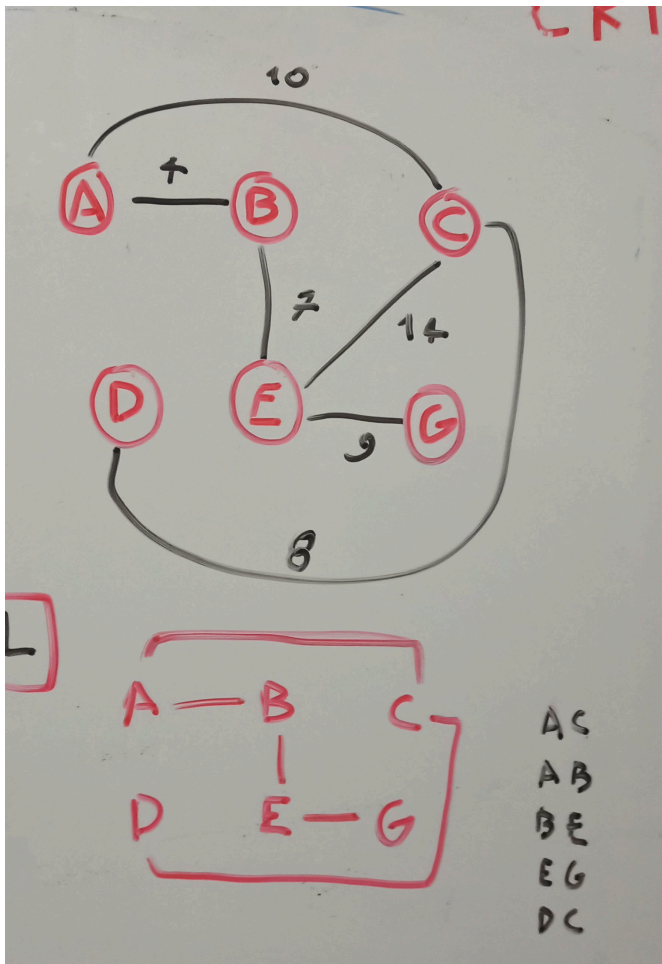


Algoritmos y Estructuras de Datos
UT8-PD2
Santiago Blanco
23-06-2025

Ejercicio #1

Ejemplo de Kruskal, hecho a mano, compare usted mismo:



Este mismo grafo que usted está observando fijamente, ha sido implementado en Java. Fijese en el código fuente, le va a encantar.

Análisis de Kruskal

Las estructuras utilizadas son:

- Linked List
- HashMap
- Map

→ Bueno, bueno, bueno. El algoritmo implementado es mediocre e ineficiente. El ordenamiento se hace mediante inserción en una lista enlazada, lo cual tiene un costo de $O(n^2)$ en el peor caso, siendo n la cantidad de aristas. Además, al unir las listas enlazadas, es necesario recorrer los elementos del conjunto destino para actualizar referencias, lo que implica un costo $O(m)$ por unión, siendo m el número de vértices. Por esto la complejidad total del algoritmo es de $O(m^2)$.

Mejoremus

1. Se puede mejorar la implementación usando `Collections.sort()` para evitar el orden manual de listas, con lo que se logra $O(m \log m)$.
2. Se usan `LinkedLists` donde podrían usarse arrays simples, para evitar un exceso en consumo de memoria.

Importancia del uso de memoria

El consumo de memoria en esta paupérrima implementación puede ser alto porque se usan listas enlazadas para representar conjuntos disjuntos, lo que genera muchas referencias duplicadas y por consiguiente desperdicio de memoria. En grafos grandes puede volverse una catástrofe.