# RECORRIDOS DE GRAFOS.

El concepto de recorrido del grafo consiste en visitar todos los vértices del grafo sucesivamente de manera sistemática de manera que cada vértice se visite ***una única vez***.

En el recorrido de un grafo, existen dos tipos de vértices:

a) *Vértices visitados:* vértices ya visitados en el recorrido

b) *Vértices frontera:* vértices que aún no ha sido visitados, pero están conectados con algún vértice visitado (están pendientes de visitar).

Al igual que un árbol (de hecho, un árbol es un tipo de grafo orientado sin ciclos), un grafo puede ser recorrido en profundidad o en amplitud.

Existen dos diferencias fundamentales a la hora de recorrer un grafo respecto de un árbol:

a) Puesto que un árbol es un grafo dirigido acíciclo, al avanzar en el recorrido no cabe la posibilidad de que se vuelva a visitar un vértice ya visitado. En el recorrido de un grafo sí cabe la posibilidad de al avanzar visitar un vértice ya visitado, de modo que debe evitarse esta situación.

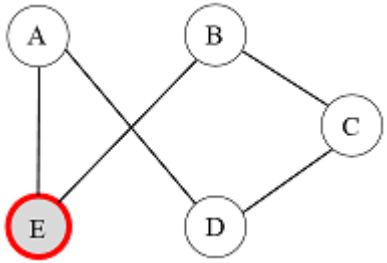
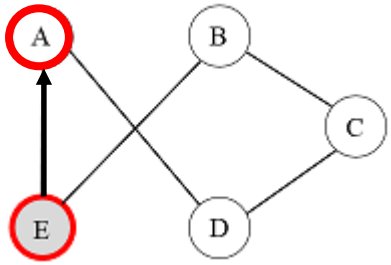
b) Partiendo de la raíz de un árbol se pueden visitar todos los vértices, mientras que en un grafo se puede dar la posibilidad de que no se alcancen todos los vértices desde un vértice. Habría que comenzar el recorrido en otro vértice para alcanzar todos los vértices.

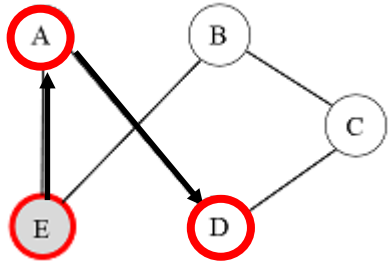
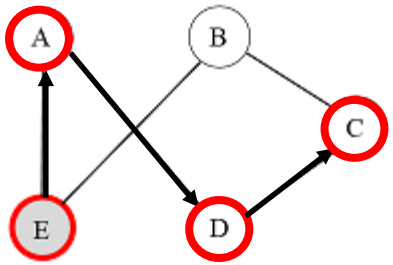
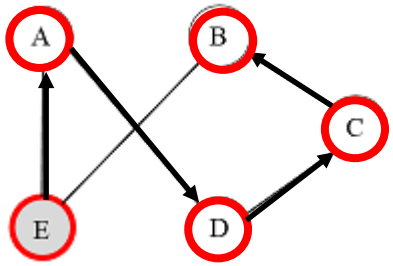
Cabe anotar que como ***no hay un nodo raíz***, es preciso fijar (de manera arbitraria ó según la situación) un origen para el recorrido.

**1. Recorrido en profundidad (Depth-First Search –DFS-).** A partir del nodo elegido como origen, avanzar a otro nodo no visitado y seguir el recorrido, de la misma manera, a partir de él.

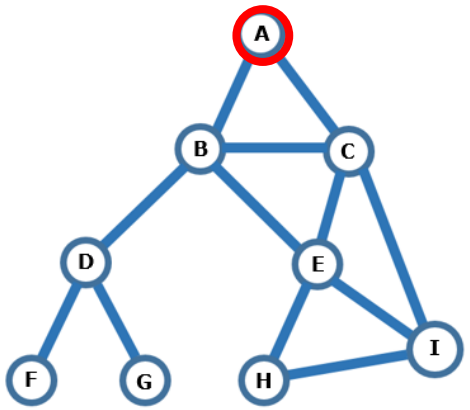
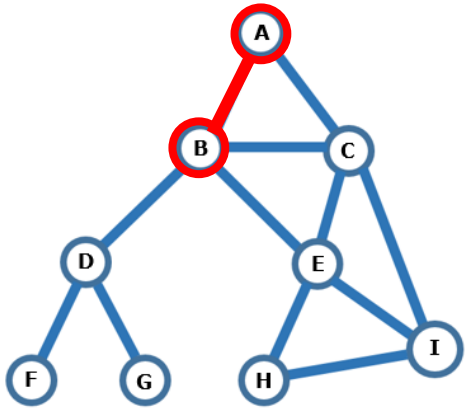
*Ejemplo 1.* Recorrido en profundidad en un grafo no dirigido.

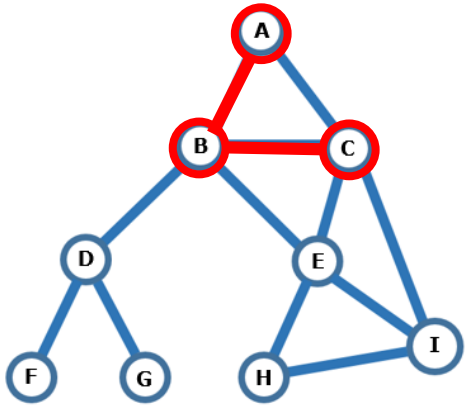
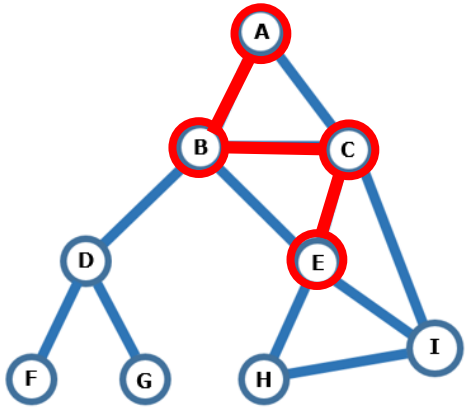
Formalmente se especifica el recorrido de la siguiente manera: Se puede tener una estructura como un arreglo en donde se indique que cada nodo del grafo está sin visitar, así se podrá llevar un seguimiento de cuáles nodos han sido visitados.

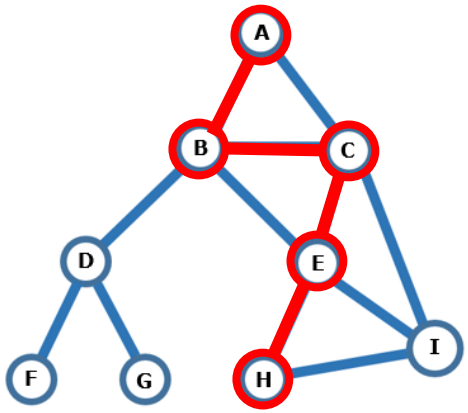
| | |
|---|---|
| [1]<br>  | **Se toma como vértice inicial E,**<br>**E ← visitado (el conjunto de vértices adyacentes al E son A , B)** |
| [2]<br>  | **Se toma vértice A**<br>**A ← visitado (el conjunto de vértices adyacentes al A son E , D)**<br>**//E ya está visitado, por lo que se continúa con D** |

| | |
|---|---|
| [3]<br>  | **Se toma vértice D**<br>**D ← visitado (el conjunto de vértices adyacentes al D es C)** |
| [4]<br>  | **Se toma vértice C**<br>**C ← visitado (el conjunto de vértices adyacentes al C son B , D)**<br>**//D ya está visitado, por lo que se continúa con B** |
| [5]<br>  | **Se toma vértice B**<br>**B ← visitado (el conjunto de vértices adyacentes al B es E)**<br><br>**//E ya está visitado, no hay más vértices,**<br>**//por lo que se termina el recorrido** |

*Ejemplo 2.* Efectuar un recorrido en profundidad del grafo tomando como inicial el nodo A.

| | |
|---|---|
|  | **Se toma como vértice inicial A,**<br>**A ← visitado (el conjunto de vértices adyacentes al A son B, C)** |
|  | **Se toma vértice B**<br>**B ← visitado**<br>**(el conjunto de vértices adyacentes al B son A, C, D, E)**<br>**//A ya está visitado, por lo que se continúa con C** |

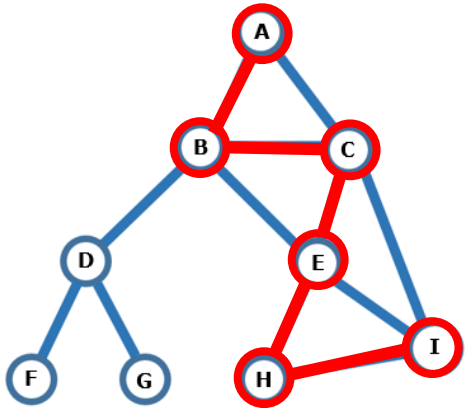
| | |
|---|---|
|  | **Se toma vértice C**<br>**C ← visitado**<br>**(el conjunto de vértices adyacentes al C son A, B, E, I)**<br>**//A , B ya están visitados, por lo que se continúa con E** |
|  | **Se toma vértice E**<br>**E ← visitado**<br>**(el conjunto de vértices adyacentes al E son B, C, H, I)**<br>**//B , C ya están visitados, por lo que se continúa con H** |




Se toma vértice H

**H ← visitado**

(el conjunto de vértices adyacentes al H son E , I)
//E ya está visitado, por lo que se continúa con I

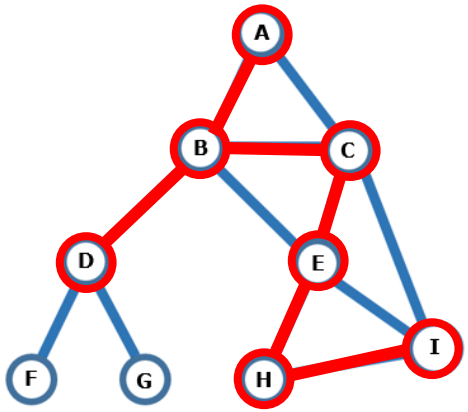



Se toma vértice I

**I ← visitado**

(el conjunto de vértices adyacentes al I son C, E, H)
//C, E, H ya están visitados,
//se termina el recorrido en profundidad a partir de I

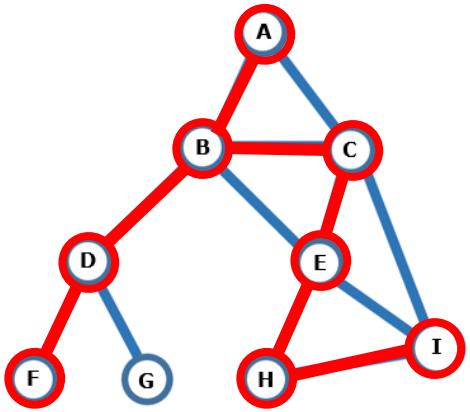
//Se hace un "backtracking" desde I y revisar sus adyacencias,
//y al retroceder hasta el nodo B, aún falta el nodo D,
//de modo que se puede continuar el recorrido




Se toma vértice D

**D ⟵ visitado**

(el conjunto de vértices adyacentes al D son F , G)
//B ya está visitado, por lo que se continúa con F

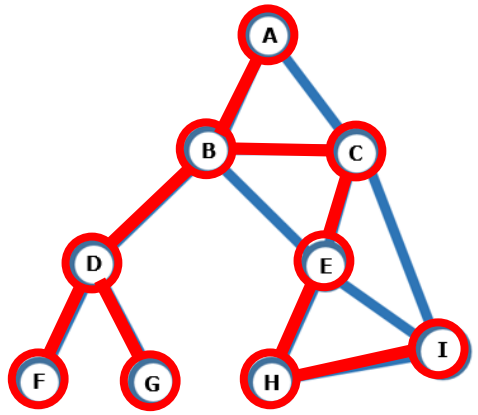



Se toma vértice F

**F ⟵ visitado**

(el conjunto de vértices adyacentes al F es D)
//D ya está visitado,
//se termina el recorrido en profundidad a partir de F

//Se hace "backtracking" desde F y revisar sus adyacencias,
//y al retroceder hasta el nodo D, aún falta el nodo G,
//de modo que se puede continuar el recorrido

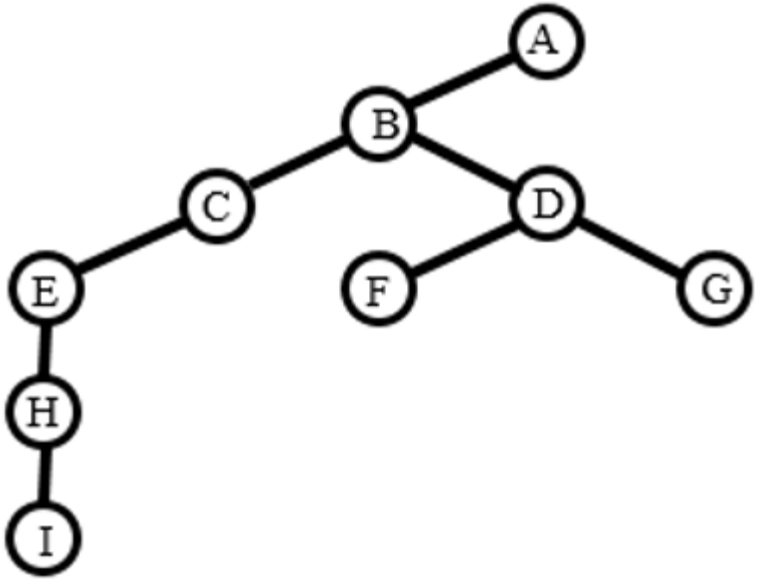



Se toma vértice G

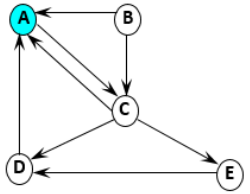
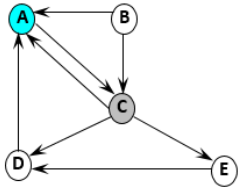
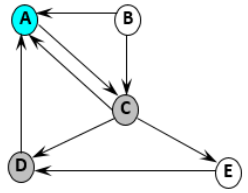
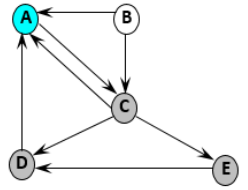
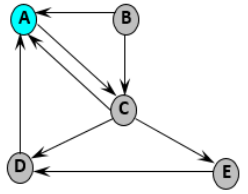
**G ← visitado**

(el conjunto de vértices adyacentes al G es D)
//D ya está visitado, no hay más vértices,
//por lo que se termina el recorrido

Y este es el árbol de recorrido:



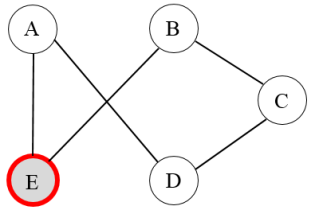
*Ejemplo 3.* Recorrido en profundidad en un grafo dirigido iniciando en el nodo A.

| | |
|---|---|
|  | Se toma como vértice inicial A,<br>**A ← visitado** (el conjunto de vértices adyacentes desde A es C) |
|  | Se toma vértice C<br>**C ← visitado** (el conjunto de vértices adyacentes desde C son A, D, E)<br>//A ya está visitado, por lo que se continúa con D |
|  | Se toma vértice D<br>**D ← visitado** (el conjunto de vértices adyacentes desde D es A)<br>//A ya está visitado, se termina el recorrido en profundidad a partir de D<br><br>//Se hace "backtracking" desde D y revisar sus adyacencias,<br>//y al retroceder hasta el nodo C, aún falta el nodo E,<br>//de modo que se puede continuar el recorrido |
|  | Se toma vértice E<br>**E ← visitado** (el conjunto de vértices adyacentes desde E es D)<br>//D ya está visitado, se termina el recorrido en profundidad a partir de E<br><br>//No es posible alcanzar más vértices desde A, entonces debe seleccionarse un nuevo vértice<br>//desde el cual recomenzar la exploración en profundidad, se elige el vértice faltante |
|  | Se toma vértice B<br>**B ← visitado** (el conjunto de vértices adyacentes desde B son A , C)<br>//A, C ya están visitados<br>//de manera que se termina el recorrido en profundidad a partir del vértice B<br><br>//No hay más vértices, por lo que se termina el recorrido. |

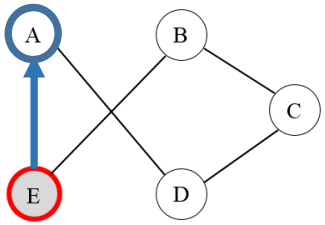
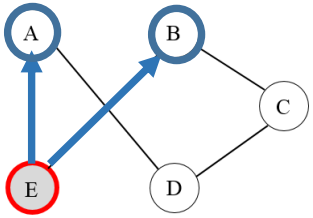
**2. Recorrido en amplitud ó anchura (Breadth-First Search –BFS-).** En un recorrido en amplitud, se elige un vértice no visitado v, como punto de partida y se pasa a visitar cada uno de sus vértices adyacentes, para continuar posteriormente visitando los adyacentes a estos últimos y así sucesivamente hasta que no se puedan alcanzar más vértices. Si queda algún vértice sin visitar, se selecciona y se vuelve a relanzar el proceso.

*Ejemplo 4.* Recorrido en amplitud en un grafo no dirigido.

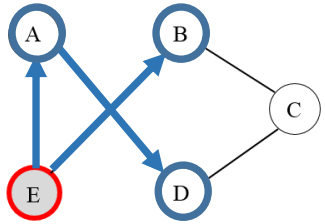
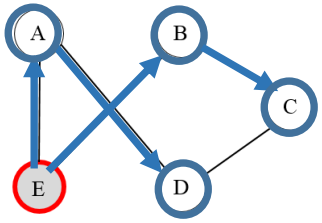
Etapa 1. Se toma el nodo de origen

| | |
|---|---|
|  | Se toma como vértice inicial E,<br>**E ← visitado**<br>(el conjunto de vértices adyacentes al E son A, B) |

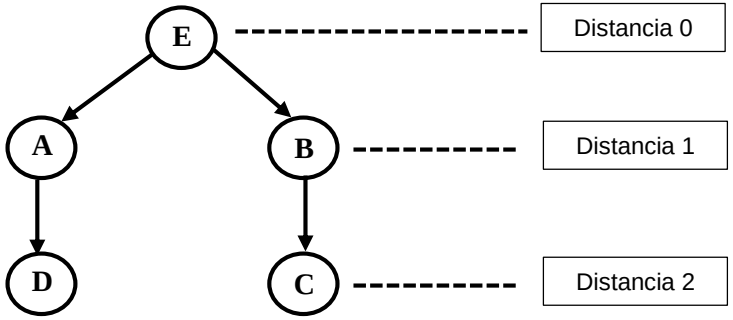
Etapa 2. Acceder a todos los nodos que están a distancia 1 del nodo origen, es decir, directamente relacionados con el origen (existe un arco que los une).

| | |
|---|---|
|  | Se toma vértice A<br>**A ← visitado**<br>(el conjunto de vértices adyacentes al A son E, D)<br>//E ya está visitado.<br>//Backtracking al nodo E y se revisan adyacencias |
|  | Se toma vértice B<br>**B ← visitado**<br>(el conjunto de vértices adyacentes al B son C, E)<br>//E ya está visitado<br>//Backtracking al nodo A y se revisan adyacencias |

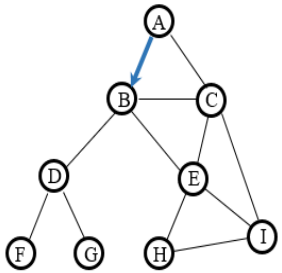
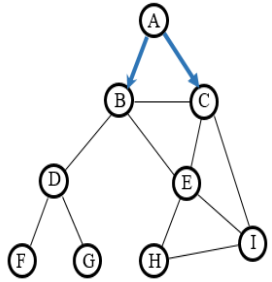
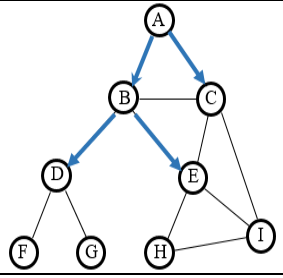
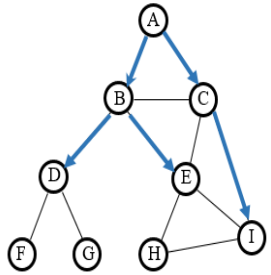
Etapa 3. Acceder a todos los nodos que están a distancia 2, es decir, directamente relacionados con los que estaban a distancia 1 en la Etapa anterior. Y así hasta cubrir los nodos.

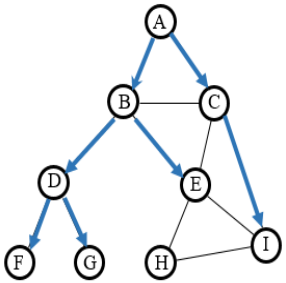
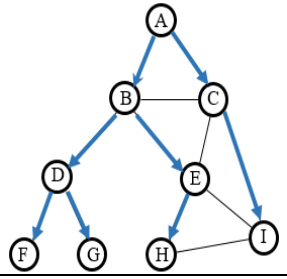
| | |
|---|---|
|  | Se toma vértice D<br>**D ← visitado**<br>(el conjunto de vértices adyacentes al D son A , C)<br>//Backtracking al nodo B y se revisan adyacencias |
|  | Se toma vértice C<br>**C ← visitado**<br>(el conjunto de vértices adyacentes a C son B , D)<br>//B , D ya están visitados,<br>//no hay más vértices, por lo que se termina el recorrido |

El resultado del recorrido es un árbol, que incluye los nodos visitados y los arcos utilizados para acceder a ellos.

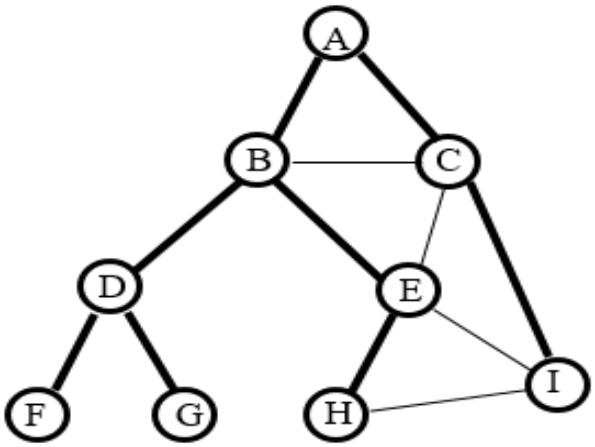


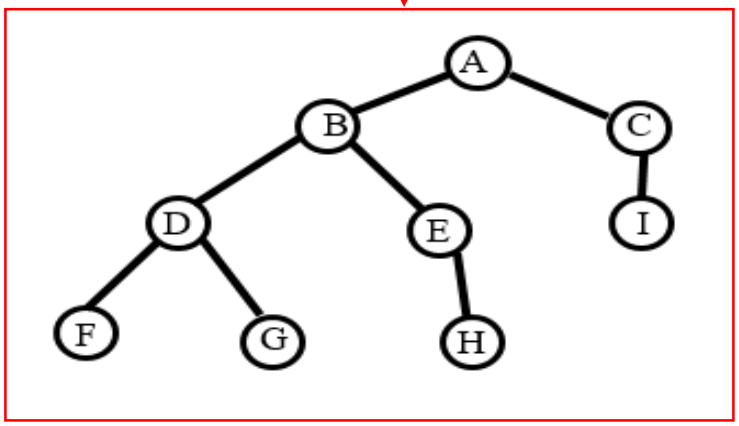
*Ejemplo 5.* Efectuar un recorrido en amplitud del grafo tomando como inicial el nodo A.

| | |
|---|---|
|  | Se toma como vértice inicial A,<br>**A ⟵ visitado**<br>(el conjunto de vértices adyacentes al A son B, C) |
|  | Se toma vértice B. **B ← visitado**<br>(el conjunto de vértices adyacentes al B son A, C, D, E)<br>//A ya está visitado. Backtracking al nodo A y se revisan adyacencias<br><br>Se toma vértice C. **C ← visitado**<br>(el conjunto de vértices adyacentes al C son A, B, E, I)<br>//A , B ya están visitados<br><br>//Backtracking al nodo B y se revisan adyacencias |
|  | Se toma vértice D. **D ← visitado**<br>(el conjunto de vértices adyacentes al D son F, G)<br><br>//Backtracking al nodo B y se revisan adyacencias<br>Se toma vértice E. **E ← visitado**<br>(el conjunto de vértices adyacentes al E son C, H, I)<br>//C ya está visitado. Backtracking al nodo C y se revisan adyacencias |
|  | Se toma vértice I<br>**I ← visitado**<br>(el conjunto de vértices adyacentes al I son C, E, H)<br>//C, E ya están visitados. Backtracking al nodo D y se revisan adyacencias |

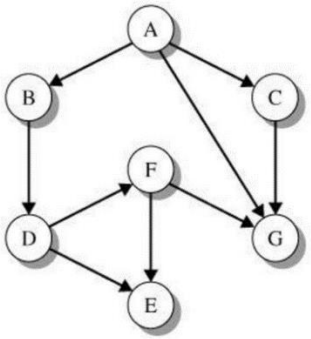
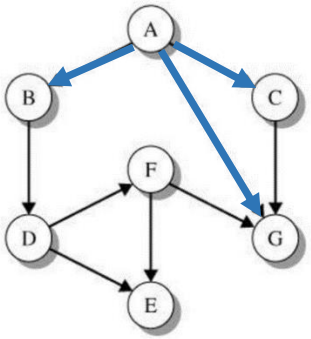
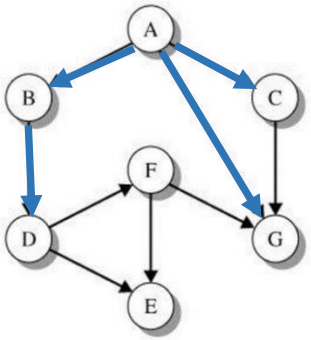
| | |
|---|---|
|  | Se toma vértice F. **F ← visitado**<br>(el conjunto de vértices adyacentes al F es D)<br>//D ya está visitado. Backtracking al nodo D y se revisan adyacencias<br><br>Se toma vértice G. **G ← visitado**<br>(el conjunto de vértices adyacentes al G es D)<br>//D ya está visitado. Backtracking al nodo E y se revisan adyacencias |
|  | Se toma vértice H. **H ← visitado**<br>(el conjunto de vértices adyacentes al H son E, I)<br>//E , I ya están visitados.<br>//No hay más vértices, por lo que se termina el recorrido |

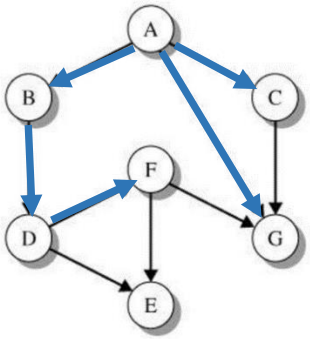
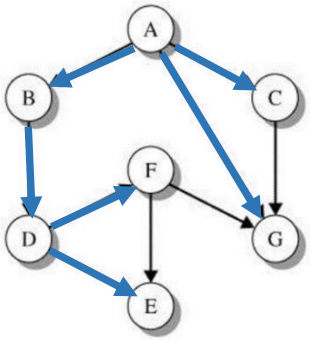
Y este es el árbol de recorrido:



*Ejemplo 6.* Recorrido en amplitud en un grafo dirigido iniciando en el nodo A.

| | |
|---|---|
|  | Se toma como vértice inicial A,<br>**A ← visitado**<br>(el conjunto de vértices adyacentes desde A son B, C, G) |
|  | Se toma vértice B. **B ← visitado**<br>(el conjunto de vértices adyacentes desde B es D)<br>// Backtracking al nodo A y se revisan adyacencias<br><br>Se toma vértice C. **C ← visitado**<br>(el conjunto de vértices adyacentes desde C es G)<br>// Backtracking al nodo A y se revisan adyacencias<br><br>Se toma vértice G. **G ← visitado**<br>(el conjunto de vértices adyacentes desde G es vacío)<br>// Backtracking al nodo B y se revisan adyacencias |
|  | Se toma vértice D. **D ← visitado**<br>(el conjunto de vértices adyacentes desde D son F, E)<br><br>// Backtracking al nodo C y se revisan adyacencias<br>//G ya está visitado<br><br>//Baktracking nodo D y se revisan adyacencias |

| | |
|---|---|
|  | Se toma vértice F. **F ← visitado**<br>(el conjunto de vértices adyacentes desde F es E, G)<br>//G ya visitado<br><br>//Baktracking nodo D y se revisan adyacencias |
|  | Se toma vértice E. **E ← visitado**<br>(el conjunto de vértices adyacentes desde F es G)<br>//G ya visitado<br><br>//No hay más vértices, por lo que se termina el recorrido |

***---------- FIN DEL DOCUMENTO***