

Documento de analisis

Alejandro Borda - a.borda@uniandes.edu.co - 202020727

Juan Sebastián Lache - js.lache@uniandes.edu.co – 201821331

Requerimiento 1:

$O(n \log n)$, en donde n representa el numero de vértices, esto se debe a que para encontrar los vértices con mayor grado se debe evaluar cada vértice y luego hacer merge sort sobre la lista de grados

Requerimiento 2:

Debido a que kosaraju tiene una complejidad de $O(V+E)$ con v el número de vértices y E los arcos esta seria la complejidad del requerimiento.

Requerimiento 3:

La complejidad de este requerimiento es en base a la relación entre el numero de arcos y vértices y hay dos posibilidades. Si el número de arcos es suficientemente menor al de vértices la complejidad es $O(V)$ pues se busca la distancia de cada aeropuerto a las 2 ciudades dadas, si la relación no es esta seria $O(E \log V)$, es decir la complejidad será el mayor de estos dos valores dependiendo del numero de arcos y vértices.

Requerimiento 4:

La mayor complejidad en este requerimiento viene de prim, con lo cual su complejidad será de $O(E \log V)$.

Requerimiento 5:

En su peor caso la complejidad es de $O(n^n)$ pues se evalúan los adyacentes de sus adyacentes hasta que se evalúen todos los adyacentes posibles y en su peor caso, a pesar de que se revise y no se agregue a la lista es posible revisar todos los vértices n veces si cada vertice es adyacente a todos los otros vértices. Sin embargo en la practica esto no se acerca en absoluto a el valor verdadero y $O(V)$ se acerca mas a el valor verdadero con una constante que podría ser el numero promedio de adyacentes y V el número de vértices del grafo.

Requerimiento 6:

La complejidad del algoritmo es $O(E \log V)$ pues los datos de los aeropuertos se obtienen de Amadeus y $O(E \log V)$ es la complejidad de Dijkstra.

Requerimiento 7:

Para cada requerimiento se evalúa una pequeña lista de vértices que en su peor caso es de V vértices y sería de complejidad $O(V)$.