Recorridos y conectividad

J. A. Rodríguez-Velázquez & A. Estrada-Moreno

URV

Definición

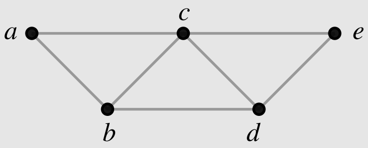
Sea $G=(V,E)$ un grafo.

- Un **recorrido** (walk) es una secuencia de vértices $v_1, v_2, \ldots, v_k$ tal que $\{v_i, v_{i+1}\} \in E$. Se denota $v_1 - v_k$ recorrido.
- Un $u-v$ recorrido es **cerrado** si $u=v$; en caso contrario se dice que es **abierto**.
- La **longitud de un recorrido** R, $\ell(R)$, es el número de aristas que lo componen.

Definición

Sea $G = (V,E)$ un grafo.

- Un **recorrido** (walk) es una secuencia de vértices $v_1, v_2, \ldots, v_k$ tal que $\{v_i, v_{i+1}\} \in E$. Se denota $v_1 - v_k$ recorrido.
- Un $u - v$ recorrido es **cerrado** si $u = v$; en caso contrario se dice que es **abierto**.
- La **longitud de un recorrido** R, $\ell(R)$, es el número de aristas que lo componen.

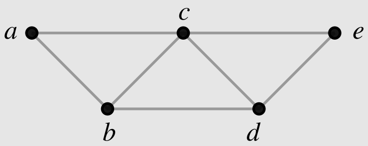


$a,b,a,b,d,e \longrightarrow$ es un recorrido abierto
$a,b,c,d,e,c,a \longrightarrow$ es un recorrido cerrado

Definición

Un recorrido es un **itinerario** si todas las aristas son diferentes. Se pueden destacar los siguientes tipos de itinerario:

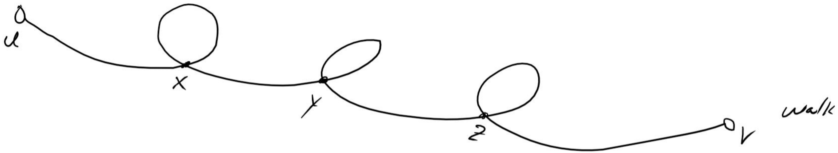
- Un **camino** (path), si no se repiten vértices.
- Un **circuito**, si es cerrado.
- Un **ciclo** es un circuito (cerrado) que, eliminando el primer vértice, también es un camino (no repite vértices).



$a, b, c, d, e \longrightarrow$ es un camino
$a, b, d, c, a \longrightarrow$ es un ciclo
$a, b, c, d, e, c, a \longrightarrow$ es un circuito, pero no es un ciclo

Proposición

Dados dos vértices diferentes u, v de un grafo $G = (V, E)$, todo $u - v$ recorrido contiene un camino.



$u - x - y - z - v$ = path.

Proposición

Si en un grafo $G = (V, E)$ hay dos recorridos diferentes que conectan un par de vértices, entonces el grafo contiene algún ciclo.

Teorema

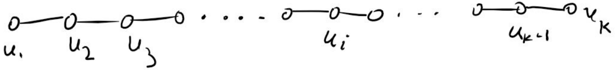
Si todos los vértices de un grafo $G=(V,E)$ son de grado superior o igual a 2, entonces el grafo contiene algún ciclo.

Teorema

Si todos los vértices de un grafo $G=(V,E)$ son de grado superior o igual a 2, entonces el grafo contiene algún ciclo.

Demostración

Sea $u_1,u_2,\ldots,u_k$ un camino de longitud máxima en G. Como u_k tiene grado mayor o igual que 2, existe al menos un vértice $a\neq u_{k-1}$ que es adyacente a u_k. Por la maximalidad de la longitud del camino seleccionado, se cumple que $a=u_i$ para algún $i\in\{1,\ldots,k-2\}$. Hemos obtenido el ciclo, $u_i,u_{i+1},\ldots,u_ku_i$.

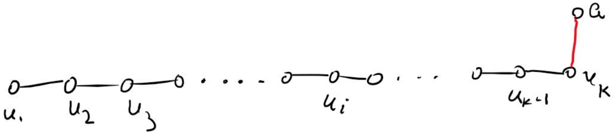


Teorema

Si todos los vértices de un grafo $G=(V,E)$ son de grado superior o igual a 2, entonces el grafo contiene algún ciclo.

Demostración

Sea $u_1,u_2,\ldots,u_k$ un camino de longitud máxima en G. Como u_k tiene grado mayor o igual que 2, existe al menos un vértice $a \neq u_{k-1}$ que es adyacente a u_k. Por la maximalidad de la longitud del camino seleccionado, se cumple que $a=u_i$ para algún $i \in \{1,\ldots,k-2\}$. Hemos obtenido el ciclo, $u_i,u_{i+1},\ldots,u_k u_i$.

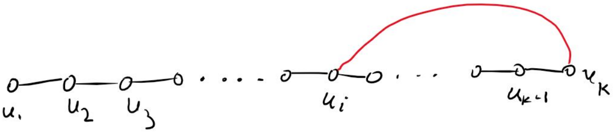


Teorema

Si todos los vértices de un grafo $G=(V,E)$ son de grado superior o igual a 2, entonces el grafo contiene algún ciclo.

Demostración

Sea $u_1,u_2,\ldots,u_k$ un camino de longitud máxima en G. Como u_k tiene grado mayor o igual que 2, existe al menos un vértice $a \neq u_{k-1}$ que es adyacente a u_k. Por la maximalidad de la longitud del camino seleccionado, se cumple que $a=u_i$ para algún $i \in \{1,\ldots,k-2\}$. Hemos obtenido el ciclo, $u_i,u_{i+1},\ldots,u_k u_i$.



Ejercicio

Determina si las siguientes afirmaciones son ciertas o falsas.

1. Todo grafo acíclico sin vértices aislados contiene vértices de grado 1.
2. No existen grafos acíclico tal que su complemento también sea acíclico.
3. Sea G un grafo de orden n y grado máximo Δ. Si $\Delta \leq n-3$, entonces G^c contiene al menos un ciclo.

Teorema

Sean v_i y v_j vértices de un grafo G. Sea A la matriz de adyacencia de G. Entonces el número de recorridos de longitud k en G, de v_i a v_j, es el elemento de la posición (i,j) de A^k.

Teorema

Sean v_i y v_j vértices de un grafo G. Sea A la matriz de adyacencia de G. Entonces el número de recorridos de longitud k en G, de v_i a v_j, es el elemento de la posición (i,j) de A^k.

Demostración

El resultado es cierto para $k=1$, ya que $A^1 = A(G)$. Supongamos que el resultado es cierto para $k \geq 1$. Sean v_h y v_j vértices adyacentes, entonces el número de recorridos de longitud $k+1$ de v_i a v_j y que contienen a v_h es $\left(A^k\right)_{ih} \cdot A_{hj}$. Por eso, el número de recorridos de longitud $k+1$ de v_i a v_j es

$$\sum_{v_h \sim v_j} \left(A^k\right)_{ih} \cdot A_{hj} = \sum_{h=1}^{n} \left(A^k\right)_{ih} \cdot A_{hj} = \left(A^{k+1}\right)_{ij}.$$

El resultado general se obtiene por inducción.

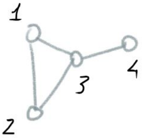
Ejercicio

¿Qué información sobre el grafo nos brinda la traza de A^2?

Ejercicio

¿Qué información sobre el grafo nos brinda la traza de A^2?

$$2m = Tr(A^2).$$



$$A = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \qquad A^2 = \begin{pmatrix} 2 & 1 & 1 & 1 \\ 1 & 2 & 1 & 1 \\ 1 & 1 & 3 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}$$

Grafos conexos

Definición

Un grafo $G=(V,E)$ es **conexo** si para cada par de vértices u y v de G existe un $u-v$ camino.

Definición

Un grafo $G=(V,E)$ es **conexo** si para cada par de vértices u y v de G existe un $u-v$ camino.

Ejemplos

- Los siguientes grafos son conexos: K_n, P_n, $K_{r,s}$, C_n, P_k+C_r.
- Los siguientes grafos **no** son conexos: $K_n \cup K_{r,s}$, $(P_3+C_4)^c$, N_n $(n \geq 2)$.

Proposición

Para todo par de grafos G y H se cumple:

1. El grafo $G+H$ es conexo.
2. $G \odot H$ es conexo si y sólo si G es conexo.
3. $G \square H$ es conexo si y sólo si G y H son conexos.
4. $G \boxtimes H$ es conexo si y sólo si G y H son conexos.
5. Si G es conexo, entonces $L(G)$ es conexo.
6. $G \circ H$ es conexo si y solo si G es conexo.

Definición

En el conjunto de vértices V de un grafo $G = (V, E)$ se define la relación $\mathcal{R}_G$ de la siguiente forma:

- $v\mathcal{R}_G v$ para todo $v \in V$.
- Dados $u, v \in V$, $u\mathcal{R}_G v$ si y solo si existe un $u - v$ camino en G.

Definición

En el conjunto de vértices V de un grafo $G=(V,E)$ se define la relación $\mathcal{R}_G$ de la siguiente forma:

- $v\mathcal{R}_G v$ para todo $v \in V$.
- Dados $u, v \in V$, $u\mathcal{R}_G v$ si y solo si existe un $u-v$ camino en G.

Proposición

Para todo grafo G, la relación $\mathcal{R}_G$ es una relación de equivalencia.
En consecuencia, se establece una partición de $V = V_1 \cup \ldots \cup V_k$ donde dos vértices de una misma clase son mutuamente accesibles por algún camino y vértices de clases diferentes son inaccesibles.

Definición

En el conjunto de vértices V de un grafo $G=(V,E)$ se define la relación $\mathcal{R}_G$ de la siguiente forma:

- $v\mathcal{R}_G v$ para todo $v \in V$.
- Dados $u,v \in V$, $u\mathcal{R}_G v$ si y solo si existe un $u-v$ camino en G.

Proposición

Para todo grafo G, la relación $\mathcal{R}_G$ es una relación de equivalencia.
En consecuencia, se establece una partición de $V = V_1 \cup \ldots \cup V_k$ donde dos vértices de una misma clase son mutuamente accesibles por algún camino y vértices de clases diferentes son inaccesibles.

Definición

Las **componentes conexas** de un grafo G son los subgrafos $G_i = \langle V_i \rangle$ generados por cada una de las clases de equivalencia de G respecto a la relación $\mathcal{R}_G$. Así, todo grafo se puede expresar como unión de sus componentes conexas:

$$G = G_1 \cup \ldots \cup G_k$$

Ejercicio

Demuestra que un grafo no trivial G es conexo si y solo si 0, como valor propio de la matriz laplaciana de G, tiene multiplicidad 1.

Ejercicio

Demuestra que un grafo no trivial G es conexo si y solo si 0, como valor propio de la matriz laplaciana de G, tiene multiplicidad 1.

Demostración

($\Rightarrow$) Sea G conexo, $V(G) = \{1, \ldots, n\}$, y sea $B = (\overrightarrow{u_1}, \ldots, \overrightarrow{u_k})$ una base de $\ker(L(G))$.

Ejercicio

Demuestra que un grafo no trivial G es conexo si y solo si 0, como valor propio de la matriz laplaciana de G, tiene multiplicidad 1.

Demostración

($\Rightarrow$) Sea G conexo, $V(G) = \{1, \ldots, n\}$, y sea $B = (\overrightarrow{u_1}, \ldots, \overrightarrow{u_k})$ una base de $\ker(L(G))$. Podemos asumir que cada uno de estos vectores está normalizado de forma que su mayor componente es 1. En particular, sea $\overrightarrow{u_l} = (x_1, \ldots, x_n)$ tal que $x_j = \max\{x_i : \ i \in \{1, \ldots, n\}\} = 1$.

Ejercicio

Demuestra que un grafo no trivial G es conexo si y solo si 0, como valor propio de la matriz laplaciana de G, tiene multiplicidad 1.

Demostración

($\Rightarrow$) Sea G conexo, $V(G) = \{1, \ldots, n\}$, y sea $B = (\overrightarrow{u_1}, \ldots, \overrightarrow{u_k})$ una base de $\ker(L(G))$. Podemos asumir que cada uno de estos vectores está normalizado de forma que su mayor componente es 1. En particular, sea $\overrightarrow{u_l} = (x_1, \ldots, x_n)$ tal que $x_j = \text{máx}\{x_i : i \in \{1, \ldots, n\}\} = 1$. En este caso, $\delta(j) = \sum_{i \sim j} x_i$, lo que implica que $x_i = 1$ para todo $i \sim j$.

Ejercicio

Demuestra que un grafo no trivial G es conexo si y solo si 0, como valor propio de la matriz laplaciana de G, tiene multiplicidad 1.

Demostración

($\Rightarrow$) Sea G conexo, $V(G) = \{1,\ldots,n\}$, y sea $B = (\overrightarrow{u_1},\ldots,\overrightarrow{u_k})$ una base de $\ker(L(G))$. Podemos asumir que cada uno de estos vectores está normalizado de forma que su mayor componente es 1. En particular, sea $\overrightarrow{u_l} = (x_1,\ldots,x_n)$ tal que $x_j = \text{máx}\{x_i :\ i \in \{1,\ldots,n\}\} = 1$. En este caso, $\delta(j) = \sum_{i\sim j} x_i$, lo que implica que $x_i = 1$ para todo $i \sim j$. El mismo argumento es válido para concluir que $x_r = 1$ para todo $r \sim i \in N(j)$, y por la conectividad de G, se deduce que $\overrightarrow{u_l} = (1,1,\ldots,1)$ para todo vector de B.

Ejercicio

Demuestra que un grafo no trivial G es conexo si y solo si 0, como valor propio de la matriz laplaciana de G, tiene multiplicidad 1.

Demostración

($\Rightarrow$) Sea G conexo, $V(G) = \{1,\ldots,n\}$, y sea $B = (\overrightarrow{u_1},\ldots,\overrightarrow{u_k})$ una base de $\ker(L(G))$. Podemos asumir que cada uno de estos vectores está normalizado de forma que su mayor componente es 1. En particular, sea $\overrightarrow{u_l} = (x_1,\ldots,x_n)$ tal que $x_j = \text{máx}\{x_i : i \in \{1,\ldots,n\}\} = 1$. En este caso, $\delta(j) = \sum_{i\sim j} x_i$, lo que implica que $x_i = 1$ para todo $i \sim j$. El mismo argumento es válido para concluir que $x_r = 1$ para todo $r \sim i \in N(j)$, y por la conectividad de G, se deduce que $\overrightarrow{u_l} = (1,1,\ldots,1)$ para todo vector de B. Por lo tanto $k = \dim(\ker(L(G))) = 1$.

Ejercicio

Demuestra que un grafo no trivial G es conexo si y solo si 0, como valor propio de la matriz laplaciana de G, tiene multiplicidad 1.

Demostración

($\Rightarrow$) Sea G conexo, $V(G) = \{1, \ldots, n\}$, y sea $B = (\overrightarrow{u_1}, \ldots, \overrightarrow{u_k})$ una base de $\ker(L(G))$. Podemos asumir que cada uno de estos vectores está normalizado de forma que su mayor componente es 1. En particular, sea $\overrightarrow{u_l} = (x_1, \ldots, x_n)$ tal que $x_j = \text{máx}\{x_i : \ i \in \{1, \ldots, n\}\} = 1$. En este caso, $\delta(j) = \sum_{i \sim j} x_i$, lo que implica que $x_i = 1$ para todo $i \sim j$. El mismo argumento es válido para concluir que $x_r = 1$ para todo $r \sim i \in N(j)$, y por la conectividad de G, se deduce que $\overrightarrow{u_l} = (1, 1, \ldots, 1)$ para todo vector de B. Por lo tanto $k = \dim(\ker(L(G))) = 1$.

—

($\Leftarrow$) Si $G_1, \ldots, G_k$ son las componentes conexas de G, entonces hay k vectores propios $\overrightarrow{u_1}, \ldots, \overrightarrow{u_k}$, asociados al valor propio 0, que son linealmente independientes y que se definen como $\overrightarrow{u_l} = (x_1^l, \ldots, x_n^l)$ donde $x_i^l = 1$ si $i \in V(G_l)$, mientras $x_i^l = 0$ para $i \notin V(G_l)$. Por lo tanto, si $k \neq 1$, entonces $\dim(\ker(L(G))) \neq 1$. $\square$

Proposición

Un grafo no trivial G tiene k componentes conexas si y solo si 0, como valor propio de la matriz laplaciana de G, tiene multiplicidad k.

Ejercicio

Todos los grafos con secuencia de grados $4, 4, 3, 3, 3, 3, 3, 3$ son conexos.

Ejercicio

Todos los grafos con secuencia de grados $4,4,3,3,3,3,3,3$ son conexos.

Solución:

Supongamos que existe algún grafo G no conexo con esta secuencia de grados.

–

Nótese que G tiene orden $n=8$. Como hay vértices de grado 4, alguna componente conexa G_i tendrá orden $n_i \geq 5$ y como el grado mínimo es 3, cualquier otra componente conexa G_j tendrá orden $n_j \geq 4$.

–

Así, $n \geq n_i + n_j \geq 9 > 8 = n$, lo que es una contradicción.

Proposición

Si G es un grafo conexo de orden n y medida m, entonces

$$m \geq n - 1.$$

Proposición

Si G es un grafo conexo de orden n y medida m, entonces

$$m \geq n-1.$$

Demostración

Si $n=1$, entonces el resultado es inmediato.

Proposición

Si G es un grafo conexo de orden n y medida m, entonces

$$m \geq n - 1.$$

Demostración

Si $n = 1$, entonces el resultado es inmediato.
Sea $G = (V, E)$ un grafo conexo de orden $n(G)$ y medida $m(G)$. Supongamos que la propiedad es cierta para todo grafo conexo G' de orden $n(G') = n(G) - 1 \geq 1$.

Proposición

Si G es un grafo conexo de orden n y medida m, entonces

$$m \geq n-1.$$

Demostración

Si $n=1$, entonces el resultado es inmediato.
Sea $G=(V,E)$ un grafo conexo de orden $n(G)$ y medida $m(G)$. Supongamos que la propiedad es cierta para todo grafo conexo G' de orden $n(G')=n(G)-1 \geq 1$.
En caso de que G tenga un vértice v de grado 1, el grafo $G'=G-v$ es un grafo conexo de orden $n(G')=n(G)-1$ y medida $m(G')=m(G)-1$.

Proposición

Si G es un grafo conexo de orden n y medida m, entonces

$$m \geq n-1.$$

Demostración

Si $n=1$, entonces el resultado es inmediato.
Sea $G=(V,E)$ un grafo conexo de orden $n(G)$ y medida $m(G)$. Supongamos que la propiedad es cierta para todo grafo conexo G' de orden $n(G')=n(G)-1 \geq 1$. En caso de que G tenga un vértice v de grado 1, el grafo $G'=G-v$ es un grafo conexo de orden $n(G')=n(G)-1$ y medida $m(G')=m(G)-1$. Aplicando la hipótesis de inducción al grafo G' se tiene que
$m(G)-1=m(G') \geq n(G')-1=n(G)-2$, de donde resulta que $m(G) \geq n(G)-1$, como se quería probar.

Proposición

Si G es un grafo conexo de orden n y medida m, entonces

$$m \geq n-1.$$

Demostración

Si $n=1$, entonces el resultado es inmediato.
Sea $G=(V,E)$ un grafo conexo de orden $n(G)$ y medida $m(G)$. Supongamos que la propiedad es cierta para todo grafo conexo G' de orden $n(G')=n(G)-1 \geq 1$.
En caso de que G tenga un vértice v de grado 1, el grafo $G'=G-v$ es un grafo conexo de orden $n(G')=n(G)-1$ y medida $m(G')=m(G)-1$. Aplicando la hipótesis de inducción al grafo G' se tiene que
$m(G)-1=m(G')\geq n(G')-1=n(G)-2$, de donde resulta que $m(G)\geq n(G)-1$, como se quería probar.
Si todos los vértices de G tienen grado mayor o igual que 2, entonces se aplica la fórmula de los grados, $2m(G)=\sum_{v\in V}\delta(v)\geq 2n(G)$ y, por lo tanto,
$m(G)\geq n(G)>n(G)-1$.

Ejercicio

Demuestra que ningún grafo conexo tiene la secuencia de grados $3, 3, 2, 1, 1, 1, 1, 1, 1$.

Ejercicio

Demuestra que ningún grafo conexo tiene la secuencia de grados $3,3,2,1,1,1,1,1,1$.

Solución

En primer lugar se puede verificar, si se utiliza el algoritmo de Havel-Hakimi, que la secuencia $3,3,2,1,1,1,1,1,1$ es gráfica.

Sea G un grafo con esta secuencia de grados.
Como G tiene orden $n=9$, si fuese conexo tendría medida $m \geq 8$. Pero, por la fórmula de los grados, $m=\frac{1}{2}(3+3+2+1+1+1+1+1+1)=7$. Por lo tanto, G no es conexo.

Árbol Generador

Definición

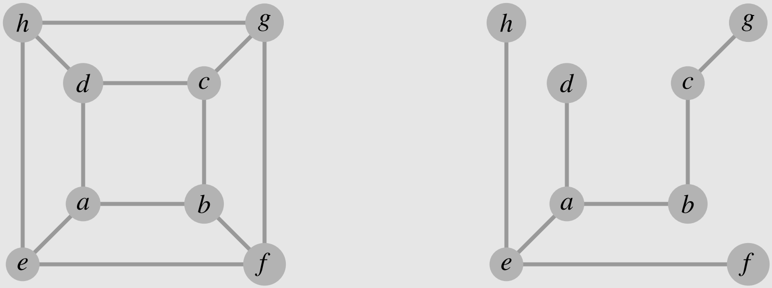
- Un **árbol** T, es un grafo que cumple que entre dos vértices cualesquiera hay un único camino que los conecta.
- Un **árbol generador (spanning tree)** de un grafo es un subgrafo generador que tiene estructura de árbol.

Definición

- Un **árbol** T, es un grafo que cumple que entre dos vértices cualesquiera hay un único camino que los conecta.
- Un **árbol generador (spanning tree)** de un grafo es un subgrafo generador que tiene estructura de árbol.

Ejemplo

El 3-cubo y un árbol generador.



Algoritmos de exploración de grafos

Test de conexión y árbol generador

- Algoritmo DFS
- Algoritmo BFS

Algoritmo DFS: Depth First Search

Estructuras necesarias para la formulación del algoritmo:

- Un grafo $G = (V,E)$ representado mediante una lista de adyacencias.
- Un conjunto P de los vértices que se han visitado, en el orden en que se ha hecho, que permita dar marcha atrás. La estructura de datos adecuada es la de una pila con las operaciones habituales: *apilar*(P,v), *desapilar*(P,v), *cima*(P).
- Una tabla de vértices ($estado$) que registra los vértices que se van visitando.
- Una lista R que contiene los vértices visitados hasta el momento (y que finalmente coincidirá con el conjunto de todos los vértices).

Algoritmo DFS

Entrada: $G(V, E)$, $v \in V$

Salida: R, vértices visitados

Algoritmo DFS(G, v)
  Inicio
    $P \leftarrow \emptyset$
    $R \leftarrow [v]$
    para $w \in V$
        $estado\ [w] \leftarrow 0$
    finpara
    $estado\ [v] \leftarrow 1$
    $apilar\ (P, v)$
    mientras $P \neq \emptyset$
        $w \leftarrow cima\ (P)$
        si w es adyacente a u con $estado\ [u] = 0$
            entonces $apilar\ (P, u)$
                $estado\ [u] \leftarrow 1$
                $añadir\ (R, u)$
            sino $desapilar\ (P)$
        finsi
    finmientras
    retorno (R)
  fin

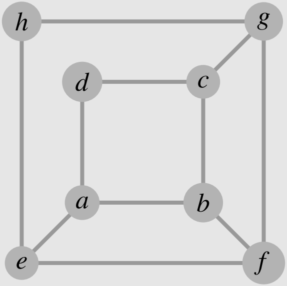
Aplicaciones DFS

Entre las muchas aplicaciones que tiene el DFS podemos nombrar las siguientes:

- Determinar si un grafo es conexo.
- Determinar ciclos en un grafo.
- Ordenamiento topológico en un grafo acíclico dirigido (digrafo acíclico).
- Encontrar aristas de corte (o puentes) en un grafo.
- Resolver puzzles con una sola solución, como los laberintos.
- Hallar las componentes fuertemente conexas (Algoritmo de Kosaraju, Algoritmo de Tarjan) en un digrafo.
- Encontrar el árbol generador en un grafo.
- Encontrar el recorrido en un circuito o camino euleriano.

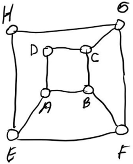
Ejemplo

Considera el grafo representado en la figura.

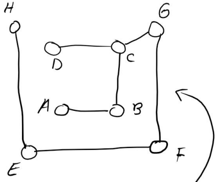


Aplica el algoritmo DFS partiendo del vértice a.

DFS

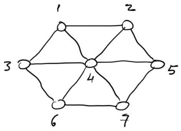


P	V.A	V.E	R
A	A	—	A
A B	B	—	A B
A B C	C	—	A B C
A B C D	D	—	A B C D
A B C	–	D	" "
A B C G	G	–	A B C D G
A B C G F	F	–	A B C D G F
A B C G F E	E	–	A B C D G F E
A B C G F E H	H	–	A B C D G F E H
A B C G F E	–	H	" "
A B C G F	–	E	" "
A B C G	–	F	" "
A B C	–	G	" "
A B	–	C	" "
A	–	B	" "
ϕ	–	A	" "



Árbol generador

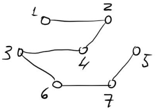
109



DFS

P	V.A	V E	R
1	1	—	1
1 2	2	-	1, 2
1 2 4	4	—	1, 2, 4
1 2 4 3	3	—	1, 2, 4, 3
1 2 4 3 6	6	-	1, 2, 4, 3, 6
1 2 4 3 6 7	7	—	1, 2, 4, 3, 6, 7
1 2 4 3 6 7 5	5	—	1, 2, 4, 3, 6, 7, 5
1 2 4 3 6 7	—	5	" "
1 2 4 3 6	—	7	" "
1 2 4 3	-	6	" "
1 2 4	-	3	" "
1 2	-	4	" "
1	-	2	" "
ϕ	-	1	" "

$R = [1, 2, 4, 3, 6, 7, 5]$



Árbol generador

Algoritmo BFS: Breadth First Search

Estructuras necesarias para la formulación del algoritmo:

- Un grafo $G = (V, E)$ representado mediante una lista de adyacencias.
- Un conjunto Q de los vértices que se han visitado, en el orden en que se ha hecho. La estructura de datos adecuada es la de una cola con las operaciones habituales: *añadir*(Q, v), *eliminar*(Q, v), *primero*(Q).
- Una tabla de vértices ($estado$) que registra los vértices que se van visitando.
- Una lista R que contiene los vértices visitados hasta el momento (y que finalmente coincidirá con el conjunto de todos los vértices).

Algoritmo BFS
Entrada: $G(V, E)$, $v \in V$
Salida: R, vértices visitados

Algoritmo BFS(G, v)
inicio
  $Q \leftarrow \emptyset$
  $R \leftarrow [v]$
  para $w \in V$
    estado $[w] \leftarrow 0$
  finpara
  estado $[v] \leftarrow 1$
  añadir (Q, v)
  mientras $Q \neq \emptyset$
    $w \leftarrow$ primero (Q)
    para u adyacente a w
      si *estado* $[u] = 0$
        entonces *añadir* (Q, u)
          estado $[u] \leftarrow 1$
          añadir (R, u)
      finsi
    finpara
    eliminar (Q)
  finmientras
  retorno (R)
fin

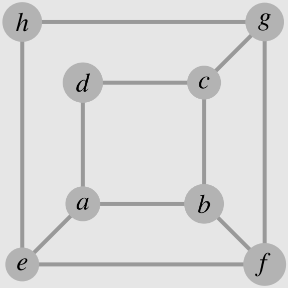
Aplicaciones BFS

Entre las muchas aplicaciones que tiene el BFS podemos nombrar las siguientes:

- Encontrar el camino más corto entre 2 vértices, medido por el número de aristas que contiene el camino.
- Probar si un grafo es bipartito.
- Encontrar el árbol generador mínimo en un grafo no ponderado.
- Hacer un rastreador web (Web Crawler)
- Sistemas de navegación GPS para ayudar a encontrar todas las ubicaciones vecinas desde la ubicación principal o de origen.
- Transmisión en red: Un paquete difundido es guiado por el algoritmo BFS para encontrar y llegar a todos los vértices para los que tiene la dirección.
- Redes P2P: BFS se puede implementar para localizar todos los nearest o nodos vecinos en una red peer to peer. Esto encontrará los datos requeridos más rápido.

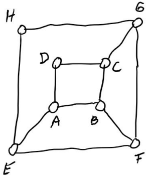
Ejemplo

Considera el grafo representado en la figura.

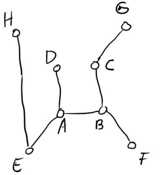


Aplica el algoritmo BFS partiendo del vértice a.

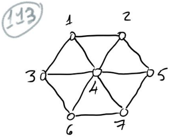
BFS



Q	V.A	V.E	R
A	A	–	A
A B	B	–	A B
A B D	D	–	A B D
A B D E	E	–	A B D E
B D E	–	A	A B D E
B D E C	C	–	A B D E C
B D E C F	F	–	A B D E C F
D E C F	–	B	" "
E C F	–	D	" "
E C F H	H	–	A B D E C F H
C F H	–	E	" "
C F H G	G	–	A B D E C F H G
F H G	–	C	" "
H G	–	F	" "
G	–	H	" "
ϕ	–	G	" "



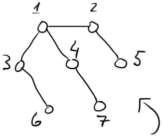
Árbol generador



BFS

Q	V.A	V.E	R
1	1	–	1
1 2	2	–	1, 2
1 2 3	3	–	1, 2, 3
1 2 3 4	4	–	1, 2, 3, 4
2 3 4	–	1	1, 2, 3, 4
2 3 4 5	5	–	1, 2, 3, 4, 5
3 4 5	–	2	" "
3 4 5 6	6	–	1, 2, 3, 4, 5, 6
4 5 6	–	3	" "
4 5 6 7	7	–	1, 2, 3, 4, 5, 6, 7
5 6 7	–	4	" "
6 7	–	5	" "
7	–	6	" "
ϕ	–	7	" "

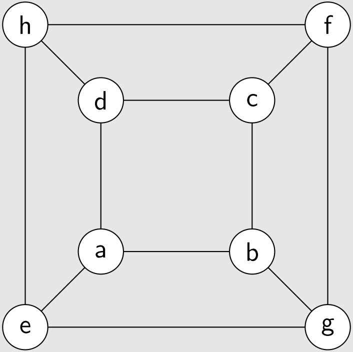
$R = [1, 2, 3, 4, 5, 6, 7]$



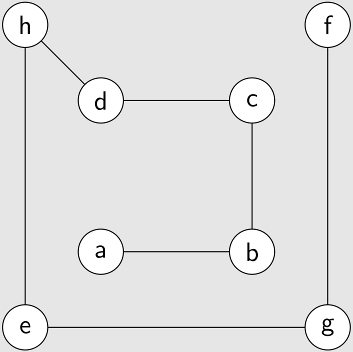
Árbol generador

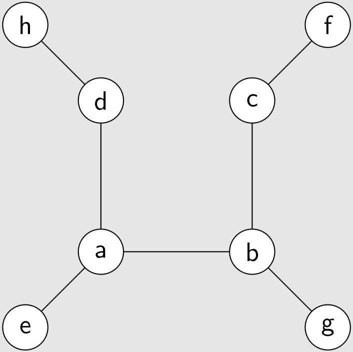
Ejercicio

Aplica los algoritmo DFS y BFS para determinar dos árboles generadores del 3-cubo partiendo del vértice a.



Solución





> El razonamiento matemático puede considerarse más bien esquemáticamente como el ejercicio de una combinación de dos instalaciones, que podemos llamar la intuición y el ingenio.
>
> Alan Turing