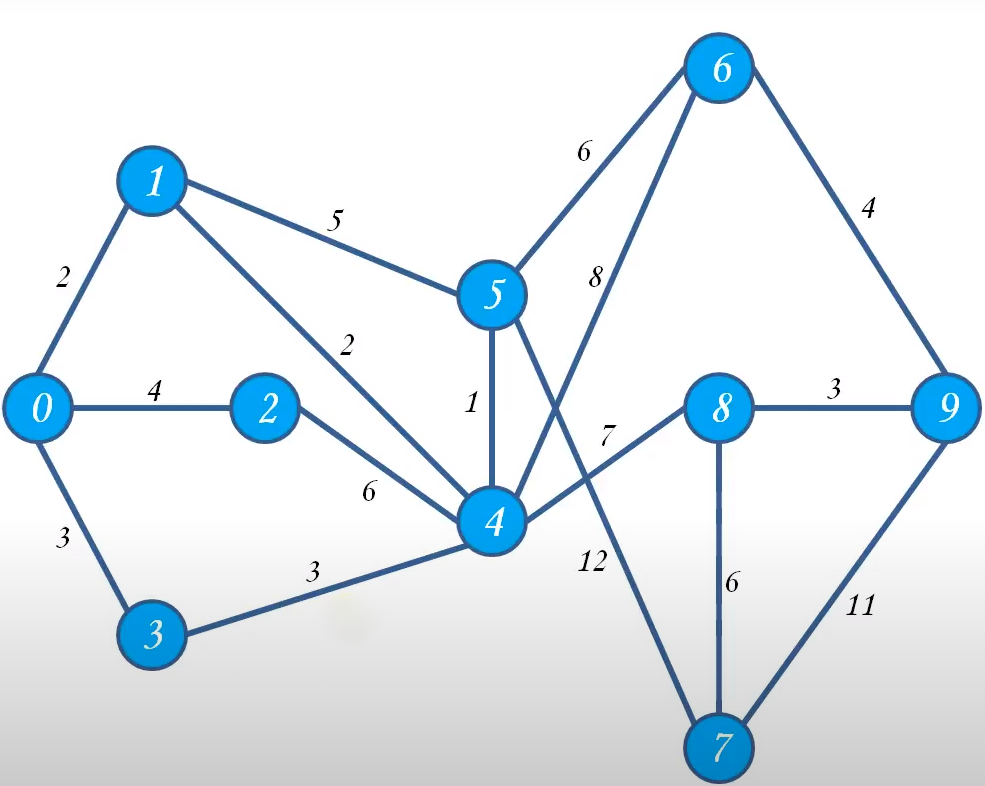
Алгоритм Краскала

Здравствуйте, сейчас мы разберем алгоритм Краскала, он служит для нахождения минимального остовного дерева на графе, и используется чаще чем алгоритм Прима, т.к. в реальности его применение гораздо быстрее и тратит меньше времени.

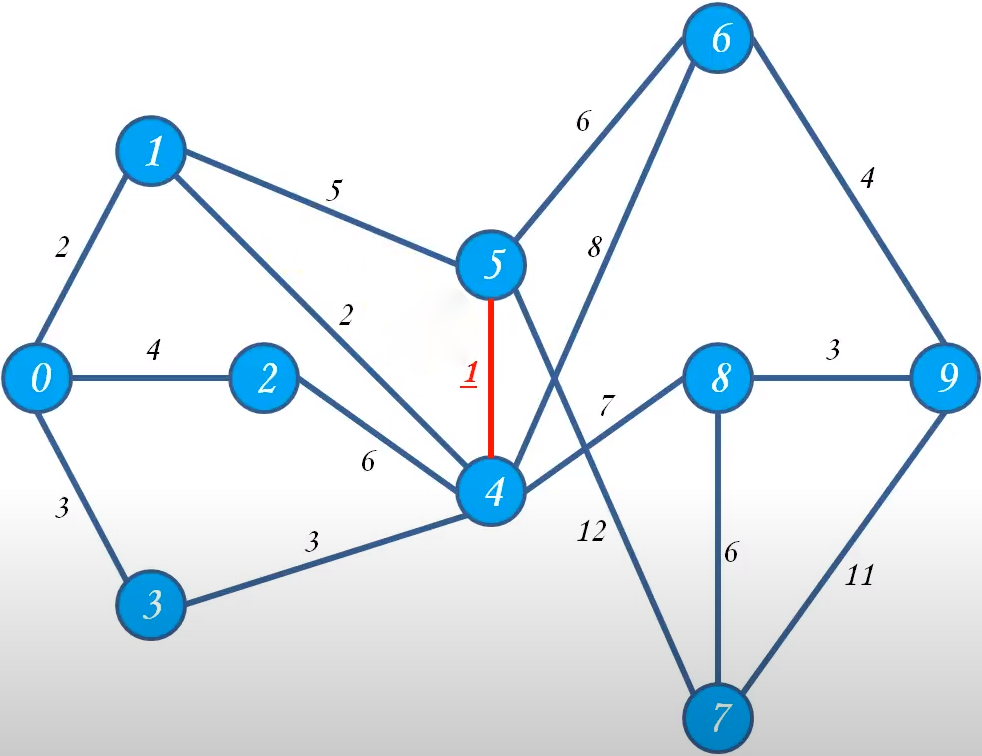
Граф с каким мы будем работать связанный (от любой вершины к каждой), неориентированный (в дугах нет направления), взвешенным (каждая дуга имеет свой вес)*. (рис 1)*



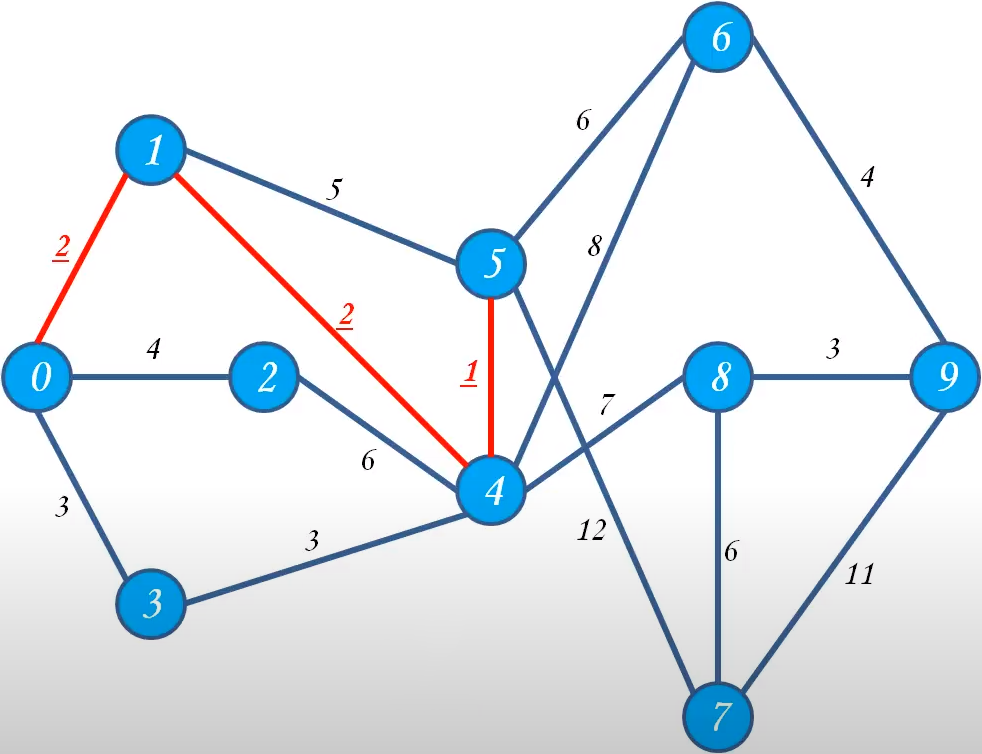
Напомню, что остовное дерево на графе – это такой связанный, ацикличный подграф, которые включает все вершины самого графа. В дальнейшем буду сокращать остовное дерево в остов.

Суть алгоритма заключается в том, чтобы мы находили дуги минимального веса из тех, что еще не включены в остов, а затем проверяли не будет ли включение данной дуги провоцировать наличие цикла в уже созданном нами остове. Иногда алгоритм объясняют следующим образом: следует представить каждую из вершин маленьким остовом, а затем все остовные деревья соединять по дугам наименьшего веса. (как кому удобно)

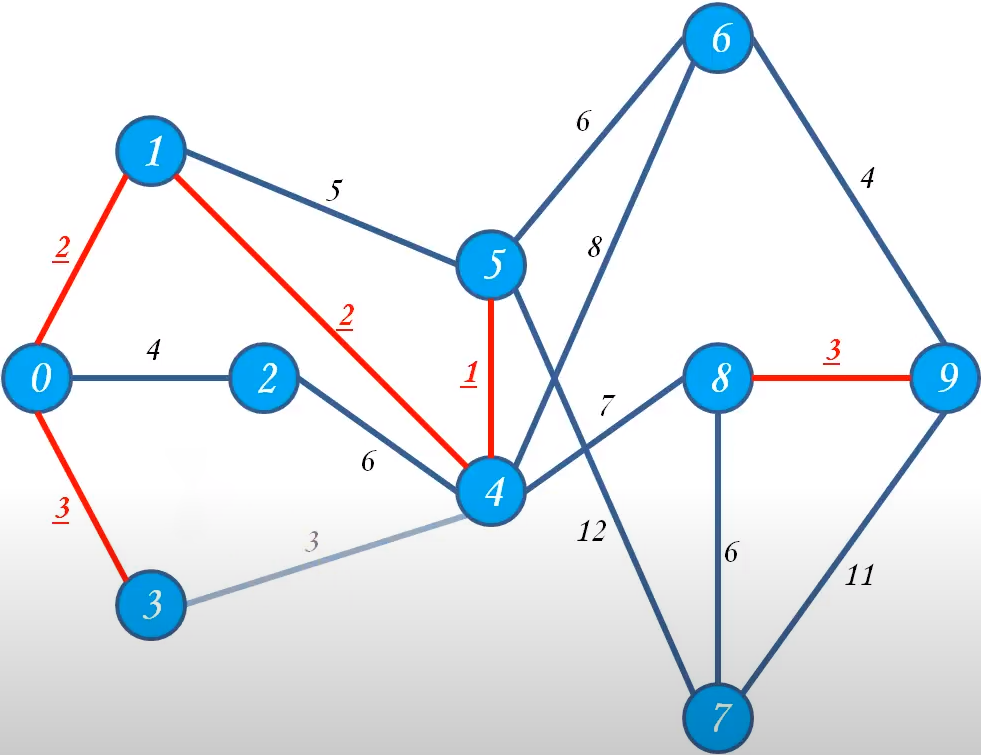
Допустим, у нас есть наименьшая дуга 4-5 (весом 1), мы ее сразу отмечаем. *(рис 2)*



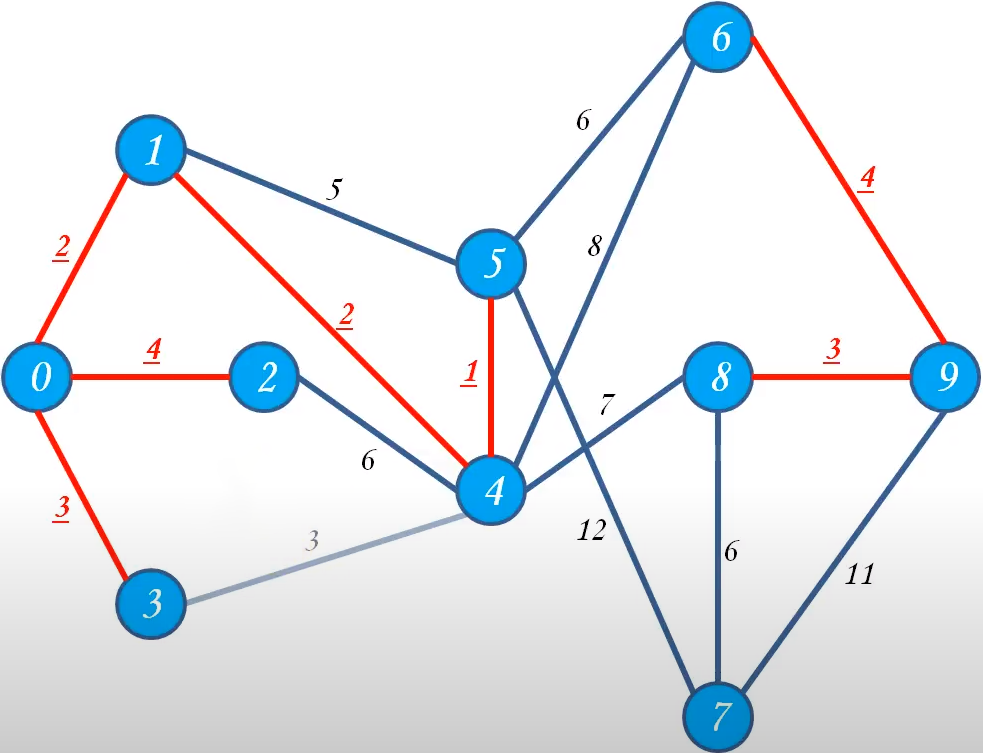
Далее мы должны найти другие минимальные дуги, но 4-5 мы уже не используем. То есть, минимальных дуг у нас получаются две с весом 2 (1-0, 1-4). Выбираем любую и смотрим, не делаем ли мы цикл. Если нет, то отмечаем, в противном случае – убираем. Т.к. ни 1-0, ни 1-4 цикла не создают, то отмечаем*. (рис 3)*



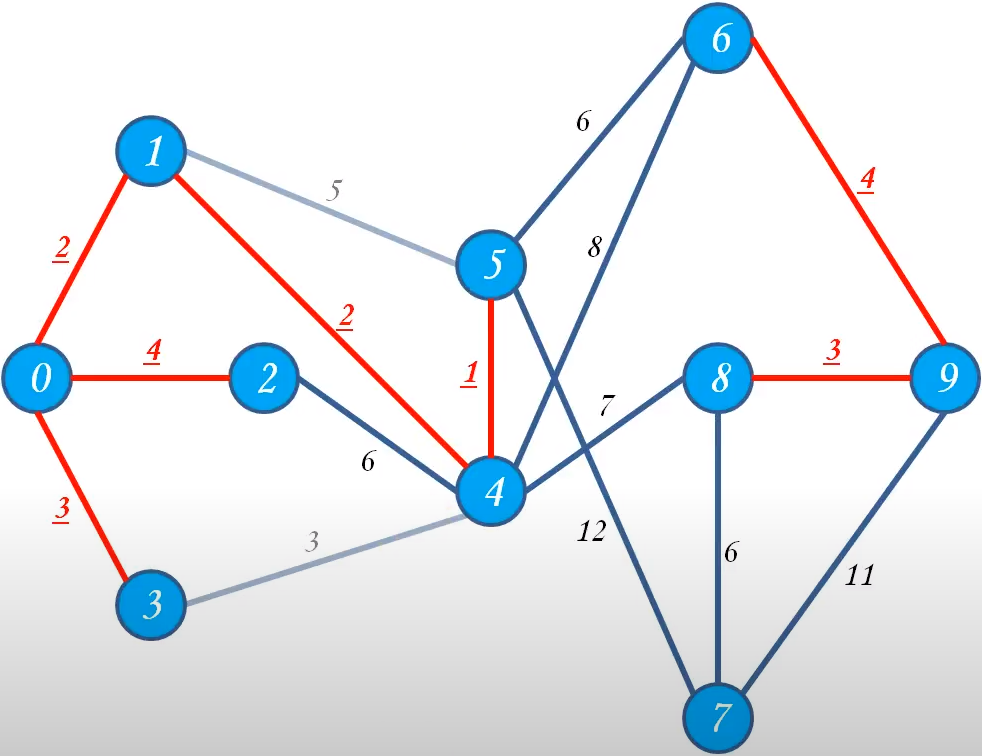
Далее ищем другие дуги, и их у нас целых три (0-3, 3-4, 8-9) с весом 3.Также можно выбрать любую, но я выберу дугу 0-3(цикла не создает).Также можно отметить дугу 8-9. А вот дугу 3-4 мы отметить не можем, так как получим цикл 4-1-0-3. Значит дугу 3-4 мы убираем и забываем. *(рис 4)*



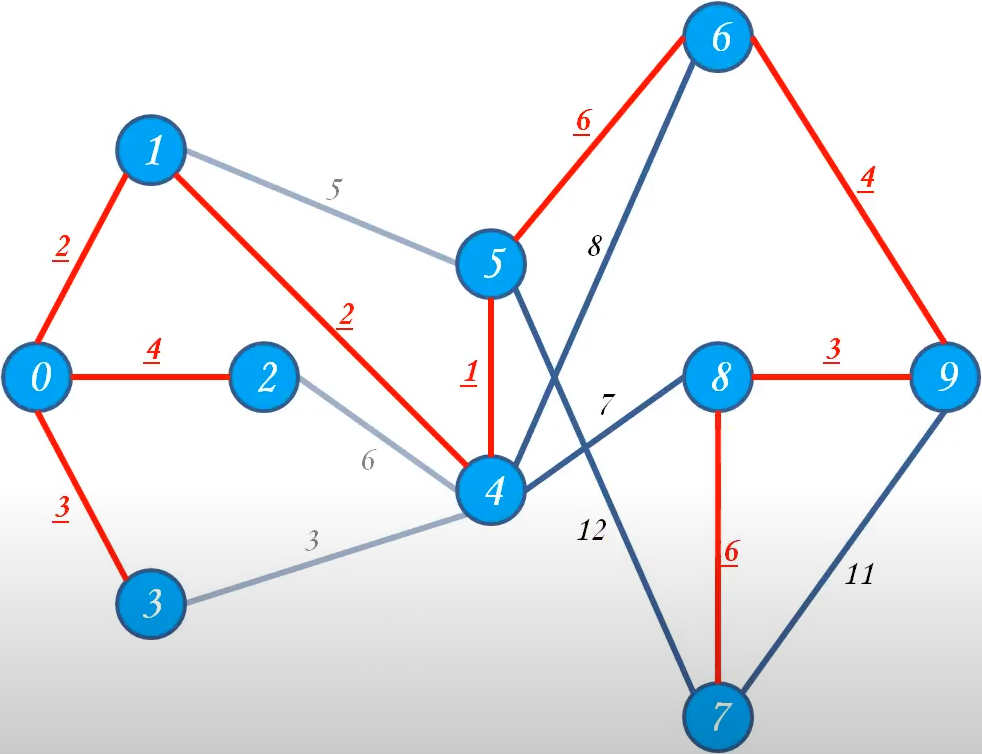
Продолжаем искать дуги с минимальным весом(сейчас это дуги 0-2 и 6-9 с весом 4). Ни одна из них на цикл не провоцирует, значит можем смело их отмечать. *(рис 5)*



Заново ищем минимальную дугу, из тех дуг, что еще не включены или убраны (это дуга 1-5 с весом 5). Но мы видим, что при включении дуги 1-5, получится цикл 1-5-4. Значит, данную дугу убираем. *(рис 6)*

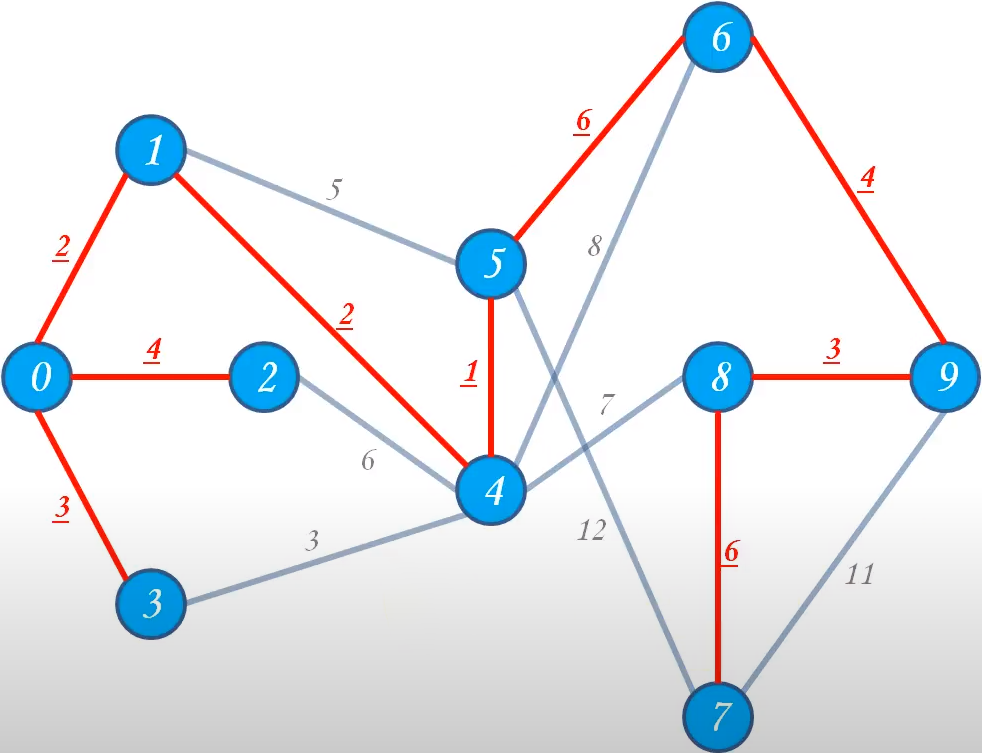


Повторяем процесс, находим дуги 2-4, 5-6, 8-7 с весом 6. Сразу видим, что дуга 2-4 образует цикл, значит убираем. А вот дуги 5-6, 8-7 не образуют цикл, смело отмечаем. *(рис 7)*

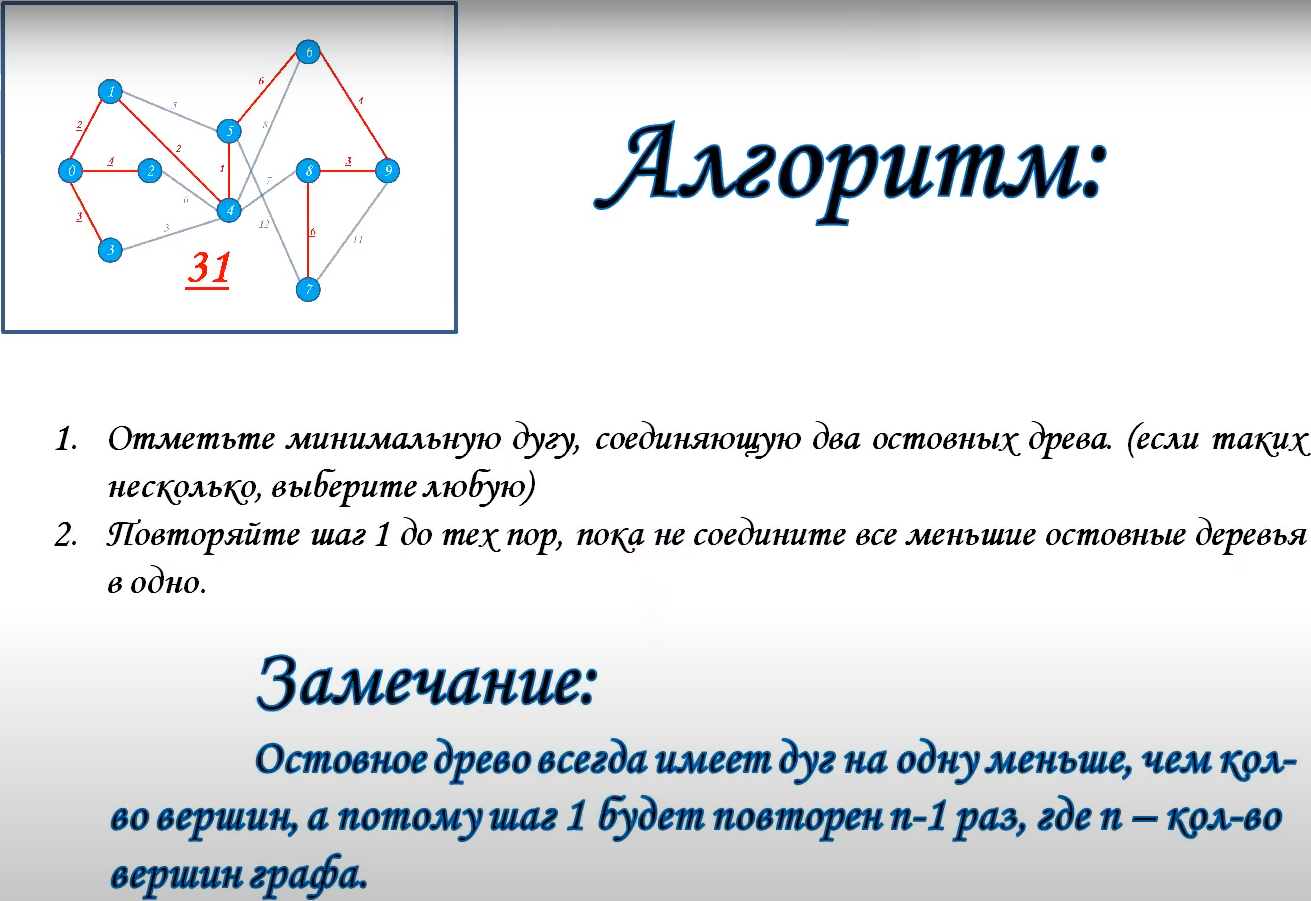


*Замечание: Если вы разбираетесь с этим алгоритмом, в виде маленьких остовных дереьев, то вам не имеет смысла искать циклы, т.к. соединение двух различных оставных деревьев, никогда не дадут цикл.*

Вот и у нас в остовном дереве уже есть все 10 вершин, которые были в изначальном графе, поэтому можем сделать вывод, что все остальные незадействованные дуги нам попросту не нужны. Убираем. *(рис 8)*



*Весовая сумма нашего оставного графа равна* ***31. (рис 9)***



***Автор и исполнитель на практике ТМОИС: Даниил Карагодин, ИИ-21***