

Лекция №11

Сложность алгоритмов в примерах и переборные алгоритмы

1. Сложность алгоритмов, выраженная в секундах. Пример:

Количество элементов	10 000	50 000	100 000	1 000 000	10 000 000
Метод выбора	0 с	9 с	34 с	–	–
Пирамидальная	0 с	0 с	0 с	1 с	13 с

2. Полный перебор: перебор всех вариантов решения задачи. Перебор с отсечением: полный перебор, в котором целые ветви дерева вызовов отсекаются проверкой какого-либо условия. Эвристический алгоритм: алгоритм, дающий верное или близкое к верному решение в большинстве случаев.
3. Задачи, связанные с обходом графа в глубину. Обход в глубину \implies перебор всех путей \implies поиск максимального и минимального пути за $O(N!)$. Для максимального это нормально; для минимального, пользуясь дополнительной информацией (любая часть минимального пути — минимальный путь) существуют намного более быстрые алгоритмы (Фloyd, Дейкстра). Усложнение: требуем посещения всех вершин (задача коммивояжера) — только $O(N!)$. Отсечение: если текущая длина пути превысила *рекорд* — прекратить обработку данного пути, откатиться назад. Эвристика: жадный алгоритм (идём всегда в ближайшую вершину).
4. Построение правильной раскраски вершин графа. Полный перебор ($O(N!)$). Эвристический алгоритм: каждую вершину красить в минимально возможный цвет.
5. Не все *йогурты* алгоритмы одинаково полезны. Полный перебор — обычно последнее средство. Примеры: счастливый билет, вычисление определителя матрицы.
6. Динамическое программирование: иллюстрация «золотого правила» в применении к программированию. Алгоритм Ахо, некоторые другие задачи на динамическое программирование.