Segundo Avance del Proyecto Final

Estructuras Computacionales Avanzadas

Asignatura:

"Estructuras Computacionales Avanzadas"

Maestro:

Miguel Ángel Meza de Luna

Alumnos:

- Luis Pablo Esparza Terrones
- Luis Manuel Flores Jiménez
- Juan Francisco Gallo Ramírez

Ingeniería en Computación Inteligente

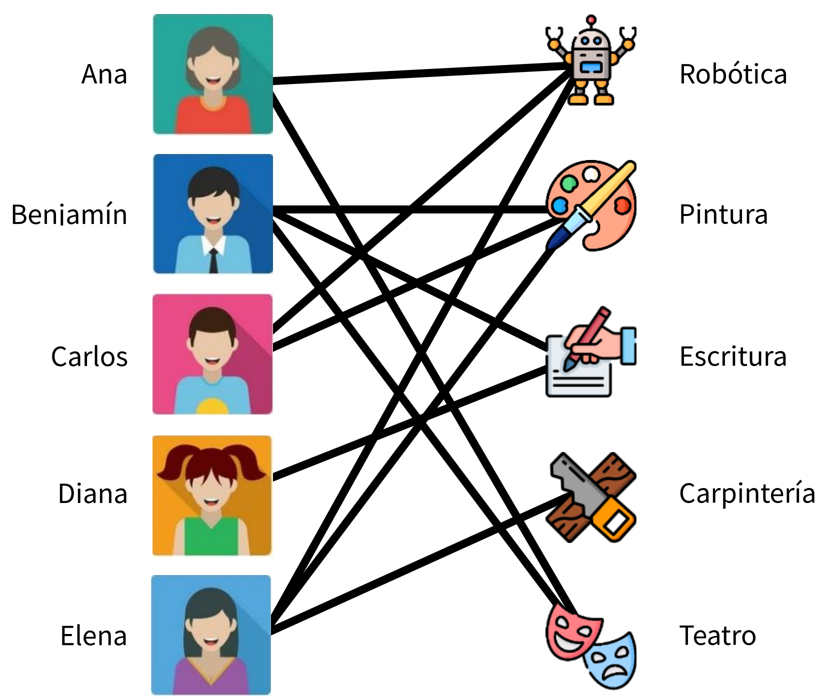
3er Semestre

Unidad III

Problemas de Pareo de grafos

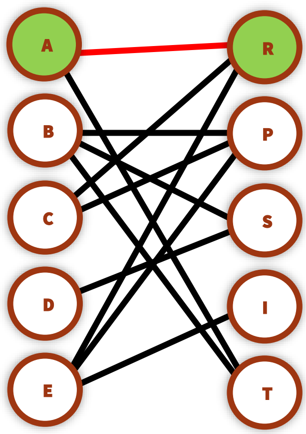
Problema #1

En la asignación de talleres de una escuela secundaria se busca que 5 alumnos recién ingresados cursen el taller de su preferencia, para ello se les pidió llenar un formulario en el cual seleccionaran a lo más 3 talleres que sean de su agrado. Las elecciones realizadas representadas gráficamente son las siguientes:

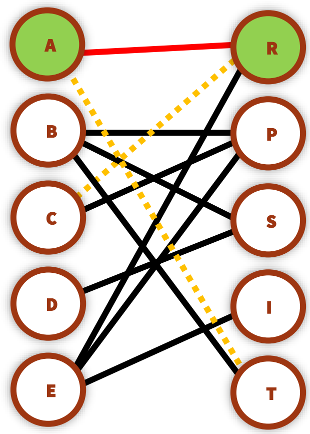


Se busca que todos estén en un taller de preferencia pero que solo ingrese una persona a cada taller.

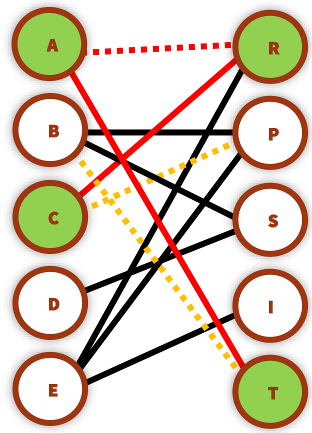
Problema #1 | Solución



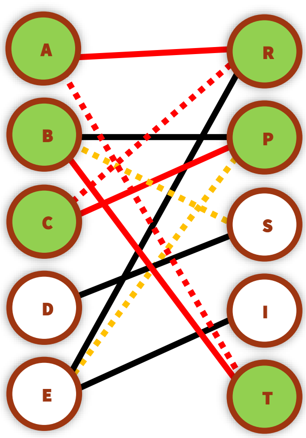
Una vez representado el problema en un grafo, seleccionamos primeramente una asignación, se muestra con **línea roja** continua una asignación, y con **relleno verde** los nodos emparejados.



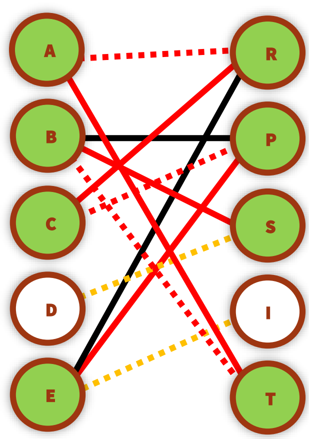
Después se busca un camino M-incrementable, es decir, que los extremos del camino no se encuentren saturados, estos extremos se representan con una **línea naranja** discontinua.



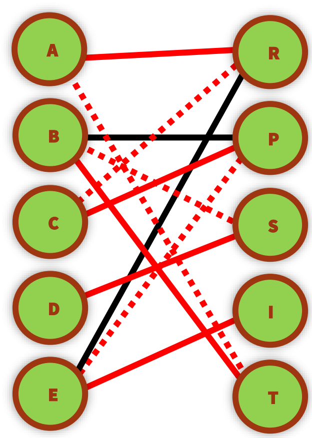
Se procede a alternar este camino de tal forma que se emparejen más nodos.



Se repiten los pasos anteriores hasta obtener un emparejamiento perfecto si existe.



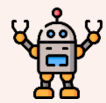
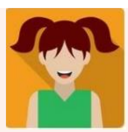
Se repiten los pasos anteriores hasta obtener un emparejamiento perfecto si existe.



Finalmente, todos los nodos quedaron emparejados.

Problema #1 | Resultado

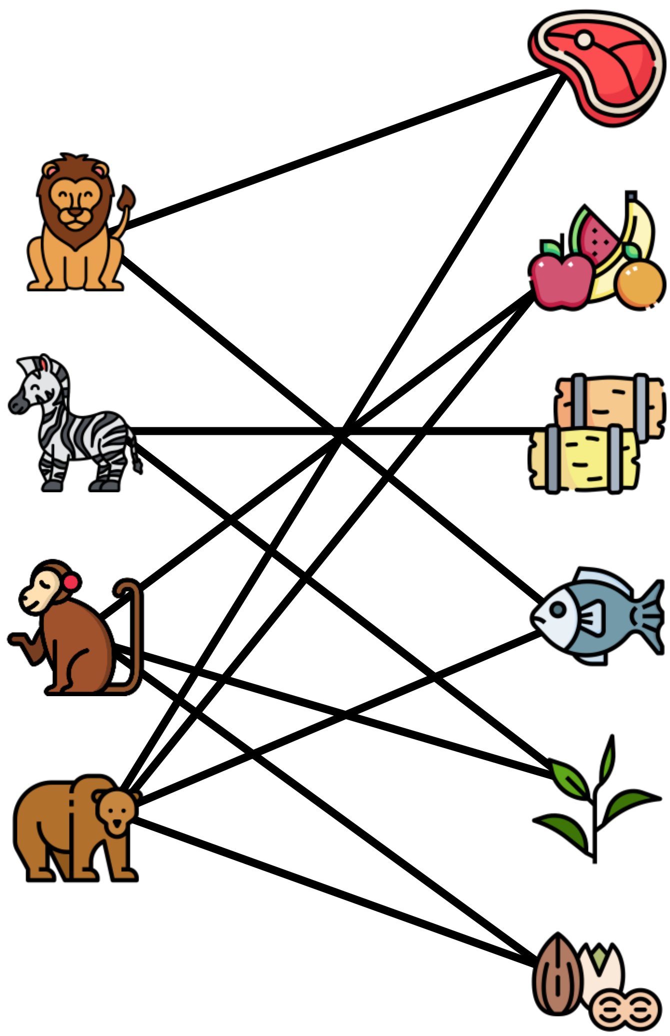
Así pues, con el procedimiento anteriormente desarrollado, el resultado del emparejamiento es un emparejamiento perfecto, ya que todos sus nodos se encuentran emparejados:

Ana		**Robótica**	
Carlos		**Teatro**	
Benjamín		**Pintura**	
Diana		**Escritura**	
Elena		**Carpintería**	

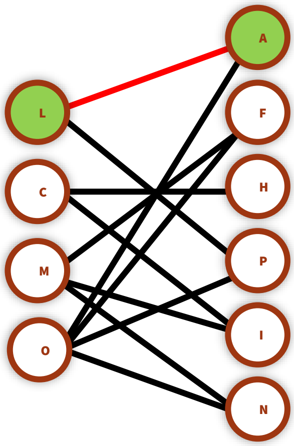
Problemas de Pareo de grafos

Problema #2

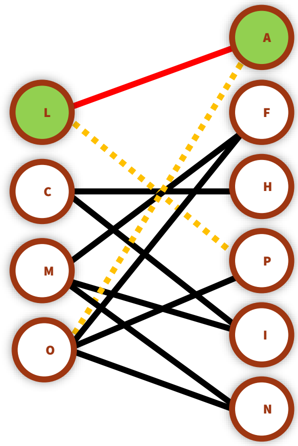
En una sección de un zoológico se tienen 4 animales, un león, una cebra, un mono y un oso. Los cuidadores cuentan con carne fresca, frutas, heno, pescado, hierva y nueces. Solo se busca dar un solo un tipo de alimento a los animales. Así pues, la representación gráfica de lo que puede comer cada animal es la siguiente:



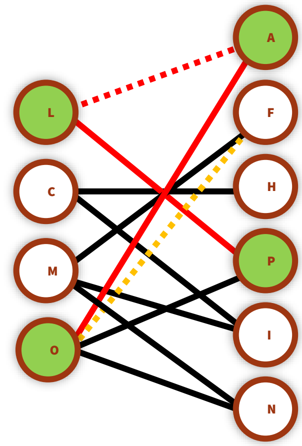
Problema #2 | Solución



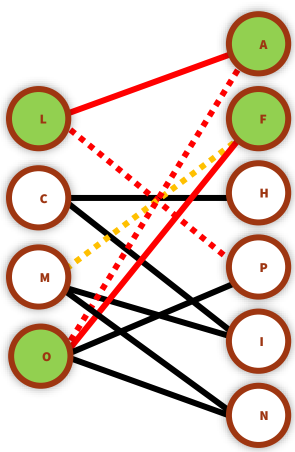
Una vez representado el problema en un grafo, seleccionamos primeramente una asignación, se muestra con **línea roja** continua una asignación, y con **relleno verde** los nodos emparejados.



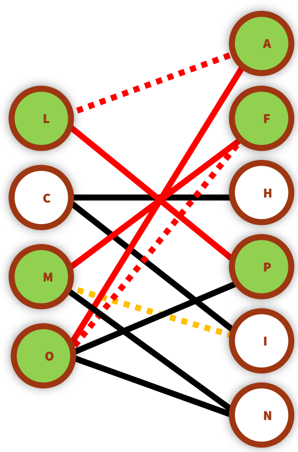
Después se busca un camino M-incrementable, es decir, que los extremos del camino no se encuentren saturados, estos extremos se representan con una **línea naranja** discontinua.



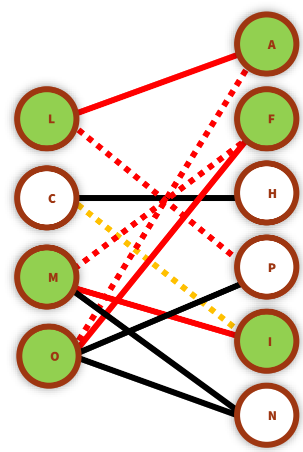
Se procede a alternar este camino de tal forma que se emparejen más nodos.



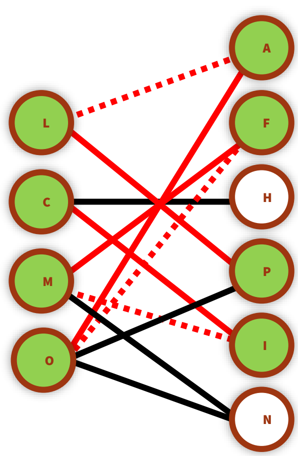
Se repiten los pasos anteriores hasta obtener un emparejamiento perfecto si existe.



Se repiten los pasos anteriores hasta obtener un emparejamiento perfecto si existe.



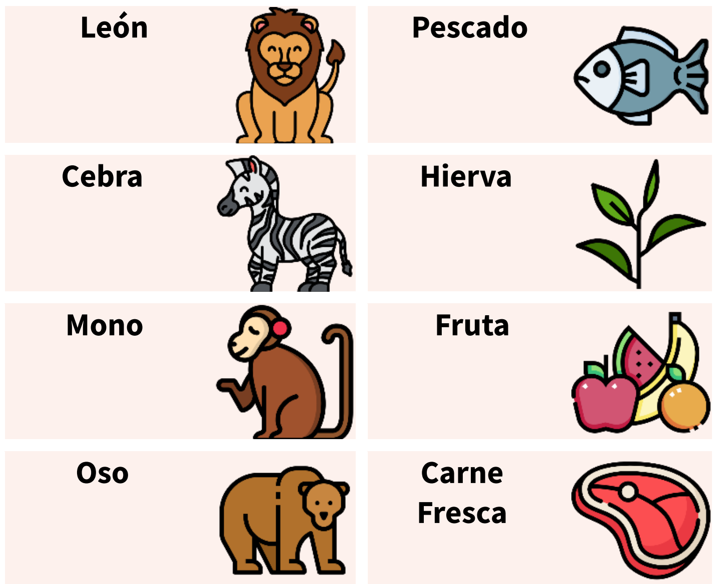
Se repiten los pasos anteriores hasta obtener un emparejamiento perfecto si existe.



Finalmente, todos los nodos quedaron emparejados.

Problema #2 | Resultado

Así pues, con el procedimiento anteriormente desarrollado, el resultado del emparejamiento es un emparejamiento máximo, sin embargo, no perfecto:



Pareos Maximales Mínimos

Conceptos

- *Emparejamiento Maximal:*

Un emparejamiento se dice que es maximal si no está contenido en otro emparejamiento. Un emparejamiento maximal es un conjunto de aristas en un grafo que no se puede extender añadiendo más aristas sin violar la condición de emparejamiento, es decir, sin compartir nodos finales entre las aristas. Un emparejamiento maximal no necesariamente cubre todos los nodos del grafo. Puede haber varios emparejamientos maximales en un grafo. Un emparejamiento maximal es un conjunto de aristas que no se puede ampliar sin romper la propiedad de emparejamiento. Puede que no sea el emparejamiento que cubre la mayor cantidad de aristas posibles en el grafo.

- *Emparejamiento Maximal mínimo:*

Un emparejamiento maximal mínimo, según la interpretación que se podría dar a esa frase, sería un conjunto maximal de aristas con la menor cantidad posible de aristas en comparación con otros conjuntos maximales en el mismo grafo.

Comparación DFS y BFS con los algoritmos de grafos dirigidos y grafos no dirigidos.

Comparación del algoritmo

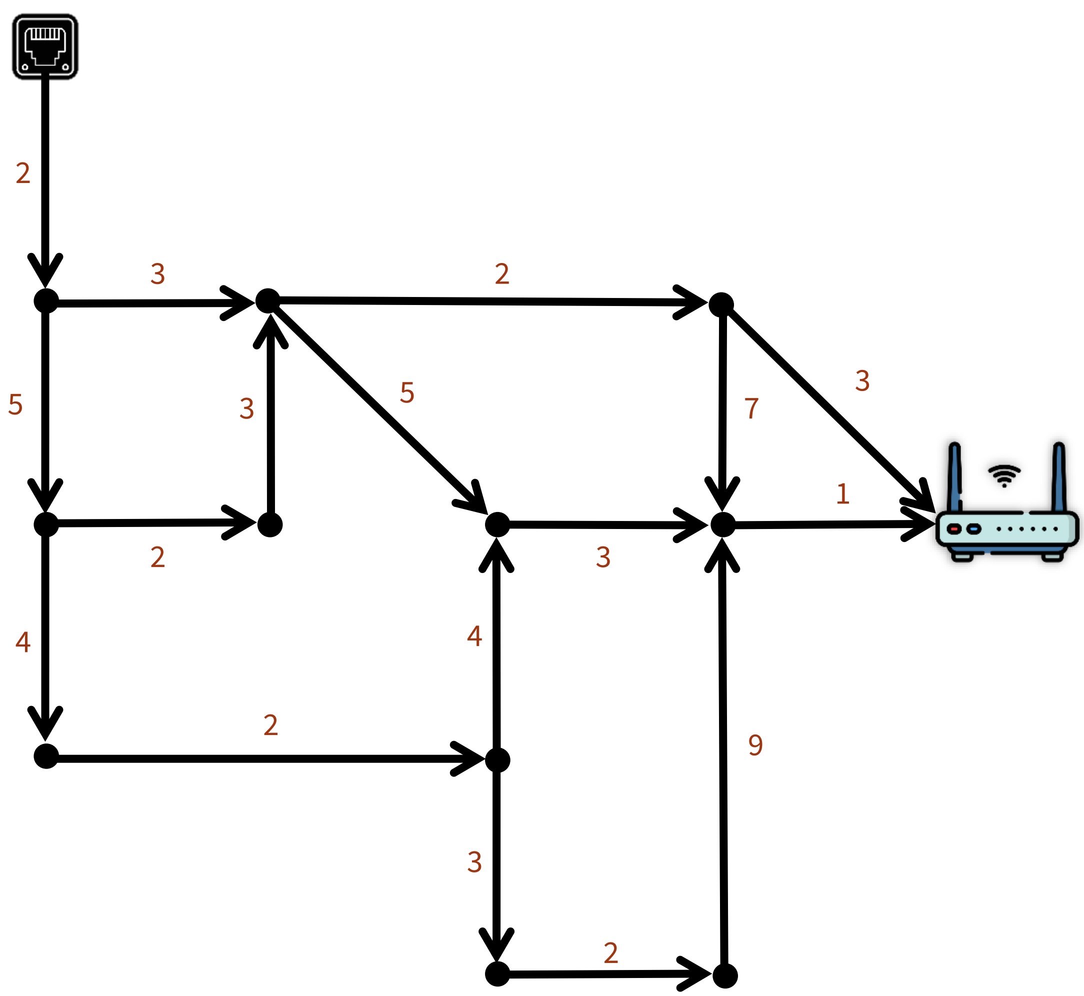
Los algoritmos de Búsqueda en Profundidad (DFS) y Búsqueda en Amplitud (BFS) son esencialmente los mismos, independientemente de si se aplican a un grafo dirigido o no dirigido. Ambos algoritmos se utilizan para recorrer o buscar en grafos, y la principal diferencia en la implementación radica en cómo manejan las aristas.

Grafo No Dirigido	Grafo Dirigido
En un grafo no dirigido, DFS y BFS pueden recorrer cualquier arista sin restricciones particulares. La principal consideración es evitar bucles infinitos y garantizar que cada nodo se visite una vez. No hay distinción entre aristas de entrada y salida en un grafo no dirigidos.	En un grafo dirigido, tanto DFS como BFS siguen las aristas dirigidas, respetando la dirección especificada por las flechas. Sin embargo, en DFS, si se alcanza un nodo desde varias ramas, puede visitarse más de una vez, mientras que en BFS, cada nodo se visita solo una vez, ya que se procesan todos los vecinos inmediatos antes de pasar a los siguientes niveles.
DFS	*BFS*
▪ DFS utiliza una estrategia de exploración en profundidad. Se adentra tanto como sea posible a lo largo de cada rama antes de retroceder. ▪ Generalmente se implementa utilizando una pila, ya sea de manera recursiva o mediante una pila explícita. ▪ Tiende a requerir menos memoria en comparación con BFS, ya que solo necesita almacenar la información de un camino a la vez.	▪ BFS utiliza una estrategia de exploración en amplitud. Explora todos los vecinos de un nodo antes de pasar a los nodos más distantes. ▪ Se implementa comúnmente utilizando una cola. El orden de llegada a la cola determina el orden de exploración. ▪ Tiende a requerir más memoria que DFS, ya que debe mantener información sobre todos los nodos en el nivel actual antes de pasar al siguiente nivel.

Problemas de Grafos Dirigidos

Problema #1

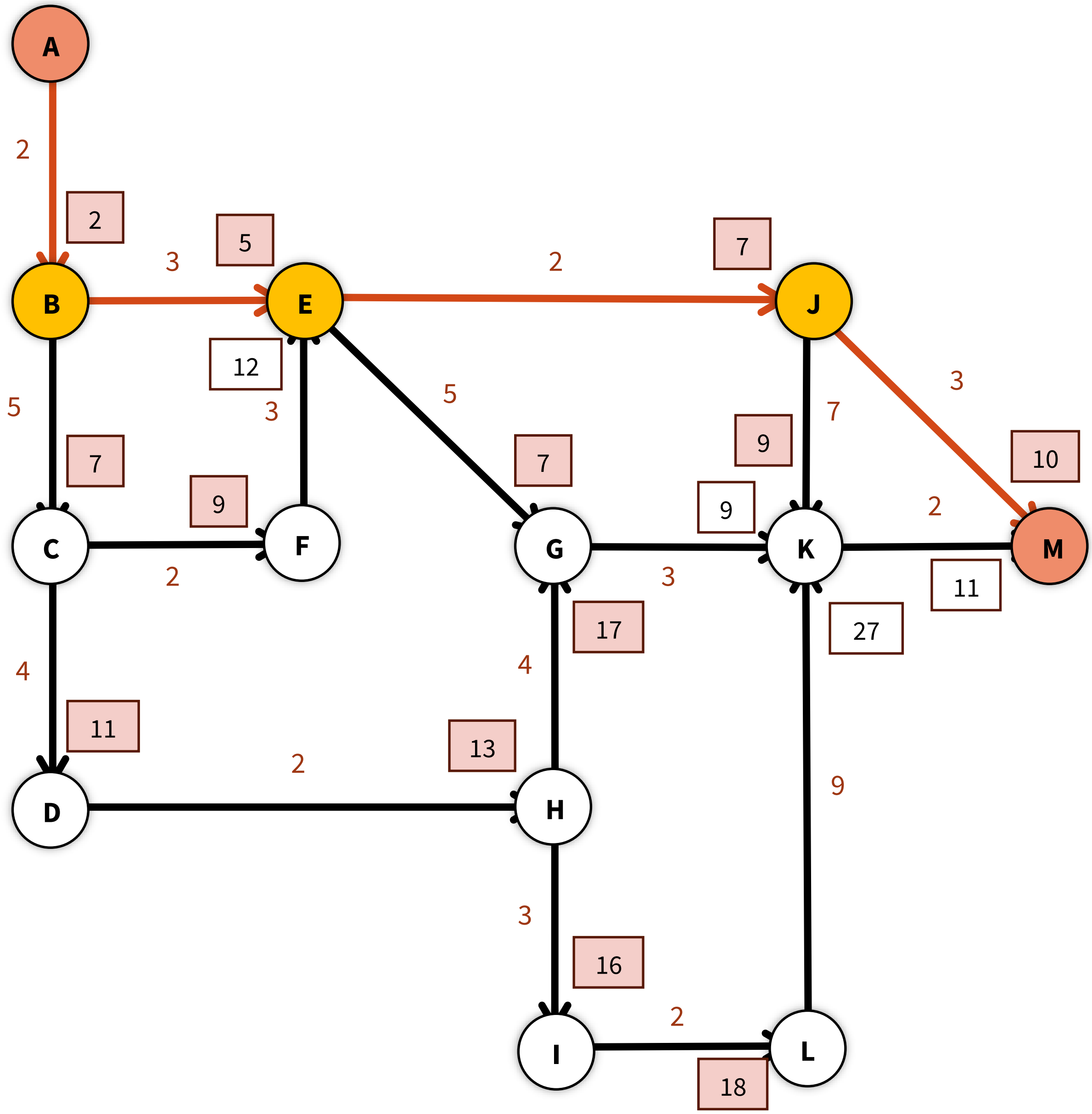
En una estructura se tienen conductos eléctricos vacíos donde se planea introducir un cable ethernet de una estancia a otra para conectar un repetidor de señal, el plano de los conductos, sus distancias y los puntos a donde se quiere llegar son los siguientes:



Se busca gastar la mínima cantidad de cable posible para ahorrar costos. ¿Por cuales puntos debería pasar el cable?

Problema #1 | Solución

El procedimiento que se puede usar para resolver el problema es el algoritmo de Dijkstra, a continuación, desarrollamos el ejercicio.



Problema #1 | Respuesta

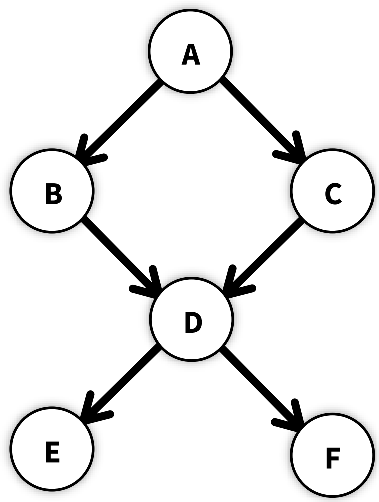
El cable debería hacer la ruta: **A - B - E - J – M**

La duración de la ruta es de 10.

Problemas de Grafos Dirigidos

Problema #2

Considera que un grafo dirigido representa la infraestructura de drenaje en un área geográfica. Cada nodo en el grafo representa una ubicación o punto específico, y las aristas dirigidas indican la dirección del flujo del agua entre estas ubicaciones. Verifica si el flujo de agua no pasa más de una vez en un nodo, es decir, verifica si hay ciclos. El grafo correspondiente es el siguiente:



Problema #2 | Solución

Para saber si existen ciclos en un grafo dirigido, se puede usar una búsqueda en profundidad (DFS), ya que, si al recorrer los nodos encontramos uno que ya tenemos en nuestra lista de visitados, y además este a su vez contiene otro en la lista, sabremos que existe un ciclo.

Pila	Vértices Procesados
A	
B C	A
B D	C
B E F	D
B E	F
B	E
Pila vacía	B

Problema #2 | Respuesta

No hay ciclos, ya que al colocar los nodos adyacentes de todos los nodos, estos no contienen algún nodo de los ya visitados que también contengan algún nodo adyacente en la lista

Fotografía de entendimiento

	Grafo en matriz y lista de adyacencia	Búsqueda en anchura	Búsqueda en profundidad	Grafo conexo	Camino más corto	Pareo grafo bipartito	Pareo grafo normal
Juan Gallo	10	10	10	10	9	9	9
Luis Flores	10	10	10	10	9	8	8
Pablo Esparza	10	10	10	10	9	8	8

1	2	3	4	5	6	7	8	9	10

No se podría realizar

Se puede realizar con ayuda

Se puede realizar sin ayuda

Referencias

OpenAI. (s.f.). *ChatGPT*. Obtenido de https://chat.openai.com/

Santamaría, M. P. (Octubre de 2020). *EMPAREJAMIENTOS ESTABLES.* Obtenido de https://repositorio.unican.es/xmlui/bitstream/handle/10902/20544/Santamar%C3%ADa%20Manteca%20Paula.pdf?sequence=1&isAllowed=y