Complexity of Chu–Liu/Edmonds' algorithm

public SpanningTree getMinimumSpanningTree(Graph graph)

{

List<Set<Integer>> forest = initForest(graph.size()); **O(V)**

List<Edge> cyclicEdges = new ArrayList<>();

SpanningTree tree = null;

List<List<Edge>> cycles;

while (true) O(V)

{

tree = getMinimumIncomingEdges(graph, forest); O(E)

cycles = tree.getCycles(); O(V)

if (cycles.isEmpty())

{

addEdgesFromCycles(tree, cyclicEdges); Neg

break;

}

//Update and merge forest

forest = updateEdgeWeights(graph, cycles); O(E)

//Update all cyclic edges

addAll(cyclicEdges, cycles); Neg

}

return tree;

}

Neg = negligible

Summary: O(V\*(2\*E+V)+V) or **O(E\*V)**