75.569 · Grafos y Complejidad · 2025-26

PEC2 - Segunda prueba de evaluación continua

Apellidos: *López Henestrosa*
Nombre: José Carlos

Presentación

Esta PEC profundiza en los conceptos básicos de la teoría de grafos que cubren los contenidos estudiados en los módulos 4 y 5 de la asignatura. Los ejercicios trabajan tanto los conceptos previos sobre grafos, como una de las clases más importantes de grafos, los árboles, así como dos de los problemas más notables de recorridos de grafos, los grafos eulerianos y los grafos hamiltonianos.

Competencias

En esta PEC se trabajan las siguientes competencias del Grado de Ingeniería Informática:

- Capacidad para utilizar los fundamentos matemáticos, estadísticos y físicos para comprender los sistemas TIC.
- Capacidad para analizar un problema en el nivel de abstracción adecuada en cada situación y aplicar las habilidades y conocimientos adquiridos para resolverlos.

Objetivos

Los objetivos concretos de esta PEC son:

- Saber caracterizar los árboles y, específicamente, los árboles con raíz.
- Saber aplicar los algoritmos de determinación de un árbol generador minimal.
- Identificar los grafos eulerianos y hamiltonianos y caracterizarlos.
- Entender el problema del viajante de comercio (TSP). Conocer y saber aplicar el algoritmo de resolución aproximada de este problema.

Respuestas

Ejercicio 1 [25 %]

[5 %] Una empresa multinacional diseña su red informática con una estructura jerárquica en forma de árbol. Cada servidor principal puede tener hasta tres servidores secundarios conectados directamente. El servidor raíz corresponde al **centro de datos central** (nivel 1) y los niveles inferiores representan delegaciones conectadas jerárquicamente.

Datos:

- Cada servidor puede tener como máximo 3 conexiones descendentes (**árbol 3-ario**).
- La red tiene un total de **6 niveles**.
- El coste de cada enlace entre servidores es de **150 €**.

a) **[5 %]** Calcula el número total máximo de servidores (nodos) que puede tener la red si todos los niveles están completamente ocupados.

Podemos calcular la cantidad de nodos en cada nivel elevando la razón de ramificación a la potencia del nivel correspondiente. Para ello, empezamos desde 0 para la raíz, por lo que el nivel 1 es 3^0:

- Nivel 1 (centro de datos): $3^0 = 1$
- Nivel 2: $3^1 = 3$
- Nivel 3: $3^2 = 9$
- Nivel 4: $3^3 = 27$
- Nivel 5: $3^4 = 81$
- Nivel 6 (último nivel): $3^5 = 243$

Sumamos los servidores que hay por nivel:

$$1 + 3 + 9 + 27 + 81 + 243 = 364 \text{ servidores}$$

5

El número total máximo de servidores que puede tener la red es de **364**.

b) **[5 %]** Determina cuántos servidores serían hojas (sin conexiones descendentes) si todos los niveles están completamente ocupados.

Sabemos que cada vértice del árbol tiene, como máximo, 3 hijos ($m = 3$). Como los niveles están completamente ocupados, estamos ante un árbol m-ario completo, donde todos los vértices internos tienen grado de salida m (3 hijos) y las hojas tienen grado de salida 0.

Si un árbol T tiene altura h y contiene t hojas, hay un máximo de m^{h-1} hojas en un árbol m-ario de altura h. En este caso, como el árbol es completo y equilibrado, calculamos ese máximo con la siguiente fórmula:

$$t = m^{h-1}$$

Sustituimos los valores, donde:

- $m = 3$ (conexiones descendentes).
- $h = 6$ (niveles).

(5)

$$t = 3^{6-1} = 243$$

Como solución, obtenemos que **243 servidores** serían hojas si todos los niveles están completamente ocupados.

c) **[5 %]** Cada enlace entre servidores cuesta 150 €. Calcula el coste total de conexiones si la red se instala con todos los enlaces necesarios si todos los niveles están completamente ocupados.

Una de las propiedades fundamentales que define a un árbol T de orden n (número de vértices/servidores) y medida m (número de aristas/enlaces) es que el número de aristas es siempre el número de vértices menos uno. Matemáticamente, esto se representa tal que así:

$$m = n - 1$$

En el apartado a), calculamos que, para esta estructura de árbol 3-ario completo con 6 niveles, el número total de servidores es $n = 364$.

Aplicamos la fórmula:

$$m = 364 - 1 = 363 \text{ enlaces}$$

Dado que cada enlace tiene un coste de 150 €, multiplicamos el número de enlaces (m) por el coste unitario.

5

$$\text{Coste total} = 363 \text{ enlaces} \times 150 \text{ €/enlace} = 54450 \text{ €}$$

El coste total de las conexiones, si la red se instala con todos los enlaces necesarios y todos los niveles están completamente ocupados, es de 54450 €.

d) **[5 %]** Si un servidor del nivel 3 deja de funcionar, ¿cuántos servidores quedarían inaccesibles? Asumimos que en la red inicial todos los niveles están completamente ocupados.

En un árbol existe un único camino entre cada pareja de vértices. En un árbol con raíz, esto implica que para llegar desde el centro de datos (raíz) a cualquier nodo descendente, el tráfico debe pasar obligatoriamente por sus padres.

Si un nodo falla, se rompe el único camino hacia sus descendientes, lo que aísla todo el subárbol que tiene como raíz al servidor averiado. El servidor averiado está en el nivel 3 y este, junto con todos sus descendientes hasta el nivel 6, quedará inaccesible.

Debemos sumar los nodos de este subárbol, que también es un árbol 3-ario completo. Desde el nivel 3 (avería) hasta el nivel 6 (final) hay **4 niveles afectados** en total (el propio servidor, los hijos de este, los hijos de los hijos, etc.).

Dicho esto, realizamos la suma de los servidores. En este caso, comenzamos por el servidor averiado (1 nodo), y multiplicamos por 3 en cada nivel descendente:

1. **Nivel 3 (servidor averiado):** $3^0 = 1$ servidor.
2. **Nivel 4:** $3^1 = 3$ servidores.
3. **Nivel 5:** $3^2 = 9$ servidor.
4. **Nivel 6 apuntes:** $3^3 = 27$ servidores.

Total servidores inaccesibles = $1 + 3 + 9 + 27 = 40$

5

El fallo de ese único nodo provoca que él mismo y sus 39 descendientes queden desconectados del centro de datos principal.

e) Indica si las siguientes afirmaciones son verdaderas o falsas y justifica brevemente tu respuesta:

Hay que dar un contraejemplo

1. **[1 %]** Todo grafo con n vértices y $n-1$ aristas es necesariamente un árbol.

0.5

Falso. Debe ser, además, conexo y acíclico.

2. **[2 %]** En un árbol con dos o más vértices, siempre hay al menos dos vértices de grado 1.

1

Verdadero. Un árbol es un grafo conexo y acíclico. Si un árbol tiene dos o más vértices, entonces necesariamente tiene al menos dos hojas. Es decir, dos vértices de grado 1. Proposición 2

3. **[2 %]** Si un grafo es acíclico y tiene una única componente conexa, es un árbol.

2

Verdadero. Un árbol es un grafo conexo acíclico. Tener una "única componente conexa" implica que el grafo es conexo. Por lo tanto, un grafo acíclico con una componente conexa es un árbol.

Ejercicio 2 [25 %]

La base marciana Prometeo está formada por varios módulos habitables $(A - F)$ conectados mediante túneles presurizados. El coste energético de mantener activo uno de los túneles depende de las condiciones ambientales, representadas por el parámetro real $r \geq 0$; mientras que otros mantienen un valor fijo. Todos los costes se muestran en la tabla siguiente:

	A	B	C	D	E	F
A	0	4	-	-	10	12
B	4	0	6	11	-	7
C	-	6	0	5	-	-
D	-	11	5	0	3	-
E	10	-	-	3	0	$1+r$
F	12	7	-	-	$1+r$	0

Un - indica que no existe conexión directa.

En los apartados en los que sea necesario utilizar algoritmos, estos deben corresponder a los presentados en los módulos de la asignatura y deben incorporarse siguiendo el mismo formato y nomenclatura utilizados en ellos.

a) **[10 %]** Determina la forma más económica, así como el coste asociado, de conectar todos los módulos en función de r.

Para ello, utilizaremos el algoritmo de Kruskal, tal y como se presenta en los módulos de la asignatura, ya que es el más adecuado para gestionar pesos variables al ordenar las aristas de menor a mayor coste.

Para aplicar el algoritmo, primero tenemos que extraer todas las aristas de la matriz de adyacencia y ordenarlas por peso (coste). Hay que tener en cuenta que la arista $\{E, F\}$ tiene un peso variable $1 + r$.

ARISTA	COSTE	NOTAS
$\{D, E\}$	3	Fijo
$\{A, B\}$	4	Fijo
$\{C, D\}$	5	Fijo
$\{B, C\}$	6	Fijo
$\{B, F\}$	7	Fijo
$\{A, E\}$	10	Fijo (coste alto)
$\{B, D\}$	11	Fijo (coste alto)

$\{A, F\}$	12	Fijo (coste alto)
$\{E, F\}$	$1 + r$	Variable ($r \geq 0$)

El algoritmo selecciona aristas de menor a mayor peso, siempre que no formen ciclos, hasta seleccionar $n - 1$ aristas, donde n representa los vértices (6 módulos habitables → 6 vértices).

Sustituimos la fórmula para determinar el número de aristas a seleccionar:

$$n - 1 = 6 - 1 = 5$$

A continuación, procedemos a determinar las aristas fijas iniciales. Independientemente del valor de r (dado que $r \geq 0$, el coste mínimo de $\{E, F\}$ es 1), las siguientes aristas son siempre las más económicas y no forman ciclos entre ellas:

1. $\{D, E\}$ (coste 3). Componentes: $\{D, E\}$.
2. $\{A, B\}$ (coste 4). Componentes: $\{D, E\}, \{A, B\}$
3. $\{C, D\}$ (coste 5). Une C con $\{D, E\}$. Componentes: $\{C, D, E\}, \{A, B\}$.
4. $\{B, C\}$ (coste 6). Une los dos componentes del paso anterior. Componentes resultantes: $\{A, B, C, D, E\}$.

En este punto, tenemos conectado todo el grafo excepto el módulo F. Tenemos 4 aristas seleccionadas y necesitamos 1 más. Para conectarlo al resto de la red, debemos elegir la arista más barata disponible que incida en F. Las opciones son:

- Conexión $\{E, F\}$ con coste $1 + r$.
- Conexión $\{B, F\}$ con coste 7.
- Conexión $\{A, F\}$ con coste 12. Al ser más cara que la conexión $\{B, F\}$, se descarta.

El algoritmo elegirá la arista $\{E, F\}$ siempre que su coste sea menor que el de la alternativa $\{B, F\}$ (o igual, dependiendo del criterio de desempate, pero asumiremos que el coste es menor o igual para la transición).

Comparamos los costes:

$$1 + r \leq 7 \rightarrow r \leq 6$$

Esto nos genera dos casos posibles:

1. Si el parámetro r es bajo ($0 \leq r \leq 6$), la conexión $\{E, F\}$ de coste variable es más eficiente que la conexión $\{B, F\}$ de coste fijo 7.
   - **Aristas seleccionadas:** $\{D, E\}, \{A, B\}, \{C, D\}, \{B, C\}, \{E, F\}$.
   - **Coste total:** $3 + 4 + 5 + 6 + (1 + r) = 19 + r$
   - **Estructura:** El módulo F se conecta a la red a través del módulo E.

2. Si el parámetro r supera el umbral de 6, el coste de la conexión $\{E, F\}$ de coste variable supera 7, por lo que el algoritmo de Kruskal seleccionará la arista fija $\{B, F\}$ antes que la $\{E, F\}$.

10

- **Aristas seleccionadas:** $\{D, E\}, \{A, B\}, \{C, D\}, \{B, C\}, \{B, F\}$.
- **Coste total:** $3 + 4 + 5 + 6 + 7 = 25$
- **Estructura:** El módulo F se conecta a la red a través del módulo B.

b) **[10 %]** Cada mañana, antes del inicio de operaciones, el sistema central de control reinicia la red de túneles. Todas las compuertas permanecen cerradas y el protocolo ordena reactivar la red comenzando desde el módulo A. En cada paso, el sistema selecciona el túnel de menor coste que conecta con alguno de los módulos ya activos, asegurándose de no formar bucles en la red.

¿Cuál crees que es el algoritmo que describe estas instrucciones? Realiza la apertura de los túneles siguiendo este procedimiento, indicando el orden de activación y el coste total del sistema un día en que $r = 8$.

El procedimiento descrito corresponde con el algoritmo de Prim. El sistema comienza construyendo la red a partir de un único módulo (A) y, en cada iteración, selecciona el túnel de menor coste que conecta un módulo activado con uno inactivo. Al incorporar siempre un módulo nuevo y descartar cualquier conexión que produciría un ciclo, el proceso siempre obtiene un árbol de expansión mínima. Esta secuencia de decisiones coincide exactamente con el funcionamiento del algoritmo de Prim aplicado a la construcción de un árbol generador minimal en un grafo conectado y ponderado.

Para realizar la apertura, actualizamos primero el coste del enlace variable $\{E, F\}$ con $r = 8$:

$$\text{Coste } \{E, F\} = 1 + r = 1 + 8 = 9$$

A continuación, ejecutamos el algoritmo paso a paso partiendo del módulo A. Seleccionamos siempre la arista más barata que conecte un módulo activo (conjunto U) con uno inactivo ($V - U$), tal y como define el algoritmo.

Estado inicial:

- **Módulos activos** (U)**:** $\{A\}$
- **Aristas candidatas**: $\{A, B\} = 4$, $\{A, E\} = 10$, $\{A, F\} = 12$

Iteración 1:

1. Seleccionamos la arista mínima conectada a U: $\{A, B\}$ (coste 4).
2. Activamos el módulo B.
3. **Módulos activos (**U**)**: $\{A, B\}$.
4. **Nuevas aristas candidatas desde B**: $\{B, C\} = 6$, $\{B, D\} = 11$, $\{B, F\} = 7$.

Iteración 2:

1. Evaluamos las aristas candidatas disponibles ($\{A, E\}$, $\{A, F\}$, $\{B, C\}$, $\{B, D\}$, $\{B, F\}$)
2. Seleccionamos la mínima: $\{B, C\}$ (coste 6).
3. Activamos el módulo C.
4. **Módulos activos (**U**)**: $\{A, B, C\}$.
5. **Nuevas aristas candidatas desde C**: $\{C, D\} = 5$.

Iteración 3:

1. Evaluamos las aristas candidatas disponibles. La menor es $\{C, D\}$ con coste 5, que es menor que $\{B, F\}$ (7) o $\{A, E\}$ (10).
2. Seleccionamos la mínima: $\{C, D\}$ (coste 5).
3. Activamos el módulo D.

4. **Módulos activos (U)**: $\{A, B, C, D\}$.
5. **Nuevas aristas candidatas desde D**: $\{D, E\} = 3$.

Iteración 4:

1. Evaluamos las aristas candidatas disponibles. La arista $\{D, E\}$ tiene coste 3. Es la más barata de todas las opciones actuales.
2. Seleccionamos la mínima: $\{D, E\}$ (coste 3).
3. Activamos el módulo E.
4. **Módulos activos (U)**: $\{A, B, C, D, E\}$.
5. **Nuevas aristas candidatas desde E**: $\{E, F\} = 9$.

Iteración 5:

1. Evaluamos las conexiones restantes hacia el único módulo inactivo (F):
   - Desde A: $\{A, F\} = 12$.
   - Desde B: $\{B, F\} = 7$.
   - Desde E: $\{E, F\} = 9$ $(r = 8)$.
2. Seleccionamos la mínima: $\{B, F\}$ (coste 7).
3. Activamos el módulo F.
4. **Módulos activos (U)**: $\{A, B, C, D, E, F\}$.

Para recapitular, el orden de selección de aristas y activación de módulos es el siguiente:

1. Activar A (inicio).
2. Túnel $\{A, B\}$ (coste 4) → Activa B.
3. Túnel $\{B, C\}$ (coste 6) → Activa C.
4. Túnel $\{C, D\}$ (coste 5) → Activa D.
5. Túnel $\{D, E\}$ (coste 3) → Activa E.
6. Túnel $\{B, F\}$ (coste 7) → Activa F.

5

Por lo tanto, el coste total del sistema con $r = 8$ es el siguiente:

$$4 + 6 + 5 + 3 + 7 = 25$$

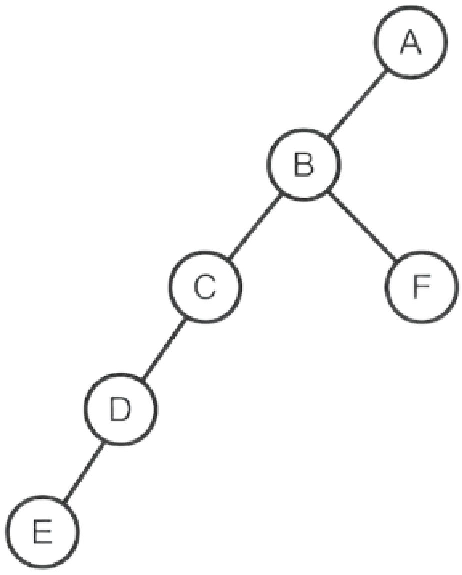
Usa el formato de los apuntes

c) **[5 %]** Con la red ya operativa para el caso $r = 8$, el sistema debe inspeccionar todas las compuertas.

Representa la red de túneles abiertos obtenida en el apartado anterior como un árbol con raíz en el módulo A. En caso de que un módulo tenga un único hijo, se considerará como hijo izquierdo. Si tiene varios, ordénalos alfabéticamente de izquierda a derecha. Si el árbol obtenido es binario, indica el orden en que el sistema visitaría los módulos si la inspección se realiza en **preorden**, **inorden** o **postorden**.

La red activa resultante conecta los módulos mediante las aristas $\{A, B\}, \{B, C\}, \{B, F\}, \{C, D\}, \{D, E\}$.

Si seguimos las instrucciones de ordenación dadas en el enunciado (hijo único → izquierdo; múltiples hijos → orden alfabético de izquierda a derecha), la estructura jerárquica resultante es la siguiente:



En este caso, podemos apreciar que estamos ante un árbol binario, ya que cada vértice tiene, como máximo, 2 hijos. En este caso, el nodo con más ramificaciones es B (tiene 2 hijos), y el resto tiene 1 o 0, por lo que cumple la condición.

Si aplicamos los algoritmos de inspección preorden, inorden y postorden al árbol binario resultante, obtenemos los siguientes resultados:

Recorrido en preorden (raíz - izquierda - derecha)

El sistema visita primero la raíz y luego explora recursivamente los subárboles:

1. Visita **A**.
2. Baja a la izquierda a **B**.
3. Desde B, baja a la izquierda a **C**.
4. Desde C, baja a la izquierda a **D**.

5. Desde D, baja a la izquierda a **E**.
6. Termina rama izquierda de D, C y B.
7. Desde B, baja a la derecha a **F**.

Orden: $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow F$

Recorrido en inorden (izquierda - raíz - derecha)

El sistema explora el subárbol izquierdo, luego visita la raíz y finalmente el subárbol derecho. Dado que A, C y D solo tienen un hijo izquierdo, la visita al nodo padre ocurre después de visitar toda su descendencia izquierda.

1. Baja por la izquierda hasta el fondo: $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E$
2. Visita **E** (no tiene hijos).
3. Sube y visita **D**.
4. Sube y visita **C**.
5. Sube y visita **B** (su izquierda queda visitada completamente).
6. Baja a la derecha de B y visita **F**.
7. Sube hasta la raíz y visita **A** (su izquierda queda visitada completamente y no tiene derecha).

Orden: $E \rightarrow D \rightarrow C \rightarrow B \rightarrow F \rightarrow A$

Recorrido en postorden (izquierda - derecha - raíz)

El sistema explora los subárboles descendentes antes de visitar el propio nodo.

1. Baja por la izquierda hasta el fondo (**E**).
2. Visita **E**.
3. Sube a D. Su izquierda queda visitada completamente y no tiene derecha, por lo que visita a **D**.
4. Sube a C. Su izquierda queda visitada completamente y no tiene derecha, por lo que visita a **C**.
5. Sube a B. Su izquierda queda visitada completamente y tiene derecha, por lo que visita a **F**.
6. Visita **F**.
7. Ahora que B ha completado su izquierda y su derecha, visita **B**.
8. Sube a A. Su izquierda queda visitada completamente y no tiene derecha, por lo que visita a **A**.

5

Orden: $E \rightarrow D \rightarrow C \rightarrow F \rightarrow B \rightarrow A$

Ejercicio 3 [25 %]

a) **[5 %]** Explica con tus palabras qué diferencia existe entre un *grafo euleriano* y un *grafo hamiltoniano*.

4

Un grafo es **euleriano** si contiene un ~~ciclo~~ circuito euleriano. Esto implica un recorrido cerrado que recorra todas las **aristas** del grafo exactamente una vez y regrese al punto de partida. Está permitido pasar por un mismo vértice múltiples veces si es necesario para cruzar todas las aristas incidentes.

En cambio, un grafo es **hamiltoniano** si contiene un ciclo hamiltoniano. Esto es un recorrido cerrado que visita todos los **vértices** del grafo exactamente una vez sin repeticiones, salvo el vértice inicial/final para cerrar el ciclo. A diferencia de un ciclo euleriano, un ciclo hamiltoniano no necesita recorrer todas las aristas del grafo. Solo las necesarias para conectar todos los vértices.

b) Indica las condiciones necesarias y suficientes (o señala que no existen/se desconocen) para que un grafo conexo sea:

1) **[2 %]** euleriano,

2

Un grafo conexo es euleriano si y solo si todos sus vértices son de grado par.

2) **[2 %]** semieuleriano (tenga recorrido euleriano),

2

Un grafo conexo es semieuleriano (con recorrido euleriano) si y solo si el grafo tiene exactamente dos vértices de grado impar.

3) **[5 %]** hamiltoniano. Son condiciones necesarias, pero no suficientes

0

Un grafo conexo es hamiltoniano si se cumplen estos requisitos:

- Todos sus vértices tienen grado mayor o igual que 2
- Es 2-conexo. Esto significa que no debe existir ningún vértice cuya eliminación desconecte al grafo.
- Para cualquier subconjunto de vértices S (no vacío), si eliminamos S del grafo, el número de componentes conexas resultantes, denotado como $c(G, S)$, no puede superar al número de vértices eliminados. Matemáticamente, esto se representa como $c(G, S) \leq |S|$.
- Si el grafo es bipartito (sus vértices se pueden dividir en dos conjuntos V_1 y V_2), entonces ambos conjuntos deben tener el mismo número de elementos: $|V_1| = |V_2|$.

c) **[5 %]** Explica por qué es más sencillo determinar si un grafo es euleriano que saber si es hamiltoniano.

Un grafo es euleriano si y solo si se cumple esta única condición: que todos sus vértices tienen grado par. En este caso, no hay ambigüedad, ya que si los grados son pares, el grafo es euleriano y, si hay alguno impar, entonces no lo es.

2

En cambio, existen varias condiciones para que un grafo sea hamiltoniano. Son más complicadas de demostrar que en el caso de los grafos eulerianos, por lo que, a veces, la única forma de saber si un grafo es hamiltoniano es intentar construir su ciclo.

No se conocen condiciones necesarias y suficientes para determinar eficientemente si un grafo es hamiltoniano o no

d) Considera el siguiente grafo $G = (V, A)$ con vértices

$$V = \{A, B, C, D, E, F\}$$

y aristas

$$A - B,\ A - C,\ B - C,\ B - D,\ C - E,\ D - E,\ D - F,\ E - F$$

1) **[2 %]** Calcula el grado de cada vértice.

Para calcular el grado de cada vértice $g(v)$, tenemos que contar el número de aristas que conectan con cada uno de ellos.

2

- **Grado de A:** 2
  - Aristas incidentes: $A - B,\ A - C$.
- **Grado de B:** 3
  - Aristas incidentes: $A - B,\ B - C,\ B - D$.
- **Grado de C:** 3
  - Aristas incidentes: $A - C,\ B - C,\ C - E$.
- **Grado de D:** 3
  - Aristas incidentes: $B - D,\ D - E,\ D - F$.
- **Grado de E:** 3
  - Aristas incidentes: $C - E,\ D - E,\ E - F$.
- **Grado de F:** 2
  - Aristas incidentes: $D - F,\ E - F$.

2) **[2 %]** Determina si G es euleriano o semieuleriano.

2

Para que un grafo sea euleriano, todos los vértices deben tener grado par. En este caso, tenemos cuatro vértices con grado impar (B, C, D y E tienen grado 3). Por lo tanto, G **no es euleriano**.

Un grafo es semieuleriano (con recorrido euleriano) si y solo si el grafo tiene exactamente dos vértices de grado impar. En este caso, tenemos cuatro vértices con grado impar, por lo que **tampoco es semieuleriano**.

3) **[5 %]** Determina si G es hamiltoniano. Si lo es, escribe un posible ciclo hamiltoniano. En caso contrario, justifica por qué.

Para ello, tenemos que explorar caminos intentando conectar todos los vértices sin repetir ninguno y cerrar el ciclo. La ruta tiene que cumplir estas tres reglas estrictas:

1. Empezar en un vértice.
2. Visitar todos los demás vértices exactamente una vez.
3. Desde el último vértice, debe haber un camino directo para volver al vértice de inicio.

Dicho esto, vamos a ponderar las opciones que tenemos en cada punto según las aristas dadas en el enunciado:

- **A** conecta con B, C
- **B** conecta con A, C, D
- **C** conecta con A, B, E
- **D** conecta con B, E, F
- **E** conecta con C, D, F
- **F** conecta con D, E

Vamos a intentar construir el camino por pasos comenzando por A.

Paso 1: Inicio

Empezamos en **A**, el cual es el primer vértice que visitamos. A partir de aquí, podemos ir a B o a C. En este caso, iremos a C.

Paso 2: Desde C

Estamos en **C**, el cual conecta con A (ya visitado). A partir de C, podemos ir a B y E. Si vamos a B, podríamos quedarnos atrapados cerca del inicio, por lo que optamos por E.

Paso 3: Desde E

Estamos en **E**, el cual conecta con C (ya visitado). A partir de E, podemos ir a D y F. Probemos a ir a F.

Paso 4: Desde F

Estamos en **F**, el cual conecta con E (ya visitado) y D. Solo nos queda una opción lógica para no repetir ni quedarnos bloqueados, la cual es D.

Paso 5: Desde D

Estamos en **D**, el cual conecta con E (ya visitado), F (ya visitado) y B. Solo nos queda una opción lógica para no repetir ni quedarnos bloqueados, la cual es B.

Paso 6: Cierre

Estamos en **B**. Ya hemos visitado todos los vértices, por lo que nos queda conectar la posición actual (B) con el vértice de inicio (A). En este caso, existe la arista $A - B$, por lo que el ciclo se cierra correctamente.

2

Como podemos comprobar, la ruta $A \to C \to E \to F \to D \to B$ demuestra que el grafo G **es hamiltoniano**.