PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
Dpto. de Ingeniería de Transporte y Logística

Curso: ICT2233
Semestre: 01'2022
Profesor: Mathias Klapp

EXAMEN
FLUJO EN REDES - ICT2233

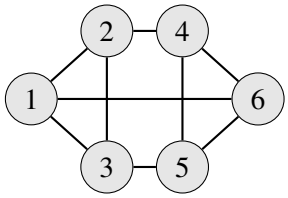
TIEMPO: 8:40 A 11:10

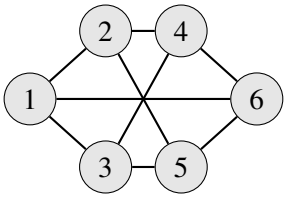
- *La prueba tiene 9 preguntas de diferente puntaje que suman 64 puntos. El 7 es con 60 puntos.*
- *¡Coloque nombre en el cuadernillo y no lo separe!* ***Se penalizará con 4 puntos a quien no siga esta instrucción****.*
- *La prueba es sin apuntes y/o equipos electrónicos. Se permite formulario de dos hojas escritas a mano.*
- *Dos rondas de preguntas al ayudante (al inicio y cuando la prueba lleve un 50% de avance). No hacer preguntas en otro momento por favor para no desconcentrar a sus pares.*

Éxito

Pregunta 1 (4 puntos):

Considere un grafo conexo y no dirigido $G := (N, A)$, donde n y m son su cantidad de nodos y arcos, respectivamente. Un grafo como el supuesto se denomina ***k*-regular** si cada uno de sus nodos posee grado k. Por ejemplo, a continuación se ilustran dos grafos 3-regulares de $n = 6$ nodos.





(a) (2p) ¿Qué debiese cumplir el valor k de un grafo k-regular para que este grafo posea un Tour Euleriano?

Respuesta: Trivial. k debe ser par.

(b) (2p) Muestre que, para un grafo k-regular, los valores de k y/o n deben ser par.

Respuesta: El número de arcos es $n \cdot k/2$ y debe ser entero, por lo tanto $n \cdot k$ debe ser par.

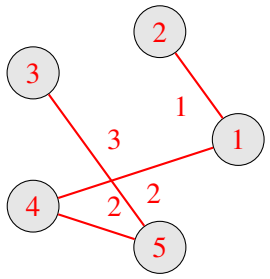
Pregunta 2 (8 puntos):

Responda las preguntas a continuación asociadas a un grafo completo y no dirigido $G = (N, A)$ de 5 nodos cuyos costos en arcos se detallan en la siguiente matriz de adyacencia nodo-nodo:

$c_{\{i,j\}}$	1	2	3	4	5
1		1	5	2	4
2	1		4	4	5
3	5	4		5	3
4	2	4	5		2
5	4	5	3	2	

(a) (3p) Aplique el algoritmo de Prim y obtenga un AMEM sobre el grafo anterior. Indique su costo, el conjunto de arcos seleccionados en el AMEM y su orden de selección según el algoritmo de Prim.

Respuesta:



El orden de inclusión de arcos si sigue el algoritmo de PRIM es: $\{1,2\};\{1,4\};\{4,5\};\{3,5\}$ y el costo del AMEM es 8

(b) En aplicaciones, como en el diseño de redes de distribución de carga, a veces es necesario restringir que algunos nodos $S \subset N$ sean terminales (*i.e.*, nodos "hoja") del árbol a diseñar.

(I) (3p) Diseñe un algoritmo computacionalmente eficiente que identifique un árbol de envergadura máxima a mínimo costo sujeto a que todos los nodos en un conjunto $S \subset N$ deben ser hojas del árbol. ¿Qué complejidad computacional posee su algoritmo?

Respuesta: La forma más sencilla es la siguiente:

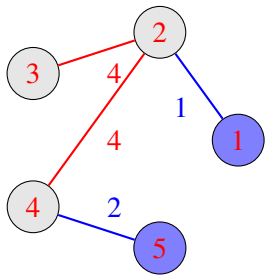
- ◇ calcular el AMEM sobre $N \setminus S$ y agregar todos los arcos de esta estructura al conjunto T
- ◇ Luego, para cada nodo $i \in S$:
  - Conectar i a la componente conectada $N \setminus S$ a través del arco a^* más barato, es decir, $a^* = \text{argmín}_{a \in \delta_i \cap \delta_{N \setminus S}} c_a$
  - $T \leftarrow T \cup \{a^*\}$
- ◇ Retornar a^*

El algoritmo anterior posee un orden de complejidad similar al AMEM más $O(\sum_{i \in S} g_i) \approx O(m)$ en calcular las conexiones restantes (este último es dominado por el cálculo del AMEM).

(II) (2p) Identifique un árbol de envergadura máxima a mínimo costo sujeto a que los nodos 1 y 5 sean hojas de árbol. ¿Cuánto cuesta imponer esta restricción sobre el costo del AMEM irrestricto?

Respuesta:

Calculamos el AMEM sobre $N \setminus \{1,5\}$ y luego conectamos los nodo 1 y 5 a costo mínimo a la componente $\{2,3,4\}$. El resultado posee costo 11 (3 más que el AMEM irrestricto) y el árbol es el siguiente:



Pregunta 3 (8 puntos):

Suponga que usted administra la venta de pasajes para una empresa que opera un solo tren con capacidad B pasajeros. El tren ejecuta un solo recorrido diario que visita a un conjunto $N := \{1,2,\ldots,n\}$ de ciudades en ese orden, es decir, inicia el recorrido en la ciudad 1, luego visita la 2, después la 3 y así sucesivamente hasta terminar el recorrido en la ciudad n.

Un estudio de demanda de transporte ha identificado un conjunto $P \subseteq \{(i,j) : i,j \in N : j > i\}$ de pares origen-destino de ciudades en los cuales existe una demanda por viajes. Específicamente, se estima que a una tarifa de $\$f_{i,j}$ pesos por pasaje habrá una demanda diaria de $b_{i,j} \in \mathbb{N}$ viajes desde i hacia j en cada par $(i,j) \in P$.

La empresa busca determinar el número de pasajes diarios $n_{i,j} \leq b_{i,j}$ a vender para cada par $(i,j) \in P$ que maximice ventas sin exceder la capacidad del tren.

(a) (6p) Formule un Problema de Flujo a Costo Mínimo (PFCM) que identifique decisiones de venta diaria de pasajes que maximizan venta. Describa el grafo asociado al PFCM y defina bien sus nodos y arcos. Luego determine valores de costo y capacidad en arcos y oferta en nodos.

AYUDA: piense en un grafo en donde fluyen pasajeros.

Respuesta: Nota de corrección: 75 % de puntaje en determinar correctamente el grafo (nodos y arcos).

Existen diversas formas de modelar esta situación, por lo que no hay una sola respuesta. A continuación se presenta una forma que requiere de un PFCM capacitado en arcos.

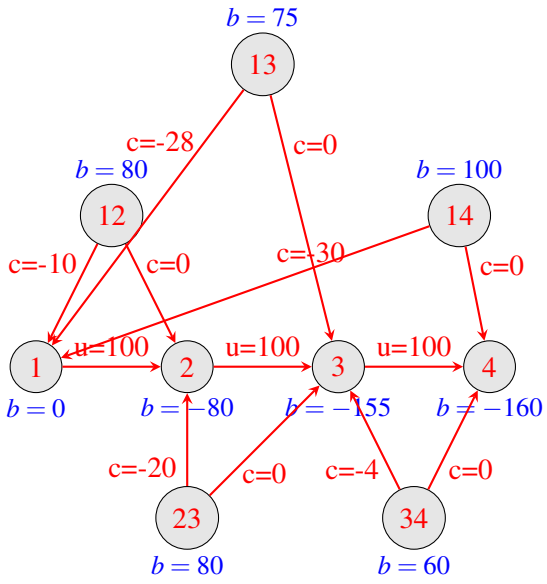
- ◇ Nodos: Definiremos un conjunto $V := N \cup P$ de nodos con dos tipos:
  - Nodos $i \in N$ que representan ciudades.
  - Nodos $(i,j) \in P$ que representan pared o-d con demanda.
- ◇ Ofertas netas:
  - Cada nodo par o-d $(i,j) \in P$ ofrece $b_{i,j}$ viajes a la red.
  - Cada nodo ciudad $c \in N$ demanda todos los viajes cuyo destino es i, es decir, posee oferta neta igual a $b'_i = -\sum_{(i,j)\in P} b_{i,j}$.
- ◇ Arcos: Definiremos un conjunto A con tres tipos de arco:
  - Arcos $(i, i+1)$ de tren con capacidad B y costo 0 para cada $i \in N \setminus \{n\}$. Estos arcos representan el flujo de pasajeros que transporta el tren.
  - Arcos $((i,j),i)$ de demanda cubierta con capacidad infinita y costo $-f_{i,j}$ para cada $(i,j) \in P$. Estos arcos representan el flujo de viajes ofrecidos para la demanda $(i,j) \in P$.
  - Arcos $((i,j),j)$ de demanda insatisfecha con capacidad infinita y costo nulo para cada $(i,j) \in P$. Estos arcos representan el flujo de viajes ofrecidos para la demanda $(i,j) \in P$.

Este grafo se vería más o menos así:

(b) (2p) Ilustre el grafo asociado al problema formulado y modelado anteriormente para $n = 4$, $B = 100$, $P = \{(1,2),(1,3),(1,4),(2,3),(2,4),(3,4)\}$ y los siguientes datos de tarifa y demanda por par origen-destino:

(i,j)	$(1,2)$	$(1,3)$	$(1,4)$	$(2,3)$	$(3,4)$
$b_{i,j}$	80	75	100	80	60
$t_{i,j}$	10	28	30	20	4

Respuesta:



Pregunta 4 (6 puntos):

Suponga que usted desea identificar una ruta de mínimo costo desde un nodo s hacia un nodo t sobre un grafo dirigido $G = (N,A)$ que posee costos no-negativos y ciclos dirigidos.

El costo de atravesar arcos que emanan desde el nodo s es $\$c_a$ para cada $a \in \delta_s^+$. Sin embargo, el costo de atravesar un arco $a \in A \setminus \delta_s^+$ es igual a $c_{a,a'}$, donde a' es el arco visitado inmediatamente antes de a en la ruta. En otras palabras, el costo de visitar un arco cambia en función del arco visitado inmediatamente antes. Este tipo de modelo permite representar situaciones más complejas como, por ejemplo, demoras generadas por virajes a la izquierda o el costo extra por acelerar desde una calle pequeña hacia una autopista.

¿Cómo resolvería este problema mediante un algoritmo computacionalmente eficiente? Explique.

Respuesta: Una forma es expandir la red a una con el siguiente conjunto N' de nodos:

- el nodo s
- un nodo (i,j) para cada par $i \in N \setminus \{s\}$ y $j \in N \setminus \{t\}$ que representa estar en i habiendo visitado antes a j.

Luego, el conjunto A' de arcos sería:

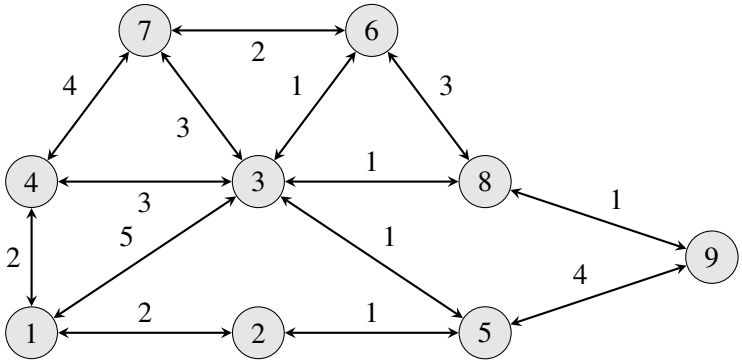
- un arco a para cada $a \in \delta_s^+$ a costo $c'_a = c_a$
- un arco $a' = ((i,j),(k,i))$ a costo $c'_a = c_{(i,k),(j,i)}$ para cada par $(i,j),(k,i) \in N'$

Finalmente se debe ejecutar Dijkstra desde s hacia adelante y la ruta mínima a nodo t original sería la ruta mínima de menor costo entre las que van a los nodos (t, i) con $i \in N \setminus \{t\}$.

El algoritmo posee complejidad computacional igual al algoritmo de Dijkstra pero sobre $O(n^2)$ nodos (con esto basta!). Si se implementa con Heap de Fibonacci sería $O(n^2 \log(n))$.

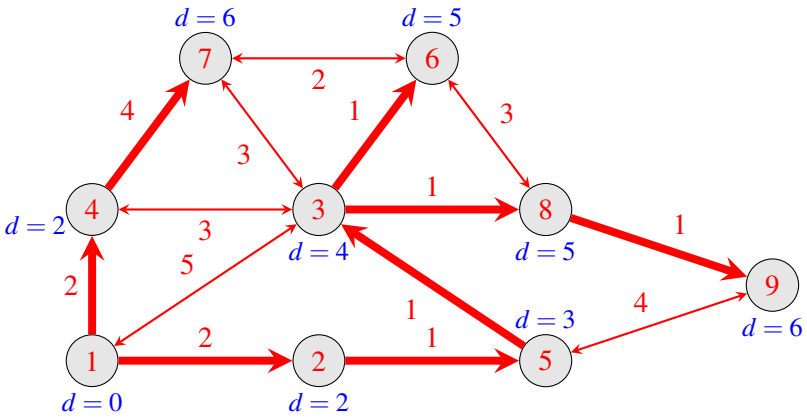
Pregunta 5 (6 puntos):

Considere el siguiente grafo de nueve nodos en donde se desea obtener la ruta de menor costo desde el nodo 1 hacia cada nodo restante. En cada arco del grafo se detalla su costo y este puede utilizarse en ambas direcciones.



(a) (4p) Ejecute el algoritmo de Dijkstra desde 1 hacia adelante y calcule los costos de ruta mínima a cada nodo. Indique el orden de cierre de los nodos.

Respuesta: La ruta mínima se marca en la siguiente figura:



El orden de cierre es: $1,2$ o $4,5,3,8$ o $6,7$ o 9

(b) (2p) Si se agrega un arco bidirigido entre los nodos 2 y 3. ¿En qué rango de valores debe estar el costo de este arco para ser utilizado en un árbol de rutas a mínimo costo que emana desde 1?

Respuesta: Debiese costar menos o igual que $d_3 - d_2 = 2$

Pregunta 6 (6 puntos):

Considere el problema de identificar la mayor cantidad de rutas disjuntas en arcos desde un nodo s hacia otro nodo t sobre un grafo dirigido $G = (N,A)$. Responda:

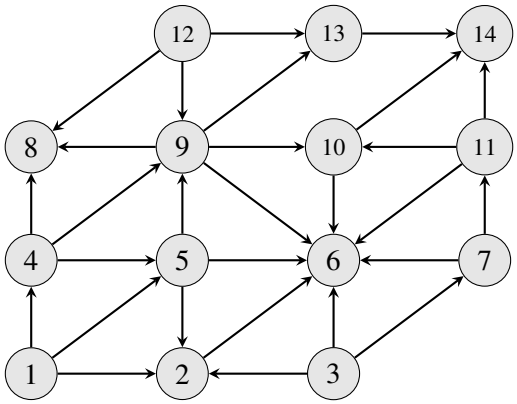
(a) (1p) Argumente que un nodo $i \in N \setminus \{s,t\}$ tal que $g_i^+ = 0$ se puede borrar (junto con todos sus arcos que tocan a este nodo) manteniendo inalterada la cantidad máxima de rutas disjuntas en G.

Respuesta: El valor del flujo máximo es igual a la mínima capacidad de un corte $s-t$ y un nodo sin arcos salientes nunca tendrá arcos salientes de un corte.

(b) (1p) Argumente que un nodo $i \in N \setminus \{s,t\}$ tal que $g_i^- = 0$ se puede borrar (junto con todos los arcos que tocan a este nodo) manteniendo inalterada la cantidad máxima de rutas disjuntas en G.

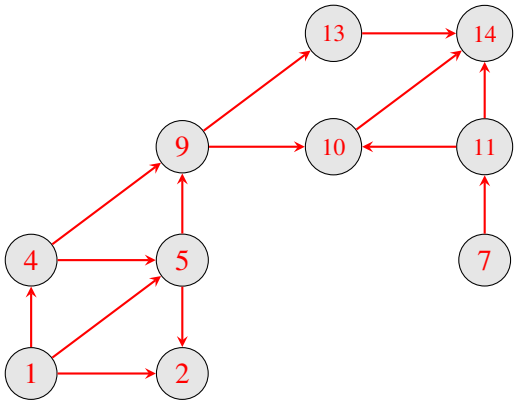
Respuesta: El valor del flujo máximo es igual a la mínima capacidad de un corte $s-t$ y un nodo sin arcos de entrada nunca tendrá flujo que pueda hacer fluir hacia el destino a través del nodo. Por ello se podría eliminar de un corte y reducir su capacidad. Por ende, nunca estará en el corte de menor capacidad.

(c) (3p) ¿Cuántas rutas disjuntas **en arcos** posee el siguiente grafo dirigido desde 1 a 14? Justifique.

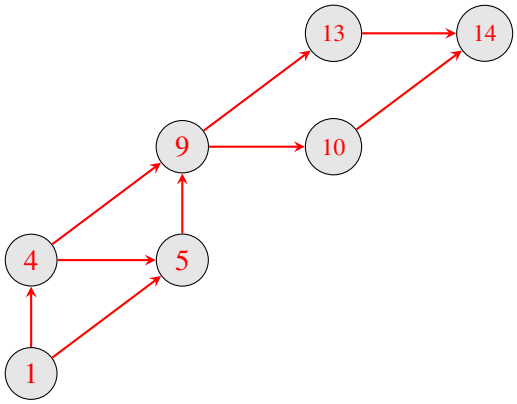


Respuesta: Viendo el grafo, inmediatamente se puede eliminar los nodos 3, 8, 12 y 6 (y todos sus arcos adyacentes) pues tienen grado de entrada o salida 0.

El grafo resultante es:



De este nuevo grafo resulta evidente que 2, 7 y 11 pueden ser eliminados sin perder rutas. El grafo resultante es:



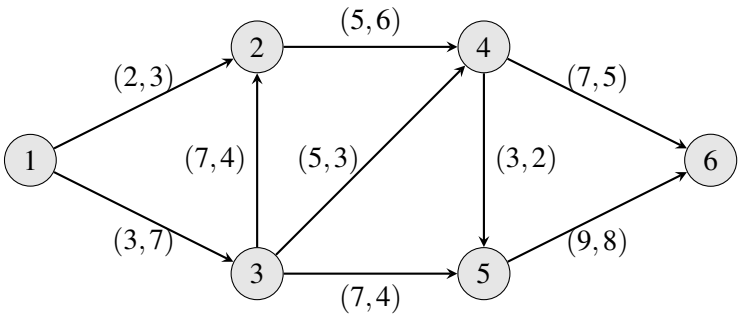
En este grafo el número de rutas disjuntas en arcos es evidente e igual a 2 (min corte es $X = 1$ desde el cual emanan dos arcos).

(d) (1p) ¿Cuántas rutas disjuntas **en nodos** posee el mismo grafo desde 1 a 14? Justifique.

Respuesta: Después del analisis anterior, es evidente que el número de rutas dijuntas en nodos es 1 (se saca el nodo 9 y se desconecta todo)

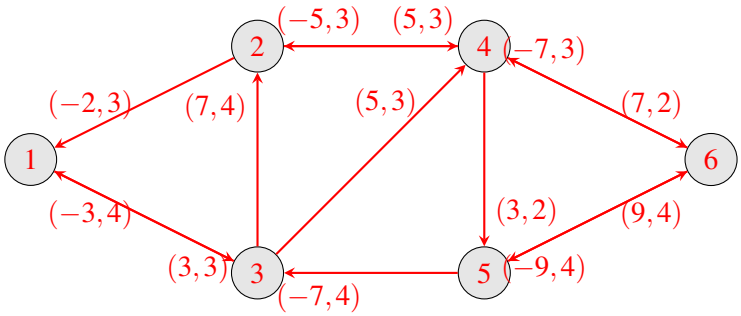
Pregunta 7 (11 puntos):

Considere el siguiente grafo dirigido donde cada arco $a \in A$ detalla un vector (c_a, u_a) con información de costo por unidad de flujo c_a y capacidad máxima de flujo u_a. Suponga que usted desea transportar flujo desde el nodo 1 hacia el nodo 6.

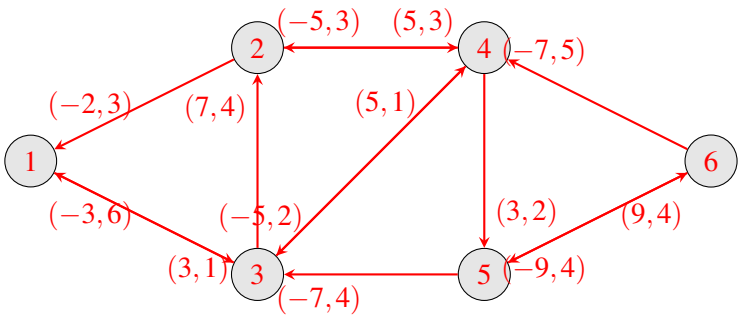


(a) (4p) Identifique, aplicando un algoritmo válido estudiado en el curso, el máximo valor de flujo F^* transportable desde 1 hacia 6. Certifique su solución identificando flujos en arcos que logren ese valor y un corte con esa capacidad.

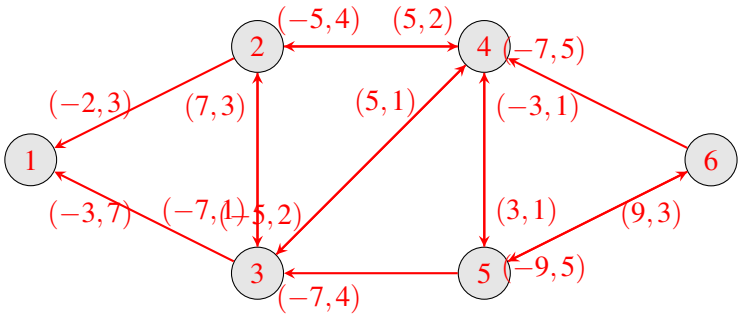
Respuesta: Aplicaremos el algoritmo de ruta aumentable (alternativamente se podría ejecutar el algoritmo Push/Relabel). llevaremos costos residuales para ayudarnos a responder la siguiente pregunta (pero en esta parte no son necesarios). Comenzaremos aumentando 4 unidades por la ruta $1-3-5-6$ y otras 3 más por la ruta $1-2-4-6$:



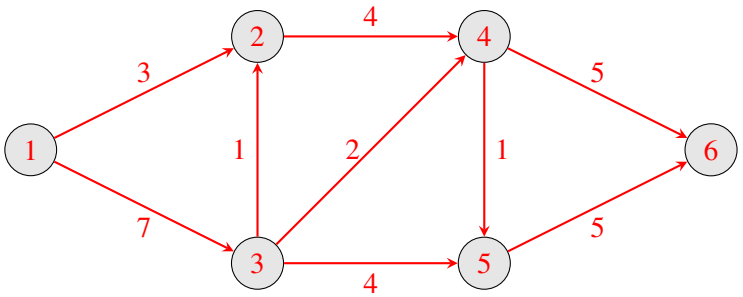
Luego, aumentaremos 2 unidades más por 1-3-4-6:



Finalmente, aumentaremos 1 unidad más por 1-3-2-4-5-6:



Por lo tanto, el mayor valor de flujo es $F^* = 10$, el costo que lo certifica es $X = \{1\}$ y un vector de flujo que lo podría llevar es:

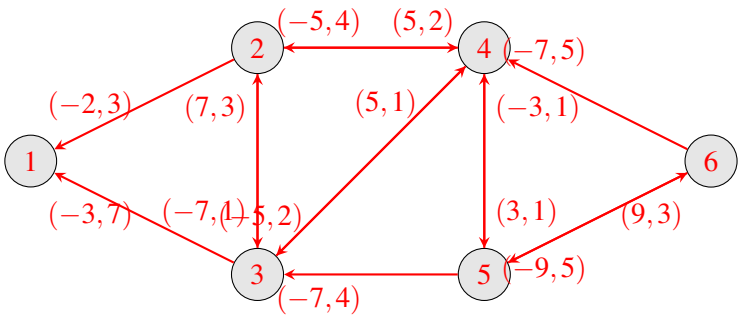


(b) (5p) Identifique ahora los flujos en arcos que transportan a mínimo costo el máximo valor de flujo F^* desde 1 hacia 6. Certifique su solución mediante una Base que cumpla la condición de optimalidad basada en costos reducidos y potenciales π'_i para cada nodo i.

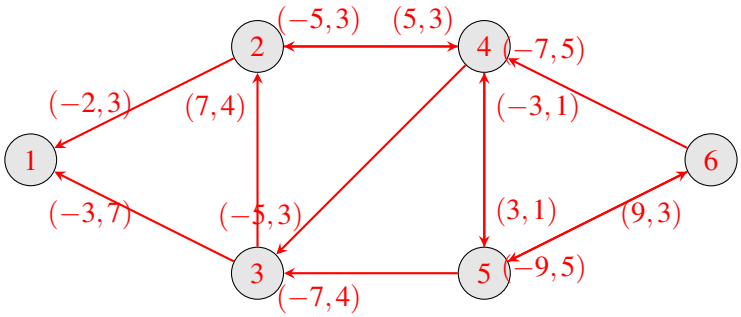
Respuesta: Acá hay tres formas de encontrar este resultado:

- ◇ Iniciar desde el flujo nulo (que es extremo de valor 0) y aumentar por rutas de mínimo costos hasta lograr el flujo extremo de mayor valor de flujo.
- ◇ Tomar en cuenta una solución (no extrema) de mayor valor de flujo e iterar enviando flujo por ciclos de costo negativo hasta lograr un flujo extremo.
- ◇ ejectuar simplex de redes o simplemente identificar al ojo una solución y validarla.

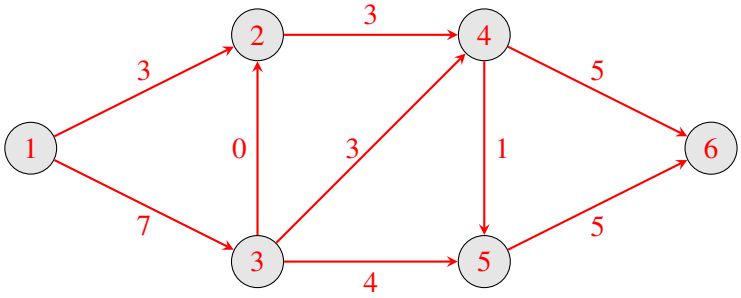
En esta ocasión ejecutaremos la segunda opción considerando el grafo residual obtenido del problema subproblema:



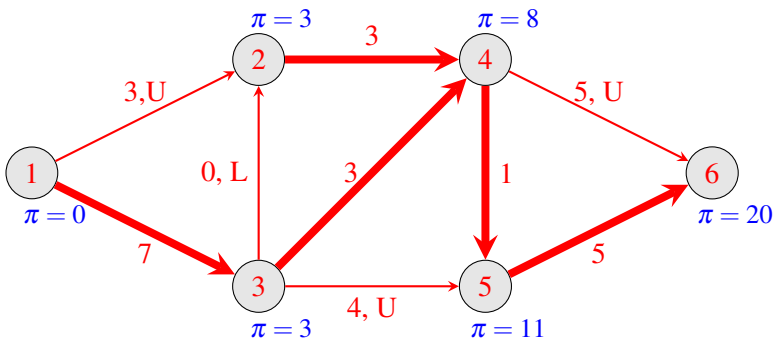
Vemos que existe el ciclo negativo 3-4-2-3 por el cual se puede enviar una unidad de flujo a costo -7 quedando el siguiente grafo residual:



Ya no hay mas ciclos negativos y, por lo tanto la solución debe ser óptima (es un flujo máximo a costo mínimo). Volviendo a los flujos originales, la solución sería la siguiente:



La siguiente base cerifica la optimalidad de los flujos:



Estos son los costos reducidos no básicos con signos correctos:

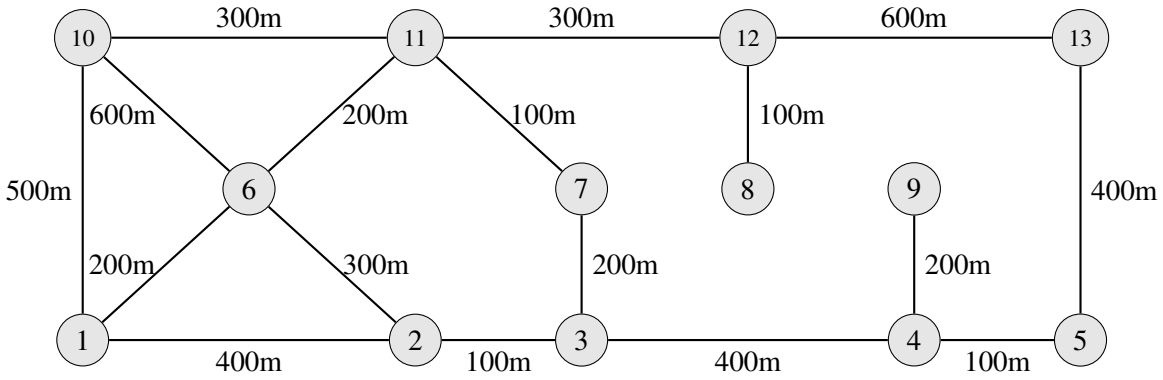
- $r_{12} = 2-(3-0) = -1 < 0$
- $r_{32} = 7-(3-3) = 7 > 0$
- $r_{35} = 7-(11-3) = -1 < 0$
- $r_{46} = 7-(20-8) = -5 < 0$

(c) (2p) Supongamos que $\pi'_1 = 0$ y que el resto de los potenciales son solución óptima. ¿Es la siguiente frase verdadera para este ejemplo? "En la solución óptima, el costo asociado a transportar cada unidad de flujo desde 1 hacia 6 es igual a π'_6."

Respuesta: Falso, es solo el costo de la última unidad transportada. Por ejemplo, el costo del flujo $1-2-4-6$ es igual a 14 y no $\pi_6 - p_1 = 0$

Problema 8 (5 puntos):

Considere el siguiente grafo que representa las calles e intersecciones de un barrio de Santiago. Cada calle (arco) detalla su longitud en metros. La suma en distancia de todas las calles es 5Km.



Suponga que, diariamente, un operador de camión se encarga de recolectar toda la basura generada ese día en el barrio. Este debe ejecutar un viaje que comienza en un depósito ubicado en el nodo 1, recorre al menos una vez cada calle (arco) del barrio en alguna de sus direcciones y termina en 1. Si el camión posee capacidad para recoger toda la basura sin necesidad de ejecutar una descarga intermedia: ¿Cuál es la mínima distancia que debe recorrer el operador a diario? Explique.

Respuesta: ¡Es un Problema de Cartero Chino!.

Para minimizar costo, los arcos $\{8,12\}$ y $\{4,9\}$ deben visitarse dos veces (ida y vuelta). Después de notar esto, se puede eliminar los nodos 8 y 9 y agregar 300m de distancia extra. Luego, quedan 4 nodos de grado impar $(1,2,3,10)$. El matching a mínimo costo es $\{2,3\}$ y $\{1,10\}$ a costo de 600m.

Así, la menor distancia mínima a recorrer es $5,9$Km (5km + 900m extra.).

Problema 9 (6 puntos + 4 extra):

Considere el Problema del Vendedor Viajero (PVV) sobre un grafo completo $G = (N, A)$, donde $N = \{1, \ldots, n\}$ es su conjunto de nodos y $A = \{(i, j) : i, j \in N, i \neq j\}$ es su conjunto de arcos. Atravesar el arco $a \in A$ cuesta $\$c_a \geq 0$. En este problema se busca un ciclo dirigido de mínimo costo que inicia en el nodo 1 (un depósito), visita a cada nodo en $N \setminus \{1\}$ exactamente una vez, y termina en 1.

Un modelo de optimización entera válido para el PVV es la formulación MTZ (Miller–Tucker–Zemlin):

$$
\begin{aligned}
\text{mín} \quad & \sum_{a \in A} c_a \cdot x_a \\
\text{s.t.} \quad & \sum_{a \in \delta_i^+} x_a = \sum_{a \in \delta_i^-} x_a = 1, \ \ \forall i \in N, \\
& u_j - u_i + (n-1) \cdot (1 - x_{ij}) \geq 1, \ \ \forall (i, j) \in A : j \neq 1, \\
& u_1 = 1, \\
& 2 \leq u_i \leq n, \ \ \forall i \in N \setminus \{1\}, \\
& x_a \in \{0, 1\}, \ \ \forall a \in A.
\end{aligned}
$$

Responda las siguientes preguntas relacionadas al modelo MTZ:

(a) (1p) Suponga que se desea restringir el espacio de búsqueda a ciclos que nunca visitan el nodo 2 antes de los nodos 3 y 4. Cambie (como usted estime) el modelo MTZ para imponer esta restricción. Explique.

Respuesta: u_i representa la posición de i en la secuencia de visita.

Agregaría las siguientes restricciones:

$$
\begin{aligned}
u_2 &\geq u_3 + 1, \\
u_2 &\geq u_4 + 1.
\end{aligned}
$$

(b) (5p) Ahora, generalice lo anterior. Suponga que para cada nodo $j \in N \setminus \{1\}$ existe un subconjunto de nodos predecesores $P_j \subset N$ que deben ser visitados antes del nodo j.

(I) (2p) Cambie (como usted estime) el modelo MTZ para asegurar el cumplimiento de estas restricciones adicionales de precedencia. Explique.

Respuesta: Agregaría las siguientes restricciones:

$$u_j - u_i \geq 1, \ \ \forall j \in N \setminus \{1\}, \forall i \in P_j$$

(II) (3p) Algunas veces las restricciones generan infactibilidad. Por ejemplo, no existe solución factible que cumpla $P_3 = \{2\}, P_4 = \{3\}, P_2 = \{4\}$. ¿Cómo podría verificar, en complejidad computacional polinomial (en función de n), si existe o no un ciclo factible? Explique.

Respuesta: Podría diseñar un nuevo grafo (N, E), donde en el conjunto de arcos $E := \{(i, j) : j \in N \setminus \{1\}, i \in P_j\}$ representa las relaciones de precedencia. Si este grafo no posee un ciclo, entonce la solución es factible. Esto se puede realizar en $O(n^2)$ mediante la búsqueda de un ordenamiento topológico.

(c) (Bono de 4 puntos extra) Retome el modelo MTZ original y suponga ahora que, adicionalmente, existen datos de tiempos de viaje $t_a \geq 0$ por arco $a \in A$ y tiempos máximos de visita τ_i por nodo $i \in N \setminus \{1\}$ (es decir, debe visitar al nodo antes de este tiempo). Suponga que el reloj comienza a contar desde el momento que la ruta inicia su recorrido en el nodo 1. Cambie (como usted estime) el modelo MTZ para asegurar que su solución visite a cada nodo antes de su tiempo máximo de visita. Explique.

Respuesta: Habría que agregar variables no-negativas por nodo (digamos $y_i \geq 0$) que modelen el tiempo de visita. El tratamiento de estas variables es muy similar al las variables u_i.

Específicamente, se debe agregar la siguiente estructura de restricciones:

$$
\begin{aligned}
& y_j - y_i + (n-1) \cdot T \cdot (1 - x_{ij}) \geq t_{i,j}, \ \ \forall (i,j) \in A : j \neq 1, \\
& y_1 = 0, \\
& 0 \leq y_i \leq \tau_i, \ \ \forall i \in N : i \neq 1,
\end{aligned}
$$

donde $T = \text{máx}_{a \in A} t_a$.