомогательных данных
* И т.д.

На практике наиболее часто оценивают именно временную сложность алгоритма

* В худшем случае (поиск отсутствующей в БД информации)
* В среднем (чаще всего, для случайных данных)

Сложности при среднем случае

* Средний случай-математическое ожидание времени работы (требуется вероятностный анализ с применением серьёзного мат. аппарата)
* Оценка рандомизированных алгоритмов требует нетривиального мат. Аппарата

Для того, чтобы качественно оценивать алгоритмы, нужна мера Порядок роста. «Оценить» порядок роста –значит найти такую функцию, которая бы с ростом значения аргумента увеличивалась точно также, как и время работы алгоритма с увеличением входных данных.

**Асимптотическая оценка(эффективность)** - порядок роста временных затрат в пределе (при бесконечном увеличении входных данных).  При выборе подходящей функции можно пренебречь константами и членами с меньшей скоростью роста.

**Асимптотически точная оценка θ(g(n))-оценка в среднем.**

Для некоторой f(n) выражение f(n) = θ(g(n)) означает, что существует некоторые с1, с2 и n0, что 0 <=c1\*g(n)<=f(n)<=c2\*g(n) для всех n=>n0 (нужно найти 2 кривых, которые бы зажимали функцию роста между собой сверху и снизу)

**Верхняя оценка Оg(n)). -оценка в худшем случае**

Для некоторой f(n) выражение f(n) = О(g(n)) означает, что существует некоторые с и n0, что 0<= f(n) <=c\*g(n) для всех n=>n0 ( есть некоторая функция, которая будет мажорировать(преобладать, ограничивать) начиная с n0 для всех аргументов над той оценкой, которую мы выбрали)

**Нижняя асимптотическая оценка Ω(g(n))-в лучшем случае**

Для некоторой f(n) выражение f(n) = **Ω** (g(n)) означает, что существует некоторые с и n0, что 0<=c\*g(n) <= f(n) для всех n=>n0 (всегда лежит под функцией оценки. Нужна для поиска средней оценки(сближаем оценки в хс и лс, там где сойдутся - средняя)).

Строгие оценки

* Верхняя о(g(n)).

Для некоторой f(n) выражение f(n) = о(g(n)) означает, что для любой положительной константы с существует n0,>0, что 0<= f(n) <=c\*g(n) для всех n=>n0

* Нижняя ω(g(n)).

Для некоторой f(n) выражение f(n) = ω (g(n)) означает, что для любой положительной константы с существует n0,>0, что 0<=c\*g(n) <f(n) для всех n=>n0