

Trabajo Práctico Obligatorio

Asignatura: Programación 2

Profesor:

Nicolás Ignacio Perez

Integrantes del grupo 9:

Bautista Julian Ignacio (1134458)

Casais Nicolás (1185300)

Etchart Facundo (1201391)

Baud Perez Martin (1188871)

Martiniano Peralta Montiel (1196810)

Cuatrimestre: 1 – Año: 2025



Universidad Argentina de la empresa

Facultad de ingeniería y ciencias exactas

Introducción

En este trabajo se aborda la problemática de diseñar una red de datos interna para la nueva sede de la CIA (Central Intelligence Agency), una organización con altos requisitos de seguridad y eficiencia. El objetivo es conectar distintas divisiones estratégicas mediante una red optimizada que minimice el costo de implementación, considerando también el riesgo de cada conexión.

Se aplicará el algoritmo de **Kruskal**, uno de los algoritmos vistos en clase, que permite encontrar el árbol de expansión mínima relacionado a un grafo pesado, y se va a realizar utilizando el paradigma de tipo de dato abstracto (TDA).

Problema real y modelado con grafos

La organización cuenta con una número determinado de divisiones:

Estas divisiones son representadas como **nodos** en un grafo. Las conexiones posibles entre ellas son las **aristas**, y cada una tiene un peso calculado como:

$$\text{peso} = A \times \text{Costo} + B \times \text{Riesgo}$$

Donde:

- **El costo** representa el costo económico del cableado (mayor profundidad o distancia, mayor costo).

- **El riesgo** representa la vulnerabilidad o exposición de la conexión.

- **A y B** son coeficientes definidos por la política de seguridad del gobierno, donde $A+B=1$

Ejemplo: $A=0.4$ $B=0.6$, priorizando la seguridad.

El objetivo es obtener una red **totalmente conectada, sin ciclos** y con el **menor peso posible**, es decir, aplicar **Kruskal** para obtener el árbol generador mínimo del grafo definido.

Solución Propuesta: Implementación de Kruskal bajo TDA

La solución que hemos podido encontrar, se basa en el uso de las siguientes clases: Arista, Cable, División, Grafo y nodo, junto a sus respectivas interfaces y testeos (Todos en sus paquetes correspondientes respetando el paradigma de TDA). El planteo se basa en el uso de clases auxiliares para que se pueda visualizar de una manera más sencilla la solución. Por ejemplo la clase División se implementa para estar contenida en la clase Nodo, y de la misma manera la clase Cable se encuentra contenida dentro de la clase Arista. Una vez se tienen dichas clases definidas, se aplica el algoritmo de Kruskal para poder encontrar el árbol de expansión mínima y poder diseñar la red minimizando el costo de conexión de nodos.

Testeo bajo TDA

Se desarrollaron pruebas unitarias que validan:

- **Insertión de nodos y aristas.**
- **Cálculo correcto de pesos con diferentes valores de A y B.**
- **Ejecución de Kruskal y verificación de que la red resultante:**
 - **Está conectada.**

- **No contiene ciclos.**
- **Tiene el peso total mínimo posible.**

Conclusiones

El algoritmo de Kruskal demostró ser ideal para resolver esta problemática de diseño de red eficiente bajo condiciones extremas de costo y seguridad. La representación del problema con grafos pesados y la implementación bajo TDA aseguraron una solución robusta, flexible y probada. La fórmula ponderada de los pesos permitió adaptar el algoritmo a distintas prioridades de la CIA.

Bibliografía:

<https://www.geeksforgeeks.org/dsa/kruskals-minimum-spanning-tree-algorithm-greedy-algo-2/>

<https://www.wscubetech.com/resources/dsa/kruskal-algorithm>