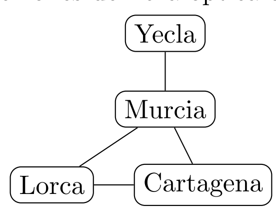
GRAFOS
ÁLGEBRA Y MATEMÁTICA DISCRETA. GII - FIUM.

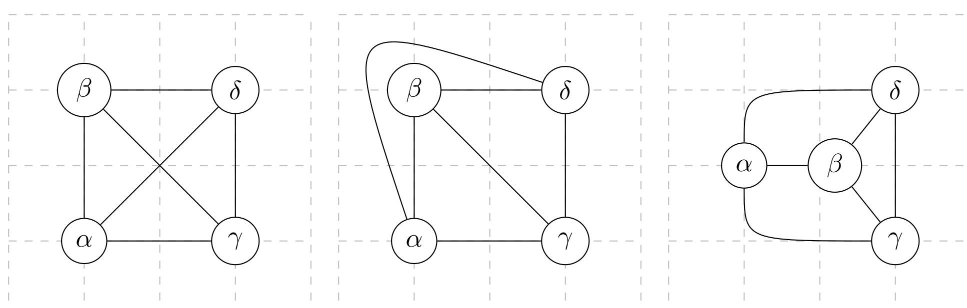
Definición 1. *Un grafo está formado por un conjunto V cuyos elementos llamaremos* **vértices** *y por un conjunto E a cuyos elementos llamaremos* **aristas**. *Una arista es un conjunto de dos vértices $\{u, v\}$ que llamaremos extremos.*

Esta definición se puede aplicar a múltiples problemas y el significado de vértices y aristas depende de lo que queramos representar. Por ejemplo, los vértices podrían ser ciudades y las aristas conexiones de fibra óptica entre ellas.



La representación gráfica no es parte fundamental del grafo. Los vértices y las aristas las podemos posicionar en lugares diferentes siendo exactamente el mismo grafo. Además, los cruces entre aristas no se considerarán vértices a no ser que lo indiquemos expresamente.

Aquí tenemos tres representaciones del mismo grafo:



Para dibujar este tipo de grafos utilizaremos una librería de latex que se llama TiKZ. El ejemplo más sencillo es cuando las aristas son líneas rectas entre los vértices (el primero de los casos) cuyo código sería el siguiente:

```
\begin{tikzpicture}[vertice/.style = {fill=white,circle,draw}]
\node[vertice] (A) at (0,0) {$\alpha$};
\node[vertice] (B) at (0,2) {$\beta$};
\node[vertice] (C) at (2,0) {$\gamma$};
\node[vertice] (D) at (2,2) {$\delta$};

\draw (A) -- (B);
\draw (A) -- (C);
\draw (A) -- (D);
```

```
\draw (B) -- (C);
\draw (B) -- (D);
\draw (C) -- (D);
\end{tikzpicture}
```

En este ejemplo hemos establecido un estilo que hemos llamado `vertice` y al que le hemos dicho que ponga un fondo blanco y rodee con un círculo. A este estilo le hemos dado nombre porque lo queremos utilizar en muchos vértices distintos. Las aristas simplemente las hemos dibujado uniendo los pares de vértices correspondientes, sin ninguna información adicional.

En el segundo dibujo hemos dado una información adicional al dibujar la arista para que no sea una línea recta sino que describa una curva mediante un punto de control. Concretamente la arista en ese caso es

```
\draw (A) .. controls (0,2) .. (D);
```

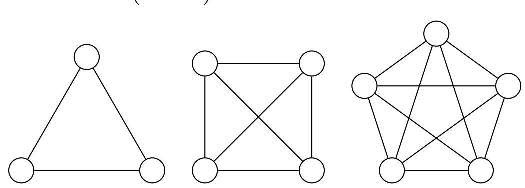
El punto de control le indica que debe curvar la línea en la dirección que indica el punto de control. La fórmula exacta de cómo lo hace no nos importa demasiado y podemos jugar con posicionar dicho punto de control en diferentes lugares para ver el resultado. En el tercer ejemplo hemos cambiado las posiciones de dos de los vértices y hemos añadido puntos de control a dos aristas. Puesto que la representación gráfica no cambia el problema, no nos debemos preocupar demasiado por ella, es una mera cuestión estética y ante la duda, podemos poner simplemente líneas rectas para las aristas.

Definición 2. *Llamaremos grado de un vértice al número de aristas que entran (o salen) de un vértice. Lo denotaremos* $\mathtt{gr}(v)$ *o* $\delta(v)$*.*

Puesto que cada arista tiene dos extremos, contribuye en dos unidades al total de la suma de los grados de los vértices de un grafo, lo que nos permite establecer el siguiente resultado:

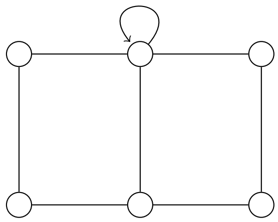
Lema 3 (Lema del Apretón de Manos)**.** *La suma de los grados de todos los vértices de un grafo es igual al doble del número de aristas.*

Podemos utilizar este lema para calcular el número de aristas del grafo K_n, que es el **grafo completo de n vértices** en el que cada vértice está unido con los $n-1$ restantes. Por lo tanto el grado de cada vértice es $n-1$ y la suma total de los grados de los vértices es $n(n-1)$. El número de aristas será pues $\frac{n(n-1)}{2}$.

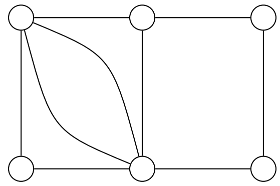


K_3 K_4 K_5

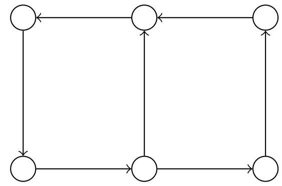
Existen conceptos relacionados con los grafos pero que no son grafos según nuestra definición. Por un lado tenemos los **pseudografos**, que son aquellos en los que permitimos aristas que unen un vértice consigo mismo (este tipo de aristas se llaman lazos o *loop* en inglés). Cuando se acepta que puedan existir varias aristas entre dos vértices de un grafo, estaremos hablando de un **multigrafo**. Si las aristas tienen una dirección para ser recorridas, estaremos ante un **grafo dirigido**.



Pseudografo



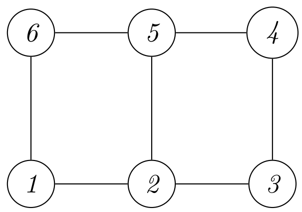
Multigrafo



Grafo Dirigido

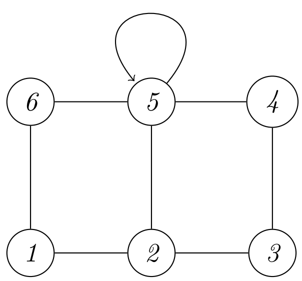
Definición 4. *Sea $G = (V, E)$ un grafo con n vértices que numeraremos $V = \{v_1, v_2, \cdots, v_n\}$. Llamaremos matriz de adyacencia de G a la matriz A tal que $A_{i,j}$ es 1 cuando exista una arista $\{v_i, v_j\}$ y 0 cuando no exista.*

Ejemplo 5. *Aquí tenemos un grafo junto con su matriz de adyacencia:*

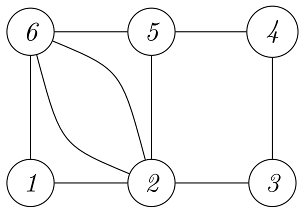


$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

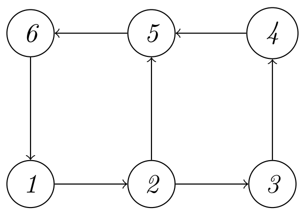
También podemos definir la matriz de adyadencia en el caso de pseudografos, multigrafos y grafos dirigidos.



$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & \mathbf{1} & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$



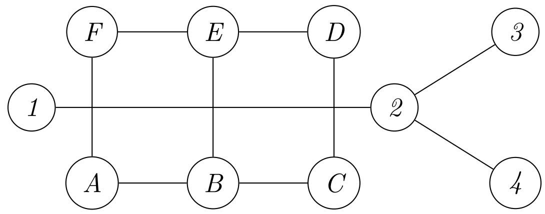
$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & \mathbf{2} \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & \mathbf{2} & 0 & 0 & 1 & 0 \end{bmatrix}$$



$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Como podemos observar, en la matriz de adyacencia de un grafo no hay elementos mayores que 1 *(estaríamos ante un multigrafo), ni elementos en la diagonal principal (sería un pseudografo) y la matriz es simétrica (si no lo fuera, sería un grafo dirigido).*

Definición 6. *Sea* $G = (V, E)$ *un grafo, por ejemplo*



- *Llamaremos un* **camino entre los vértices** u **y** v *a una sucesión de vértices* $w_0, w_1, \cdots, w_n \in V$ *tales que* $w_0 = u$*,* $w_n = v$ *y* $\{w_{i-1}, w_i\} \in E$ *para* $i = 1, \cdots, n$*. Lo denotaremos* $(w_0, w_1, \cdots, w_n)$ *o por* $w_0 - w_1 - \cdots - w_n$*. Un ejemplo de camino en el grafo anterior sería* $A - B - E - D - C$ *que usanto TiKZ podríamos pintar como*

  ```
  \draw (A) -- (B) -- (E) -- (D) -- (C);
  ```
- *La* **longitud del camino** *es el número de aristas que lo forman. Si el camino es* $(w_0, w_1, \cdots, w_n)$ *su longitud es* n*.*
- *Diremos que los vértices* u *y* v *están* **conectados** *cuando exista un un camino entre ellos. En el ejemplo que estamos usando, los vértices marcados con letras están conectados entre sí y los vértices marcados con números están también conectados entre sí.*
- *Se considera que todo vértice* v *está unido consigo mismo por un camino de longitud* 0 *que es* (v)*.*
- *Si tenemos un camino que une* u *y* v *y uno que une* v *y* w *podemos* **yuxtaponerlos** *que no es mas que poner uno a continuación del otro para obtener un camino entre* u *y* w*.*
- *Si tenemos un camino que une* u *y* v*, recorriéndolo en sentido inverso, tenemos un camino que une* v *y* u*.*
- *Un grafo se dice* **conexo** *si todos los vértices están unidos por algún camino. El grafo del ejemplo no es conexo.*
- *Todo grafo se puede dividir en* **componentes conexas** *que son los trozos formados por los vértices que están conectados entre sí. Será conexo cuando tenga una única componente conexa. En el ejemplo, el grafo tiene dos componentes conexas, la formada por los números y la formada por las letras.*
- *Un camino se dice* **simple** *si no se pasa dos veces por ningún vértice. Siempre que exista un camino entre dos vértices, existe un camino simple porque podemos eliminar tramos innecesarios, por ejemplo, el camino que une* A *y* C *formado por*

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

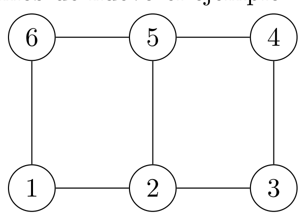
  no es simple porque repite varios vértices, por ejemplo B*. Entonces podemos eliminar todo el recorrido que hay desde que pasamos por primera vez por* B *hasta que pasamos por última vez por dicho vértice y eliminar la repetición. Haciéndolo con todas las repeticiones llegaríamos finalmente a un camino* *simple:*

$$A - B {\color{red}- E - D - E - F - E - B} - C \quad \rightarrow \quad A - B - C$$

Teorema 7. *Sea G un grafo con vértices numerados $V = \{v_1, v_2, \cdots, v_n\}$ y sea A su matriz de adyacencia. Entonces para todo k, la posición (i,j) de la matriz A^k nos da el número de caminos de longitud k entre los vértices v_i y v_j.*

Si consideramos que cualquier matriz elevada a 0 es la identidad, este resultado también es válido para los caminos de longitud 0.

Consideremos de nuevo el ejemplo



$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

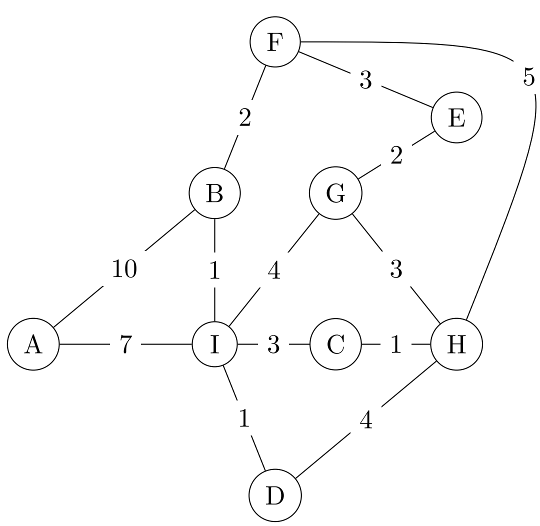
En este caso tenemos que

$$A^3 = \begin{bmatrix} 0 & 5 & 0 & 3 & 0 & 4 \\ 5 & 0 & 5 & 0 & 7 & 0 \\ 0 & 5 & 0 & 4 & 0 & 3 \\ 3 & 0 & 4 & 0 & 5 & 0 \\ 0 & 7 & 0 & 5 & 0 & 5 \\ 4 & 0 & 3 & 0 & 5 & 0 \end{bmatrix}$$

Esta matriz nos dice que entre el vértice (1) y el vértice (4) tenemos tres caminos de longitud 3 y si lo comprobamos, podemos ver que son $1-2-3-4, 1-2-5-4$ y $1-6-5-4$.

Definición 8. *Un* **grafo con pesos o ponderado** *es un grafo en el que cada arista tiene asignado un valor numérico que llamaremos peso o coste.*

Ejemplos de utilidad para este tipo de grafos serían ciudades y distancias entre ellas o redes en las cuales debemos calcular el tiempo que se tarda en moverse de un vértice a otro o incluso monedas y comisones de compra-venta para convertirlas entre ellas.



El grafo anterior se podría pintar del siguiente modo usando TiKZ :

```
\begin{tikzpicture}[vertice/.style = {fill=white,circle,draw}]
\node[vertice] (A) at (-1,1) {A};
\node[vertice] (B) at (2,3) {B};
\node[vertice] (C) at (4,1) {C};
\node[vertice] (D) at (3,-1) {D};
\node[vertice] (E) at (6,4) {E};
\node[vertice] (F) at (3,5) {F};
\node[vertice] (G) at (4,3) {G};
\node[vertice] (H) at (6,1) {H};
\node[vertice] (I) at (2,1) {I};

\draw (A) -- node[fill=white] {10} (B);
\draw (A) -- node[fill=white] {7} (I);
\draw (B) -- node[fill=white] {1} (I);
\draw (I) -- node[fill=white] {3} (C);
\draw (I) -- node[fill=white] {1} (D);
\draw (H) -- node[fill=white] {4} (D);
\draw (I) -- node[fill=white] {4} (G);
\draw (G) -- node[fill=white] {2} (E);
\draw (G) -- node[fill=white] {3} (H);
\draw (C) -- node[fill=white] {1} (H);
\draw (E) -- node[fill=white] {3} (F);
\draw (B) -- node[fill=white] {2} (F);
\draw (F) .. controls (8,5) .. node[fill=white] {5} (H);
\end{tikzpicture}
```

Lo de poner `[fill=white]` en los pesos de las aristas es para que nos pinte un fondo blanco que borre un trozo de arista y así se vea mejor el número pintado sobre ella.

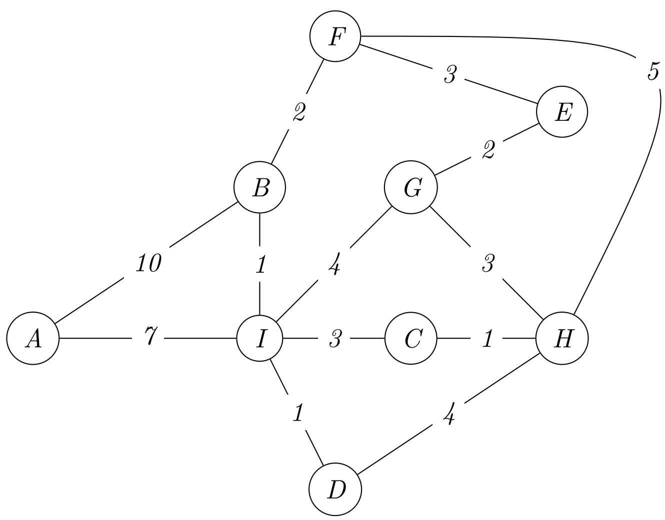
Definición 9. *Sea G un grafo, por ejemplo el anterior.*

- *Un* **camino cerrado** *o* **circuito** *es un camino que une un vértice v consigo mismo. Por ejemplo*
$$A - I - C - H - G - I - B - A$$
- *Un* **ciclo** *es un camino cerrado de longitud mayor que 0 que no repite vértices ni aristas. Por ejemplo*
$$I - G - E - F - B - I$$
- *Para pintar los circuitos o ciclos en TiKZ se utilizará*
  `\draw (I) -- (G) -- (E) -- (F) -- (B) -- cycle;`
- *Aunque en notación matemática se pone el origen y el extremo del ciclo o circuito dos veces, en realidad sólo se pasa una vez por dicho vértice.*
- *Puesto que el camino está cerrado y volvemos al mismo punto, podemos escribir el ciclo o circuito empezando por cualquier vértice, así el ciclo $I - G - E - F - B - I$ es el mismo que el ciclo $F - B - I - G - E - F$.*
- *Un* **árbol** *es un grafo conexo y sin ciclos.*
- *Un grafo conexo es un árbol si y sólo si $|V| = |E| + 1$.*

- *También podemos ver que un grafo es un árbol si y solo si entre cada dos vértices existe uno y solo un camino.*
- *Llamaremos árbol generador de un grafo conexo a un subgrafo que sea un árbol y que mantenga la conexión entre todos los vértices. Cuando el grafo tenga pesos, diremos que el árbol generador es de peso mínimo o minimal si es el que tiene menos de entre todos los posibles, entendiendo como peso total la suma de los pesos de todas las aristas que lo forman.*

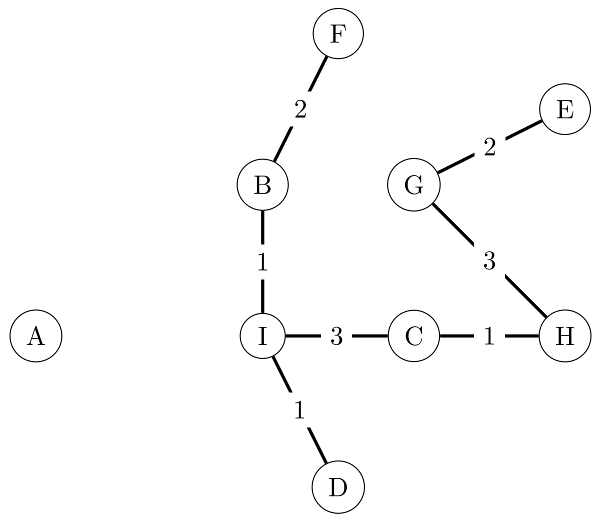
Para calcular el árbol generador minimal hay dos algoritmos, el de Kruskal y el de Prim. Vamos a verlos con unos ejemplos:

Ejercicio 10. *Calcula el árbol generador minimal del siguiente grafo usando el algoritmo Kruskal.*



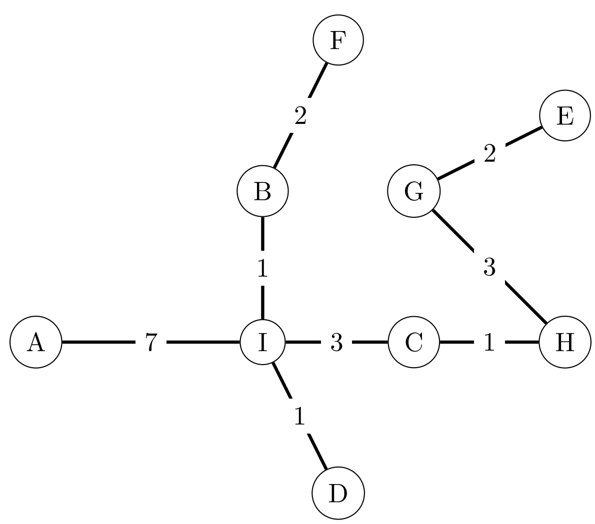
Solución:
Vamos a ordenar las aristas de menor a mayor peso. Tenemos las siguientes aristas:

(1) Las de peso 1 son BI, ID y CH.
(2) Las de peso 2 son GE y BF.
(3) Las de peso 3 son IC, GH y EF.
(4) Las de peso 4 son HD e IG.
(5) La de peso 5 es FH.
(6) La de peso 7 es AI.
(7) La de peso 10 es AB.



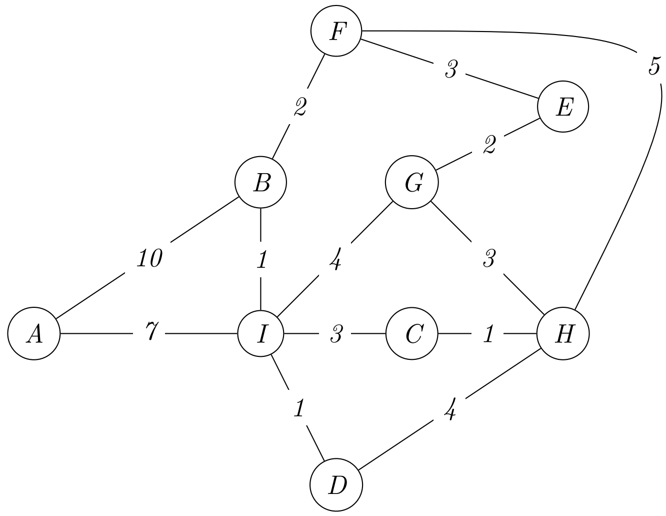
Hemos puesto todas las aristas de menor a mayor peso, hasta que nos encontramos con la arista EF que no debemos poner porque esos vértices ya están conectados. Nos pasa lo mismo con HD, IG, FH.

Cuando llegamos a la arista AI vemos que esa sí es necesaria para conectar el vértice A



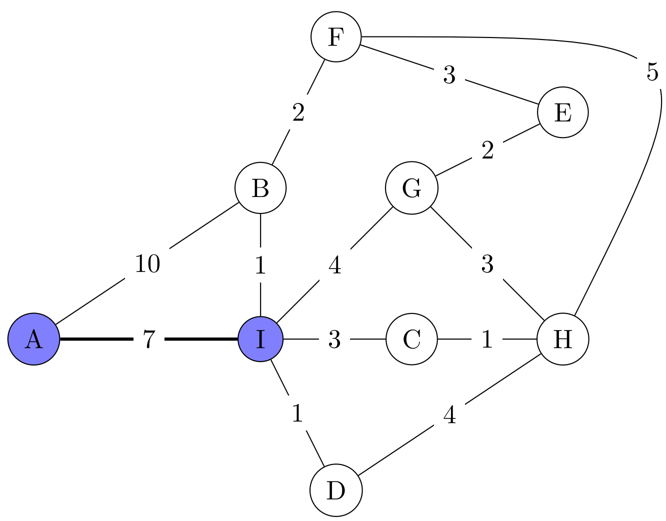
Como tenemos todos los vértices conectados, el algoritmo ha terminado.

Ejercicio 11. *Calcula el árbol generador minimal del siguiente grafo usando el algoritmo de Prim.*

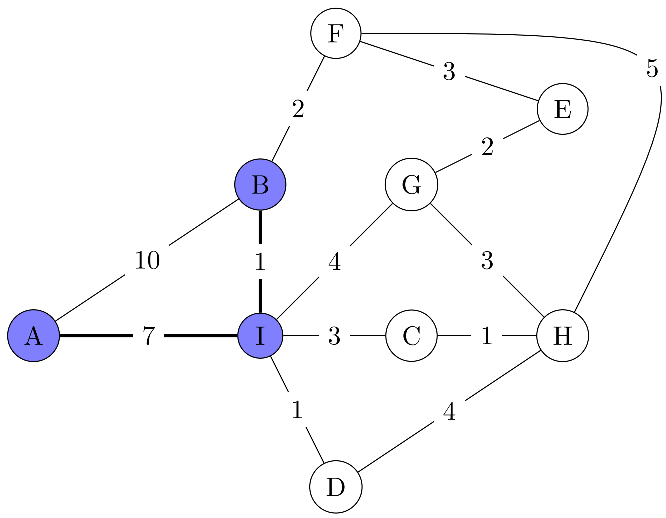


Solución:

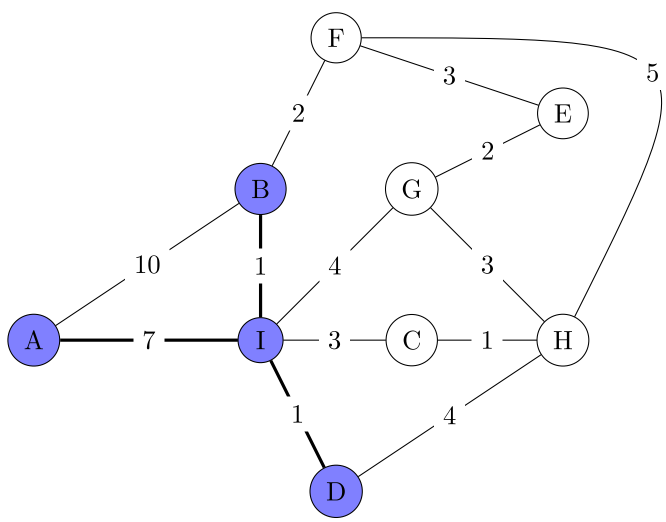
Para aplicar el algoritmo de Prim partimos de cualquier vértice, por ejemplo el vértice A que pintaremos de color azul. Para salir de la zona azul tenemos dos opciones, una de coste 7 y otra de coste 10. Elegiremos la más barata que nos llevará al vértice I.



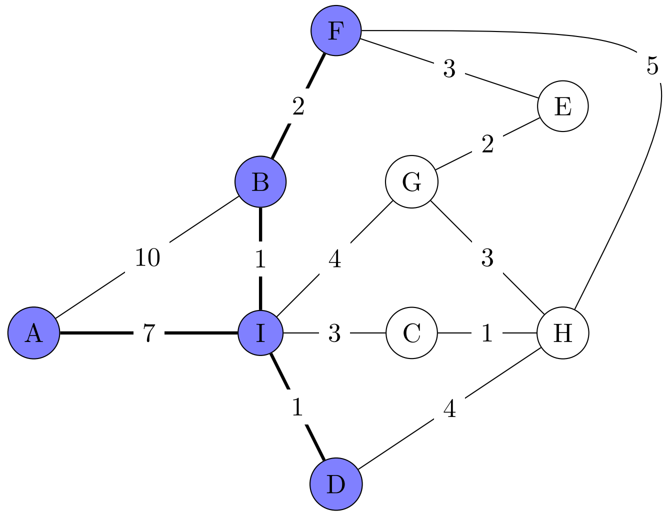
Para salir de la zona marcada en azul vamos a utilizar una arista de coste 1, por ejemplo la que nos une a B.



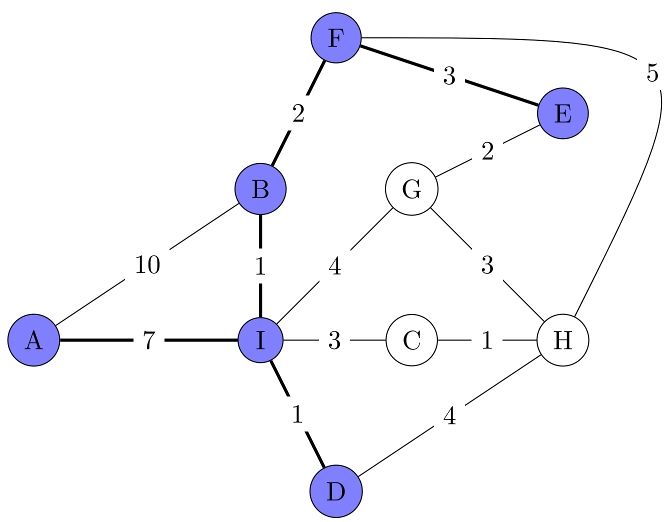
Para salir de la zona azul utilizaremos la arista de coste 1 entre I y D.



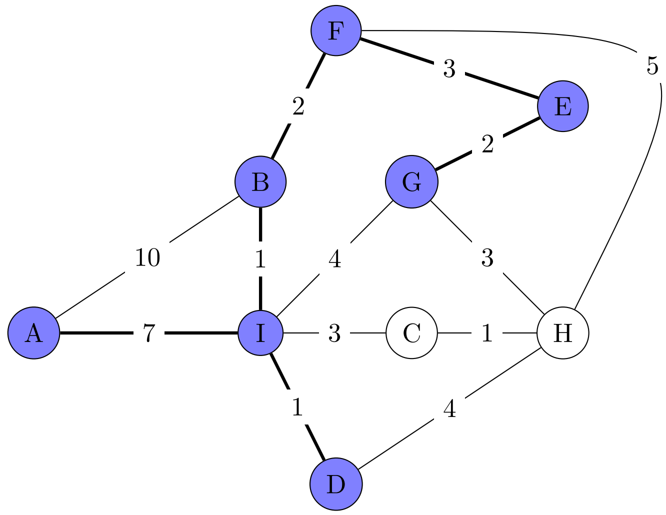
Ahora saldremos por la arista de coste 2 que nos lleva a F.



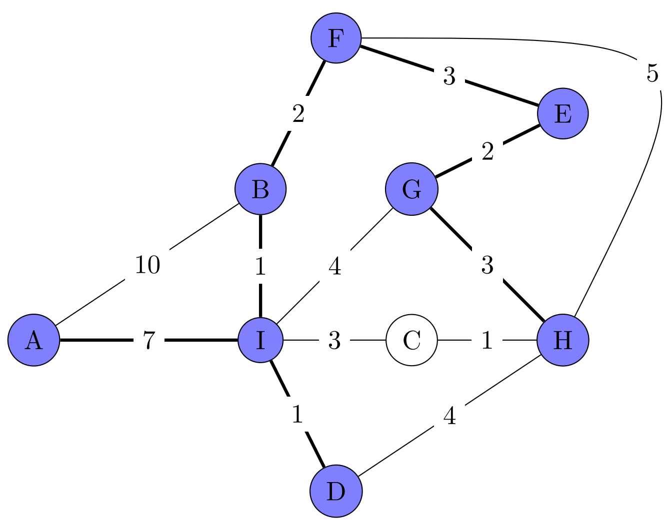
Ahora elegimos la arista de coste 3 que nos lleva al vértice E.



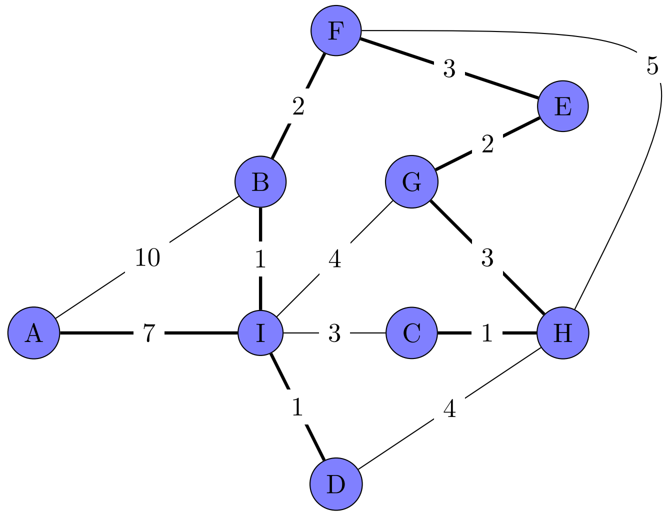
Ahora iremos al vértice G usando una arista de coste 2.



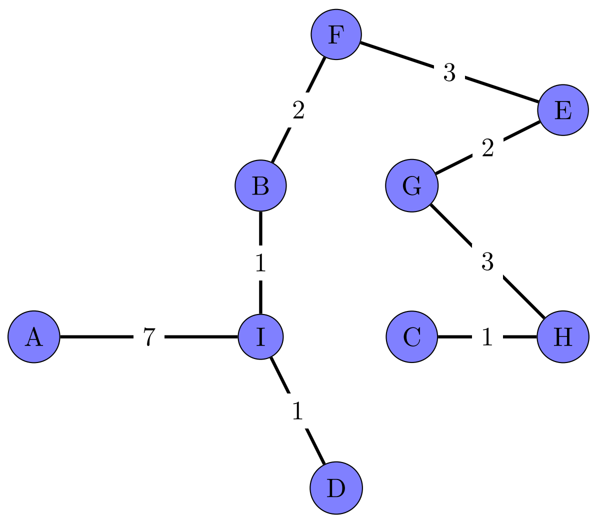
Ahora iremos al vértice H usando la arista GH de coste 3.



Ahora iremos al vértice C usando la arista HC de coste 1.



Eso nos une todos los vértices con un árbol generador minimal



Definición 12. *Un grafo se dice* **euleriano** *si existe un circuito que recorre todas las aristas pasando una única vez por cada una de ellas.*

Existe un teorema que nos caracteriza cuando un grafo es euleriano.

Teorema 13. *Un grafo es euleriano si es conexo y todos sus vértices son de grado par.*

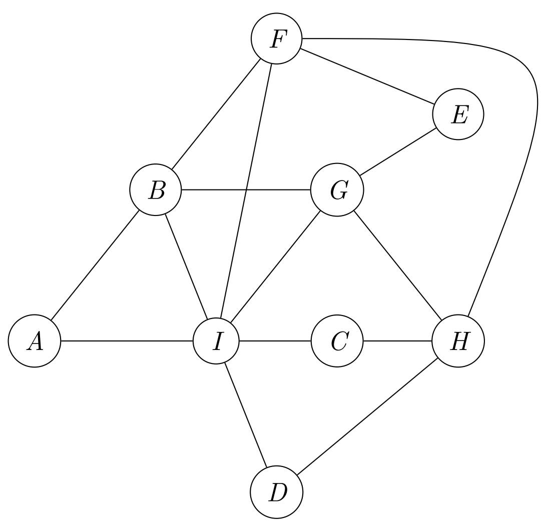
También podemos considerar el problema de que exista un camino entre dos vértices distintos u y v que recorra todas las aristas y pase una única vez por cada una de ellas. Estos grafos no son eulerianos (el camino no es cerrado) pero a este tipo de caminos los llamaremos caminos abiertos eulerianos. Se pueden caraterizar fácilmente puesto que si unimos u y v con una arista extra tendríamos un grafo propiamente euleriano. La caracterización es la siguiente:

Teorema 14. *Un grafo tiene un camino abierto euleriano si y solo si es conexo y tiene todos sus vértices de grado par salvo dos de grado impar. El camino abierto irá desde uno de los vértices de grado impar hasta el otro.*

Para encontrar este camino abierto, pondríamos una arista "ficticia" entre u y v, calcularíamos el camino cerrado euleriano y luego romperíamos el circuito quitando la arista uv que habíamos introducido.

Lo que vamos a ver ahora es cómo encontrar el circuito euleriano mediante un ejemplo:

Ejercicio 15. *Determina si el siguiente grafo es Euleriano y en caso de serlo encuentra un circuito euleriano.*

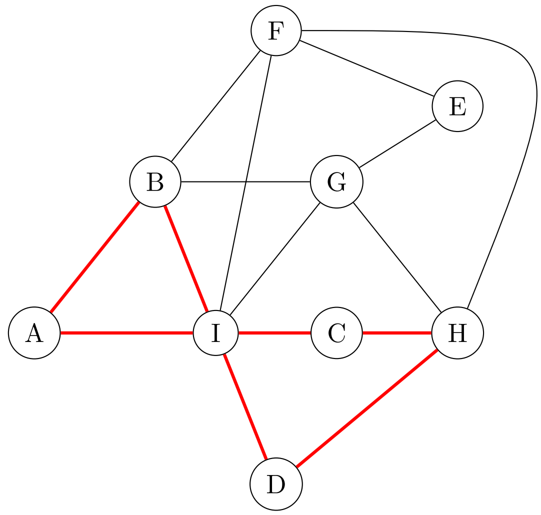


Solución:

Como el grafo es conexo y todos los vértices tienen grado par, el grafo es euleriano.

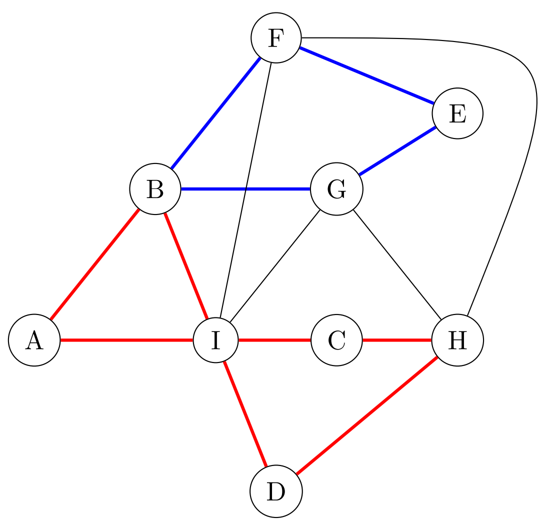
Para encontrar el circuito euleriano, empezamos por cualquier vértice y vamos recorriendo las aristas hasta que nos quedemos parados, necesariamente en el vértice de origen. Por ejemplo, hacemos el recorrido

$$(A) - (B) - (I) - (C) - (H) - (D) - (I) - (A).$$



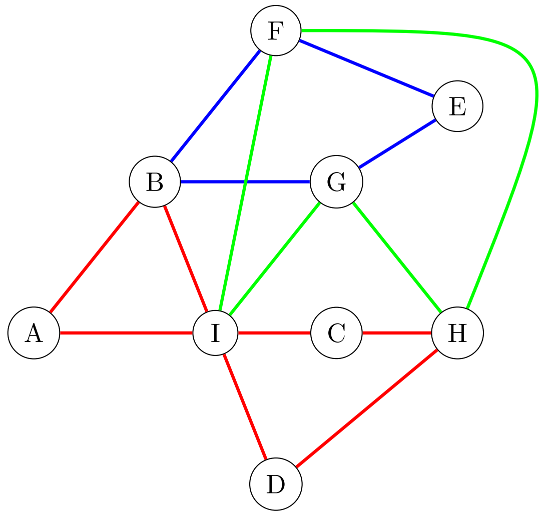
Partimos ahora de cualquier otro vértice que tenga aristas disponibles. Hacemos el recorrido

$$(B) - (G) - (E) - (F) - (B).$$



Partimos ahora de cualquier otro vértice que tenga aristas disponibles, por ejemplo F y hacemos el recorrido:

$$(F) - (H) - (G) - (I) - (F).$$



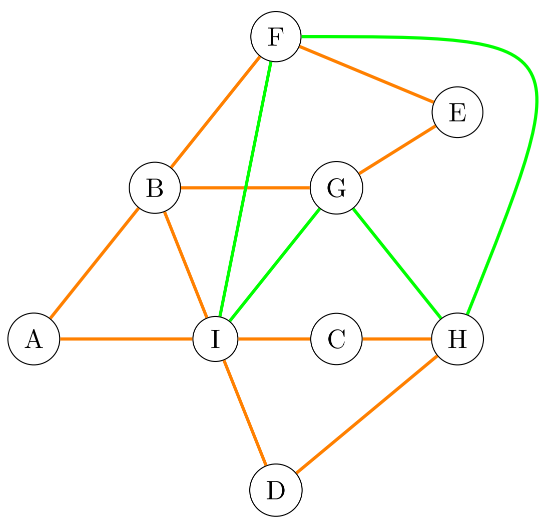
Para unir el circuito rojo con el azul utilizamos como inicio del recorrido el vértice común B, con lo que tenemos:

$$(B)-(I)-(C)-(H)-(D)-(I)-(A)-(B)$$

$$(B)-(G)-(E)-(F)-(B)$$

que unidos nos dan

$$(B)-(I)-(C)-(H)-(D)-(I)-(A)-(B)-(G)-(E)-(F)-(B)$$



Para pegar la parte naranja con la parte verde usamos el vértice común F.

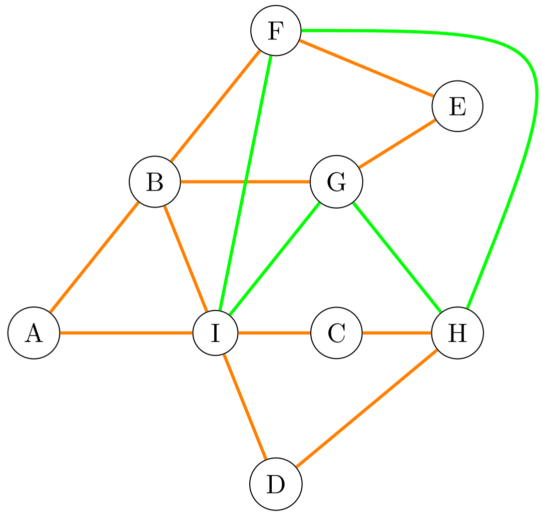
$$(F)-(B)-(I)-(C)-(H)-(D)-(I)-(A)-(B)-(G)-(E)-(F)$$

$$(F)-(H)-(G)-(I)-(F).$$

Los unimos y obtenemos

$$(F)-(B)-(I)-(C)-(H)-(D)-(I)-(A)-(B)-(G)-(E)-$$

$$(F)-(H)-(G)-(I)-(F).$$

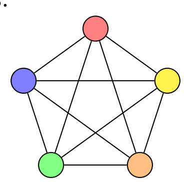


El resultado final es

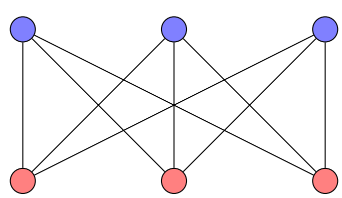
$$(F)-(B)-(I)-(C)-(H)-(D)-(I)-$$
$$(A)-(B)-(G)-(E)-(F)-(H)-(G)-(I)-(F).$$

Definición 16. *Sea $G=(V,E)$ un grafo y C un conjunto de colores, llamaremos* **coloración del grafo** *G a una aplicación $c: V \to C$ que a cada vértice le asigna un color de tal forma que para toda arista $\{u,v\}$ de E, los colores de los extremos son distintos, es decir $c(u) \neq c(v)$.*

Por ejemplo, para pintar el grafo K_5 tenemos que usar 5 colores porque tenemos aristas uniendo todos los vértices:



Pero también podemos tener grafos con muchos vértices que se puedan pintar con sólo dos colores. Por ejemplo, los grafos completos $K_{r,s}$ son los formados por dos conjuntos, uno con r elementos y otro con s elementos en los que tengamos todas las conexiones posibles entre los elementos del primer conjunto y los del segundo. Por ejemplo, $K_{3,3}$ tendría esta forma



Para describir estos colores en TiKZ crearemos estilos diferentes para cada color indicando el color de fondo que queremos poner. Así, por ejemplo el grafo anterior se escribiría:

```
\begin{tikzpicture}[
rojo/.style = {fill=red!50,circle,draw},
```

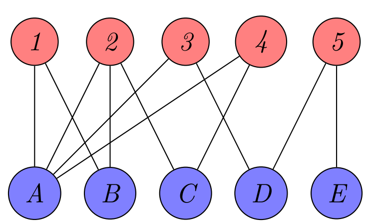
```
azul/.style = {fill=blue!50,circle,draw}
]
\node[rojo] (A) at (0,0) {};
\node[rojo] (B) at (2,0) {};
\node[rojo] (C) at (4,0) {};
\node[azul] (1) at (0,2) {};
\node[azul] (2) at (2,2) {};
\node[azul] (3) at (4,2) {};

\draw (A) -- (1);
\draw (A) -- (2);
\draw (A) -- (3);
\draw (B) -- (1);
\draw (B) -- (2);
\draw (B) -- (3);
\draw (C) -- (1);
\draw (C) -- (2);
\draw (C) -- (3);
\end{tikzpicture}
```

Existe un teorema que nos caracteriza cuando un grafo se puede pintar con únicamente dos colores. A los grafos que se pueden pintar con sólo dos colores se les llama **bipartidos** o **bicoloreables**.

Teorema 17. *Un grafo es bipartido si y solo si no tiene ningún ciclo de longitud impar.*

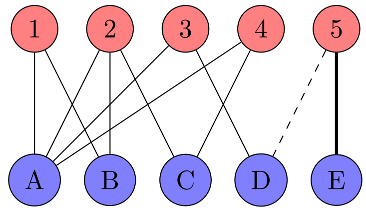
Los grafos bipartidos nos permiten representar problemas de elección. Dichos problemas consisten en que tenemos dos conjuntos A y B de tal forma que los elementos del conjunto A tienen ciertas preferencias para elegir elementos del conjunto B. El objetivo es ver si es posible asignar a cada elemento del conjunto A un elemento del conjunto B de entre sus preferencias. Vamos a ver cómo se hace en los siguientes ejemplos:

Ejercicio 18. *Determina si existe un emparejamiento posible en el siguiente grafo y cálculalo en caso de existir.*

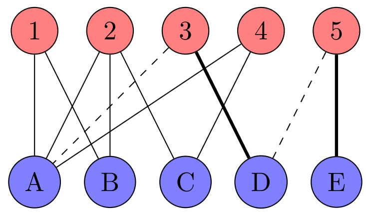


Solución:

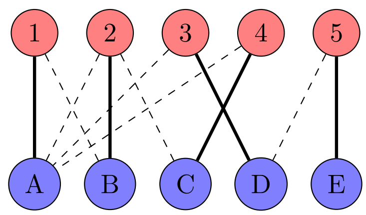
Para que el grafo tenga un emparejamiento, necesariamente E tiene que estar emparejado con 5 puesto que es su única elección posible.



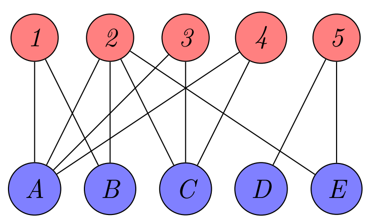
Como 5 no puede elegir ya a D, necesariamente este tendrá que venir elegido por 3.



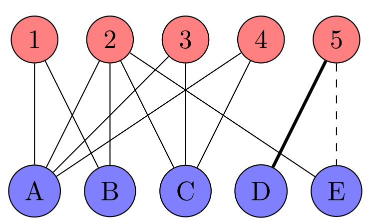
Completamos la elección dando posibles valores a 1, 2 y 4.



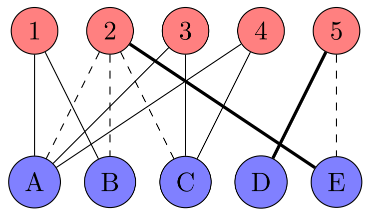
Ejercicio 19. *Determina si existe un emparejamiento posible en el siguiente grafo y cálculalo en caso de existir.*



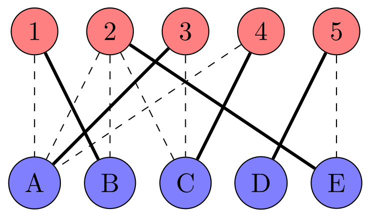
Solución:
La única elección posible para D es 5.



La única elección posible para E ahora es 2.



La única elección posible para B ahora es 1. Los dos vértices restantes, los elegimos entre sus dos opciones posibles.



La solución de un problema de emparejamiento no siempre existe. Hay situaciones en las cuales eso no es posible, por ejemplo si dos elementos del conjunto A quieren como única elección posible un mismo elemento del conjunto B. Lo mismo sucede si 3 elementos de A se disputaran únicamente dos elementos de B. En general se puede probar que el problema de emparejamiento tendrá solución si solo si se da la **condición de la diversidad** que nos dice que para todo k y todo subconjunto de k elementos del conjunto A, existen al menos k elementos seleccionados por ellos. Para demostrar que el emparejamiento no es posible, tendremos pues que encontrar k elementos de A que elijan entre ellos menos de k elementos del conjunto B. Eso producirá una situación irresoluble que nos garantizará que la elección no es posible.