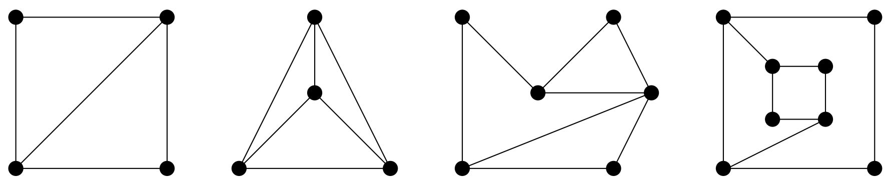
# Correspondencia, Clique, Conjunto Independiente y Cubrimientos

7.1. Para cada grafo de la figura, encontrar:

a. las correspondencias maximales y cuáles son máximas.
b. los conjuntos independientes maximales y cuáles son máximos.
c. los recubrimientos minimales de vértices por aristas y cuáles son mínimos.
d. los recubrimientos minimales de aristas por vértices y cuáles son mínimos.



7.2. Decidir si es verdadera o falsa cada una de las siguientes afirmaciones. Si es verdadera probarla y si es falso encontrar un contraejemplo.

a. Todo grafo tiene al menos una correspondencia.
b. Todo grafo tiene al menos un conjunto independiente.
c. Todo grafo tiene al menos un recubrimiento de vértices por aristas.
d. Todo grafo tiene al menos un recubrimiento de aristas por vértices.
e. En todo grafo, cualquier recubrimiento de vértices por aristas tiene una cantidad de elementos mayor o igual que la de cualquier correspondencia.
f. En todo grafo, cualquier recubrimiento de aristas por vértices tiene una cantidad de elementos mayor o igual que la de cualquier conjunto independiente.
g. Cualquier recubrimiento de vértices por aristas contiene una correspondencia máxima.

7.3. En este ejercicio, recubrimiento indica recubrimiento de vértices por aristas. Probar que:

a. Un grafo tiene un recubrimiento si y sólo si no tiene vértices aislados.
b. Si un grafo tiene $n$ vértices, cualquier recubrimiento tiene al menos $\lceil n/2 \rceil$ aristas.
c. Todo recubrimiento contiene un recubrimiento minimal.
d. Un recubrimiento minimal no contiene circuitos.
e. ¿Cuál es el máximo número de aristas que puede tener un recubrimiento minimal?
f. Un recubrimiento es minimal si y sólo si no contiene caminos ni circuitos simples de longitud mayor o igual a 3.

7.4. Revisar los ejercicios de modelización de la guía de introducción a grafos. ¿Cuáles se pueden modelizar como problemas de correspondencia, recubrimiento o de conjunto independiente? ¿Cómo se los puede resolver?

7.5. Probar que si $G$ es un grafo, se verifica que $\tau \geq \nu$ donde:

- $\tau$ es el cardinal de cualquier recubrimiento de aristas por vértices
- $\nu$ es el cardinal de cualquier correspondencia

7.6. Probar que si $G$ es un grafo sin vértices aislados, se verifica que $\rho \geq \alpha$ donde:

- $\rho$ es el cardinal de cualquier recubrimiento de vértices por aristas
- $\alpha$ es el cardinal de cualquier conjunto independiente

7.7. Sea $G$ un grafo bipartito de $m$ ejes. Sea $\alpha_{max}$ la cantidad de vértices de un conjunto independiente máximo de $G$. Sea $\tau$ la cantidad de vértices de cualquier cubrimiento de aristas por vértices de $G$. Demostrar que $m \leq \alpha_{max} * \tau$.

7.8. Analizar el siguiente algoritmo para determinar un conjunto independiente en un grafo G:

`PASO 1:` `Poner` $I \leftarrow \emptyset$
`PASO 2:` `Mientras` $V \neq \emptyset$ `hacer`
`PASO 3:` `Encontrar un vértice` $x$ `tal que` $d(x, G)$ `sea mínimo.`
`PASO 4:` `Poner` $I \leftarrow I \cup \{x\}$
`PASO 5:` `Poner` $V \leftarrow V \backslash (\{x\} \cup \{y : y$ `adyacente a` $x\})$.
`PASO 6:` `Poner` $G \leftarrow (V, X(V))$.
`PASO 7:` `Informar` $I$

a. Probar que el algoritmo anterior encuentra un conjunto independiente en $G$.
b. Determinar la complejidad del mismo. ¿Qué técnica utiliza?
c. Mostrar que el algoritmo no necesariamente encuentra un conjunto independiente máximo.
d. Dado que no se conocen algoritmos de complejidad polinomial para el problema de determinar un conjunto independiente máximo en un grafo, ¿se podría usar este algoritmo como algoritmo aproximado para resolver el problema? ¿Sería un "buen" algoritmo aproximado?

7.9. Analizar el siguiente algorimo para colorear un grafo $G$:

`PASO 1:` $U \leftarrow V$ `,` $k \leftarrow 0$
`PASO 2:` `Mientras` $U \neq \emptyset$ `hacer`
`PASO 3:` `Poner` $k \leftarrow k + 1$
`PASO 4:` `determinar un conjunto independiente máximo` $W$ `en el subgrafo de` $G$ `generado por` $U$.
`PASO 5:` `Asignar a los nodos de` $W$ `el color` $k$.
`PASO 6:` `Poner` $U \leftarrow U \backslash W$.
`PASO 7:` `Informar` $k$`, que es el número de colores con que se ha coloreado el grafo.`

a. Probar que el algoritmo produce un coloreo válido de $G$. ¿Es cierto que $k = \chi(G)$?
b. ¿Cuál es la complejidad del algoritmo?
c. Si el problema de encontrar un conjunto independiente máximo pudiera ser resuelto en forma polinomial, ¿esto implicaría, a partir del algoritmo anterior, que lo mismo ocurriría para el problema de coloreo de grafos?
d. Probar que reemplazando el `PASO 4` por el algoritmo del ejercicio 7.8 se obtiene una heurística de complejidad polinomial para el problema de determinar el numero cromático de un grafo. ¿Es una "buena" heurística?