10.

Задача: построить алгоритм, который находит оптимальный маршрут в задаче ТСР в форме задачи распознавания за полиномиальное время.

Дано: Алгоритм $\mathcal{A}$, принимающий на вход конечное множество вершин $\mathcal{V}$ и функцию стоимости $\mathcal{V} \times \mathcal{V} \to \mathbb{N}$. Будем искать алгоритм с таким же входом. Входные данные можем представлять как полный взвешенный граф.

Решение:

1. Выбираем произвольную вершину $v \in \mathcal{V}$. Запускаем на входных данных алгоритм $\mathcal{A}$ и запоминаем стоимость оптимального маршрута.

2. Перебираем все пары рёбер, инцидентных этой вершине. Это можно сделать за $O(n^2)$ (где n - мощность множества $\mathcal{V}$). Для каждой пары рёбер мы "удаляем"все остальные рёбра, которые инцидентны вершине v. Под "удалением"будем понимать следующее: модифицируем весовую функцию, увеличив веса всех других рёбер, инцидентных вершине v на величину суммарного веса всех рёбер. И запускаем для исходного множества $\mathcal{V}$ и модифицированной функции весов алгоритм $\mathcal{A}$. Если значение стоимости оптимального маршрута не изменилось, значит зафиксированная пара рёбер принадлежит оптимальному маршруту.

3. Обозначим концы найденных рёбер за v_1 и v_2. Берём $v1$, и для него перебираем $n-1$ инцидентное ребро, рассматривая пары рёбер $v - v_1, v_1 - v_i$, также, как и в пункте (2). Таким образом находим все рёбра, принадлежащие оптимальному маршруту. Заканчиваем перебор, когда второй конец найденного ребра равен v_2.

4. При помощи описанной выше процедуры находим последовательность вершин из $\mathcal{V}$, которая задаёт оптимальный маршрут, алгоритм работает за $O(n^2 p(n))$, где $p(n)$ - полиномиальная оценка сложности для алгоритма $\mathcal{A}$.