**1** (2). В ориентированном взвешенном графе есть ровно одно ребро $(u \to v)$ с отрицательным весом. Описать эффективный алгоритм поиска кратчайшего пути между заданной парой вершин $(a, b)$ — вход задачи: матрица весов и вершины $a$ и $b$.

**2**(3). Дан неориентированный граф $G = (V, E)$, веса рёбер которого не обязательно различны. Для каждого из утверждений ниже приведите доказательство, если оно истинно, или постройте контрпример, если оно ложно:

**а)** Если самое лёгкое ребро графа $G$ уникально, то оно входит в любое минимальное остовное дерево.

**б)** Если ребро $e$ входит в некоторое минимальное остовное дерево, то оно является ребром минимального веса из пересекающих некоторый разрез.

**в)** Кратчайший путь между двумя вершинами является частью некоторого минимального остовного дерева.

**Определение.** Граф, который получается из графа $G$ удалением некоторых вершин и рёбер, называют *(рёберным) подграфом* графа $G$. В случае, если при изготовлении подграфа, рёбра удалялись только вместе с удалением вершин, подграф называют *индуцированным*.

**3** (4). Пусть $T$ — минимальное остовное дерево графа $G$, а $H$ — связный подграф $G$. Покажите, что рёбра, входящие как в $T$, так и в $H$, входят в некоторое минимальное остовное дерево графа $H$.

**4** (3). Рассмотрим алгоритм Union-Find без улучшения со сжатием путей[1]. Приведите последовательность из $m$ операций Union и Find над множеством из $n$ элементов, которая потребует времени $\Omega(m \log n)$.

**5**(3). Новый оператор мобильной связи выходит на рынок целой страны. Он уже закупил и расставил свои сотовые вышки, которые нужно соединить в единую сеть. Также в каждом городе есть городская (уже связанная) сеть вышек, с которым оператор может соединять и свои вышки. Соединение пары вышек $(i, j)$ стоит $c_{i,j} = c_{j,i}$. Постройте эффективный алгоритм, который находит минимальное нужное число соединений минимальной суммарной стоимости.

**6** (4). На вход задачи подаётся неориентированный взвешенный граф $G(V, E)$ и подмножество вершин $U \subseteq V$. Необходимо построить остовное дерево, минимальное (по весу) среди деревьев, в которых все вершины $U$ являются листьями (но могут быть и другие листья) или обнаружить, что таких остовных деревьев нет. Постройте алгоритм, который решает задачу за $O(|E| \log |V|)$. Обратите внимание, что искомое дерево может не быть минимальным остовным деревом.

**7** (6). Курьеру Ozon требуется построить эффективный маршрут, начинающийся на складе, посещающий всех клиентов из списка, и возвращающийся на склад. Программист Ozon знает, что эта задача непростая (она NP-полна), но он сообразил как построить 2-приближённый алгоритм, решающий эту задачу. Ключевым моментом решения

[1] В этом улучшении при вызове Find$(x)$ все предки $x$ вместе с $x$ становятся детьми корня.

является следующее наблюдение: для любых точек маршрута $u, v, w$ справедливо неравенство $\rho(u, v) \leqslant \rho(u, w) + \rho(w, v)$, где $\rho(x, y)$ — расстояние между точками маршрута $x$ и $y$. Помогите программисту Ozon построить линейный 2-приближённый алгоритм, решающий задачу (т.е., вес решения, которое находит алгоритм, должен не превышать удвоенное оптимальное значение веса). Вход: взвешенный граф, вершины которого — точки маршрута, а веса — расстояния между точками маршрута; между любыми двумя точками маршрута расстояние определено. Требуется минимизировать сумму весов вдоль пути курьера (который обязан посетить каждую точку маршрута ровно один раз).