Асимптотика алгоритма.
Инициализация ребер, т.к. Мы исследуем полный граф, проходит за O(N^2) (N - число вершин в графе).
Затем проходит сортировка всех ребер. Мною была использована стандартная питоновская функция сортировки sort (Timsort внутри), её сложность в худшем случае - nlogn (n - число элементов). Сортируем ребра => ElogE.
Далее для каждого ребра происходит вузов функции union. Её сложность составляет сложность функции Find + O(1) операций. Find - рекурсивная функция, её глубина не превосходит глубины дерева O(logN) (за счет применения эвристик сжатия пути и подвешивания более низкого дерева к более высокому). Таким образом, сложность функции Union O(logN) + O(1) = O(logN), а сложность функции getCount O(ElogN). Итого, сложность обработки одного Test Case: O(ElogE) + O(ElogN) = O(ElogE) = O(N^2logN).

Доказательство корректности алгоритма:
Требуется соединить все города при помощи дорог минимальной длины => решением будет MST для полного графа, вершины в котором - города, а веса ребер - расстояния между ними . По мере построения MST, мы суммируем длины обычных дорог и железных дорог, числом штатов является число железных дорог + 1 (т.к. В остовном дереве циклов нет).
Используя тот факт, что основное дерево является базой графового матроида (по определению) и теорему «Если область истинности предиката P - матроид S над A, то полученное жадным алгоритмом подмножество u_n доставляет максимум функционалу F», получаем, что алгоритм Краскала, являющийся жадным, корректен и находит MST. В предыдущих терминах S - все независимые подмножества ребер графа, A - множество ребер графа, F - сумма весов ребер, u_n - MST (база матроида). В нашей задаче мы минимизируем функционал F.