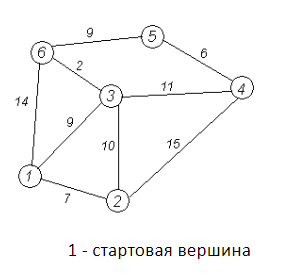
**Билет №5**

**Алгоритм Дейкстры. Постановка задачи, описание алгоритма за O(V^2 + E)**

**Постановка задачи:** Дан ориентированный или неориентированный взвешенный (у каждого ребра есть свой вес) граф с n вершинами и m ребрами. Веса всех ребер неотрицательны. Указана некоторая стартовая вершина s. Нужно найти длины кратчайших путей из вершины s во все остальные вершины графа:



**Описание алгоритма:**

1. Сопоставим каждой вершине 2 значения: 1-минимальное известное расстояние от стартовой до текущей (например:1,3 = 9), 2- массив visit[v] = true/false (посетили ли мы эту вершину или нет)
2. Определяем минимальные расстояния от вершины s до остальных: до самой себя = 0, до других = ∞ (на начале). Все вершины графа изначально не посещённые (visit[v] = false)
3. Будем посещать вершину расстояние от s до которой минимально. Если такой вершины нет, что не существует такого пути не равного ∞, то алгоритм завершается и вершина будет не посещена. А если нашлась такая вершина, то она будет посещена.
4. Пытаемся улучшить расстояние для соседних вершин суммой расстояния до выбранной вершины и длины ребра соседа. Те рассмотреть обходные пути к вершине через другие вершины (пример вершина 3). = пытаемся найти посредника
5. Теперь когда мы минимизировали кратчайший путь до какой-то вершины, то возвращаемся к пункту 2.

**Временная сложность:**

Представим наш алгоритм как 2 части: нахождение минимального пути на каждом шаге, перебор всех соседей

Существует 2 варианта реализации алгоритма Дейкстры: 1) искать минимум перебором всех вершин; 2) хранить пары (расстояние вершина) в куче на минимум

Рассмотрим 1 вариант, тк он самый простой для того, чтобы найти минимум нужно перебрать все вершины – O(n). Но всего таких шагов, когда нам надо каждый раз перебирать n вершин может быть максимум n, тк мы каждый раз можем находить вершину, которая может являться минимумом зачёт нее оптимизировать другие вершины дальше опять также и также и таких не больше n. => для этого понадобиться O(n^2). Также перебор каждого соседа займет O(m) => **O(n^2+m)**

**Задача: Изображение выглядит как текст

Автоматически созданное описание**

**Решение:** <https://pastebin.com/GcNwm7bz>