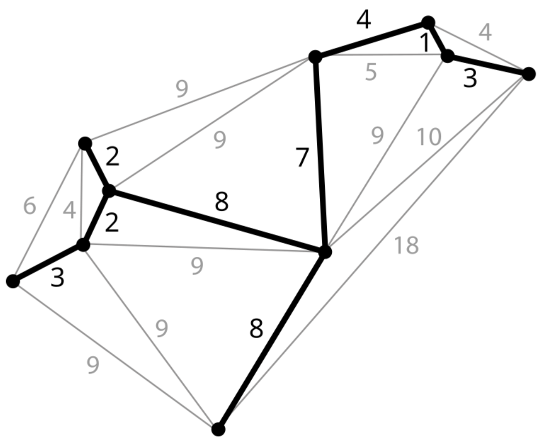
Problema 2:

Demuestre que el minimum-spanning-tree problem y el problema de sitios de hospedaje descritos en clase se pueden modelar como matroides ponderadas (es decir, identifique y describa el conjunto S, la función de pesos y la familia de conjuntos independientes). Demuestre que cumplen con las propiedades de herencia e intercambio.

Minimum-spanning-tree problem



El problema MST consiste en encontrar un subconjunto de aristas que una a todos los vértices, creando un árbol sin ciclos con la condición de que la suma total de los pesos debe de ser mínima. Como se ve en la imagen, tenemos un grafo conexo en el fondo y remarcado está la solución del problema, un árbol sin ciclos con aristas que unen todos los vértices/nodos minimizando la suma total de los pesos. Ahora, viendo el contexto es posible demostrar que este problema se puede modelar como matroides ponderadas.

Componentes de la matroide:

<u>Conjunto base (S)</u>

Este es relativamente sencillo, nuestro conjunto base son las aristas del grafo. Esto lo podemos saber ya que dentro de este problema los elementos que contienen pesos son las aristas que unen los vértices. Entonces **S = {e1,e2, ..., em}**, donde cada e, es una arista que tiene un peso específico.

<u>Familia de conjuntos independientes (I)</u>

Primero hay que definir que es la independencia en el contexto del problema. En este caso un conjunto es independiente si este no forma una estructura cíclica prácticamente, ya que es la restricción principal del problema. Entonces la familia de conjuntos independientes, son todos los subconjuntos de aristas que no forman ciclos. Ósea todos los subárboles del grafo. **$I = \{A \subseteq S \mid A$ no contiene ciclos$\}$**

<u>Función de peso (w)</u>

Como en una matroide lo que buscamos es maximizar, podemos hacer un ajuste con una función para modificar los pesos de las aristas. El problema pide minimizar el costo, entonces lo que podemos hacer nosotros es un proceso inverso. Podemos colocar los pesos negativos y luego maximizar los pesos negativos. Si maximizamos los pesos negativos estaremos minimizando los pesos.

Por ejemplo, si tuviéramos los pesos 9, 5, 4 los pasamos a –9,-5,-4 y el máximo de esos pesos es el –4. La arista con el valor –4 sería elegida como solución óptima de una de los problemas. Podemos escribir la función como: **$w(e_i)=-c(e_i)$**, donde $c(e_i)$ es el costo original de la arista, para cualquier "e" que pertenezca al conjunto de aristas. (prácticamente solo les cambiamos el signo).

Propiedad de Herencia:

La herencia indica que si un conjunto es independiente, todos sus subconjunto también lo son (quitar elementos no rompe la independencia). En este problema decimos que la independencia representa un set de aristas que no generan ciclos. Entonces para demostrar esto, podemos decir que si tenemos un conjunto de aristas no tiene ciclos, cualquier subconjunto de este tampoco tendrá ciclos. Por ejemplo si tenemos las aristas {A-B, B-C}, estas no forman ciclo. Y si vemos los subconjuntos {A-B} y {B-C}, estos tampoco generan ciclos. SI quitamos algún vértice sigue la independencia. Por lo que podemos decir que la herencia se cumple, ya que un conjunto de aristas que no genera ciclos tiene subconjuntos de aristas que tampoco los generan.

Propiedad de intercambio:

En este caso se mira si dos conjuntos independientes A y B, con A más pequeño que B, se puede agregar algún elemento de B a A sin perder independencia. Ósea en este caso involucraría que podemos agregar una arista a otro conjunto independiente y este debe de seguir siendo independiente. Y esto se cumple para el MST también, si tenemos los conjuntos: A = {A-B} (1 arista), B = {B-C, C-D} (2 aristas) se puede agregar B-C o C-D a A sin formar ciclo y sabemos que esto se cumple para cualquier arista y vértice ya que trabajamos con un grafo conexo (debe haber al menos un camino de un vértice a un vértice b).Entonces:

$A,B \in I$ con $|A|<|B|$, existe $e \in B\setminus A$ tal que $A \cup \{e\} \in I$. (Esto se cumple porque B tiene más aristas conectando componentes conexas sin ciclos.

Problema de sitios de hospedaje(Booking)

Problema de *Bookings*

Consideremos un problema más sencillo. Imaginemos una página *web* para encontrar hospedaje, donde los lugares tienen una calificación calculada a partir de retroalimentación de antiguos clientes. La página desea proveer una funcionalidad que permita al usuario determinar un número k y obtener los k lugares de hospedaje cuya suma de calificaciones sea la más alta. En otras palabras, este problema desea el conjunto de k lugares con calificaciones más altas (pues esto resultará en la suma más alta) de entre todos los disponibles.

Este problema es más sencillo. Tenemos lugares de hospedaje y cada uno de estos tiene una calificación/rating. En la web lo que se busca es obtener los lugares de hospedaje mejor calificados de una ciudad, limitados por una cantidad k que elige el usuario visitando la web.

Componentes de la matroide:

<u>Conjunto base (S)</u>

En este caso el conjunto con todos los elementos será el conjunto de lugares de hospedaje (digámosle h). Se sabe esto porque los elementos con el peso son los lugares de hospedaje, el peso podría ser representado con la calificación (digámosle r de rating). Entonces tenemos **$S=\{h_1, h_2, \ldots, h_n\}$**, donde h_i tiene calificación r_i

<u>Familia de conjuntos independientes (I)</u>

La independencia en este caso se puede modelar con la restricción de k. Un conjunto será independiente si la cantidad de h (lugares de hospedaje) es menor a la cantidad de k ingresada por el usuario. Si el usuario pide 5 hospedajes, los grupos independientes serán todos aquellos subconjuntos con 5 o menos lugares. Esto se puede escribir como: **$I=\{ A \subseteq S \mid |A| \leq k\}$**.

<u>Función de peso (w)</u>

Esta es bastante sencilla, ya que se busca maximizar los pesos en una matroide, en este caso no hay que hacer mayor cambio a los datos que tenemos. Los pesos de los elementos estarán dados por su calificación, la cuál previamente llamamos r. Entonces tenemos: **$w(h_i)=r_i$, (el peso de un hospedaje h es dado por su rating r)**

Propiedad de Herencia:

Nuevamente aquí lo que se busca es que, si un conjunto es independiente, sus subconjuntos también lo sean. En este contexto la independencia es ser menor o igual que K. Esto se cumple y es bastante obvio. Por ejemplo si el usuario pide k=5 hospedajes, un conjunto de 4 hospedajes (A) cumple con la independencia. Entonces si vemos todos los subconjuntos de (A), estos serán de 4 o menos elementos. Y por lo tanto son independientes también ya que los subconjuntos serán menores a K=5. Esto se puede escribir como: Si **B ∈ I (|B| ≤ k) y A ⊆ B**, entonces **|A|≤k**, por lo tanto, **A∈I.**

Propiedad de intercambio:

Ahora aquí nuevamente lo que buscamos ver es si teniendo 2 conjuntos que cumplan la independencia, de tal forma que un conjunto A se menor al conjunto B. Esto también se cumple. Si tenemos los conjuntos:

K = 3

A = {h1,h2}

B = {h4,h5,h6}

Ambos cumplen con la condición de ser menor o igual a K en tamaño, por lo que son independientes. Y si tomamos cualquier elemento de B y lo colocamos en A, por ejemplo tomamos h5, tendríamos A = {h1,h2, h5} y esto sigue siendo independiente al ser menor o igual que K=3. Entonces podemos decir que dados 2 conjuntos A y B que sean independientes (menores o iguales K), si A es menor que B en tamaño y tomamos un valor de B y lo colocamos en A, A seguirá siendo independiente (seguirá siendo menor o igual a K):

Dados **A,B ∈ I** con **|A|<|B|,** existe **h∈B\A tal que A∪{h}∈I**.

Ya que **|B|≤k y |A|<|B|, |A∪{h}|≤k.**