Documentación Proyecto 2
Juan Pablo Reyes Romero – 202310852
Juan Sebastián Maldonado - 202311840

1. Re-contextualización del problema

Como se sabe, lo que se mencionó en el enunciado de este proyecto es que se requiere diseñar una red que logre conectar N computadores usando dos tecnologías de enlace; fibra óptica (k=1), y cable coaxial (k=2). Para ese propósito, cabe decir que se comenta que la comunicación entre dos nodos A y B, puede ocurrir si existe un camino específico de enlaces de la misma tecnología que conecte A con B.

Ahora bien, igualmente se menciona que dicha red es redundante, cuando para todo par de nodos A-B, A y B son conectables por fibra sí y sólo sí están conectables a través del cable coaxial. En otras palabras, si los conjuntos de pares conectables por fibra y por coaxial coinciden. Considerando eso mismo, se solicita determinar, tras cada inserción de conexión, si la red sigue siendo redundante.

2. Entradas y Salidas del problema

Una parte fundamental para la elaboración de la solución del problema es la explicación e indagación de la estructura y la forma que siguen las entradas y las salidas del problema. Por dicha razón, a continuación, se enuncian las correspondientes:

Entradas:

- N: en este caso, trata de la cantidad de nodos (o de acuerdo con el enunciado, la cantidad de computadores).
- Secuencia de m conexiones añadidas a lo largo del proceso. Cabe destacar que cada conexión descrita está dada por una tripla (i, j, k), con la siguiente restricción:

$$1 \leq i < j \leq N$$

Además, k puede tomar los valores {1,2}. Siendo 1 fibra óptica, y 2 cable coaxial.

Salidas:

- En este caso, al añadir una conexión, se imprime **1**, si la red es redundante en dicho momento, y **0**, en caso de que no lo sea. Cabe aclarar que estos dígitos se imprimen separados por un espacio.

3. Explicación del algoritmo empleado

De forma general, la idea fue mantener dos estructuras de Union-Find, específicamente DSU´s o Disjoint Set Union, las cuales representan las conexiones tipo fibra óptica y de cable coaxial respectivamente. La gracia es que, en el proceso, luego de cada unión, o inserción de un nuevo enlace, se comprueba si las particiones inducidas por **dsu_f** y **dsu_c** coinciden. En caso de que, si coincidan, es decir, que exista una correspondencia de tipo uno a uno entre componentes de fibra, y componentes de coaxial, entonces la red si sería redundante.

<u>¿Por Qué Se Usan Disjoint Set Union – DSU´s?</u>

La razón de uso de este tipo de estructura es que puede ejecutar el proceso de unión de forma repetitiva sin problema, haciendo la solución bastante dinámica y eficiente. Adicionalmente, para comprar particiones entre cada tecnología, por cada nodo, se puede consultar rápidamente la componente a la que pertenece.

Proceso General

Inicialmente, lo que se realiza es la actualización de la partición correspondiente;

- En caso de que k == 1, se une i y j en la DSU de fibra
- En caso de que k == 2, se une i y j en la DSU de cable coaxial

Luego, se obtiene la vista actual de las particiones, en donde para cada nodo **v,** se consulta su representante en cada DSU. Ello se hace a través de las siguientes funciones elaboradas:

- a = find(v) en el DSU de fibra
- b = find(v) en el DSU de cable

Ahora, dichos "representantes" identifican los bloques o conjuntos de la partición, en donde, todos aquellos que tengan un mismo **a** forman una componente de tipo fibra óptica. Por el contrario, todos aquellos que cuenten con el mismo valor en **b,** forman una componente diferente por cable coaxial.

Finalmente, una vez se finaliza lo anterior, se comparan las particiones a través de dos estructuras de datos, que son dos diccionarios dict12, y dict21. Lo que se hace es que al recorrer cada v, se establece la condición de que todos los nodos del mismo representante de fibra, deben señalar al mismo representante de cable coaxial, y viceversa. Si se llega a cumplir dicha condición entonces la red sería redundante, sin embargo, en caso de no cumplirse, sería no redundante.

4. Complejidad Temporal de la solución

Complejidad de operaciones de la clase DSU:

Sobre las operaciones Union-Find, se puede decir que cada una es **O(N)**, siendo N la cantidad de nodos totales. La razón de ello es que la función find como tal es instantánea, ya que las estructuras de datos permiten el acceso directo en toda búsqueda. Por otro lado, el Union evita que la red se desbalancee, por lo que se termina siempre acotando toda operación de recorrido, volviéndola O(N).

Complejidad de la función es_Redundante:

La complejidad de esta función es de **O(N)** igualmente, esto es ya que para cada computador **v**, se realizan dos operaciones find (O(n)), que corresponden a las que sean hacen en los DSUs de las tecnologías (una en cada una). Cabe decir que ese es el costo asintótico más grande, ya que los procesos que se llevan a cabo en los diccionarios tienen un costo de O(1), pues son operaciones instantáneas. Por lo que, en el peor caso, la complejidad inherentemente es O(N).

Complejidad del proceso de inserción:

En el momento en el que se agrega una conexión de tipo (i, j, k), se ejecuta la operación de Union, pero como ya se ha mencionado desde antes, dichas operaciones cuestan O(1). Ahora bien, en el llamado a **es_redundante,** como se mencionó previamente, se tiene un costo de O(N), pero considerando el hecho de que se realiza este proceso de inserción **m** veces, la complejidad del algoritmo de forma total es de O(m*N). Por eso mismo…

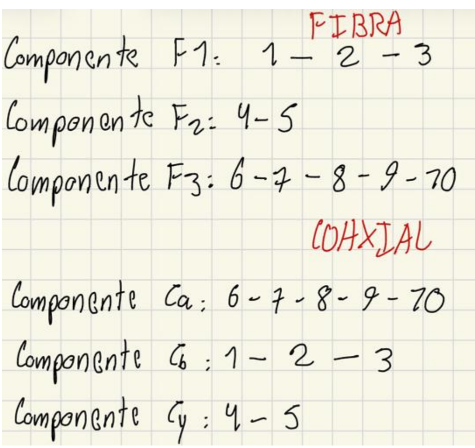
Complejidad temporal asintótica (peor caso) de la solución: O(mN).

5. Complejidad Espacial de la solución:

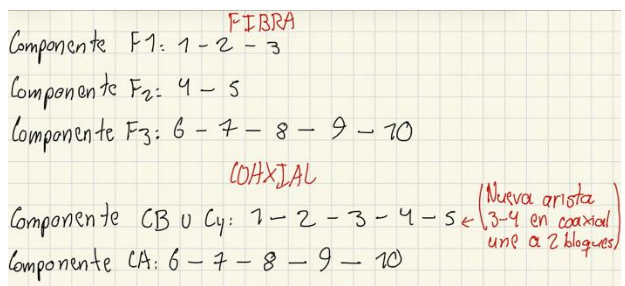
Sobre el espacio que utiliza la solución, cabe decir que se tienen en total cuatro estructuras en total. Las dos DSU´s, que tienen un costo de **O(N),** y los dos diccionarios de uso temporal que son utilizados en cada verificación, con costo de **O(N)** igualmente. Por eso mismo se puede afirmar de forma asintótica que…
Complejidad espacial asintótica (peor caso) de la solución: O(N).

6. Gráfico de apoyo

Caso A: Redundante.



Caso B: No redundante



Caso A	Caso B
En este estado, cada componente por FIBRE corresponde uno a uno a un componente por COAXIAL con exactamente el mismo conjunto de nodos. es_redundante retorna True, imprime 1 tras la inserción.	La partición inducida por coaxial es {1,2,3,4,5} y {6,7,8,9,10} mientras que por fibra sigue siendo {1,2,3}, {4,5}, {6,7,8,9}. - No hay correspondencia 1-1 con los mismos conjuntos. - es_redundante retorna False, imprime 0 tras la inserción.

Imagen 1 / Gráfico solución problema Grafos - Elaborado por Juan Pablo y Juan Sebastián