

Algoritmos y estructuras de datos

Búsqueda en profundidad

Francisco Javier Zaragoza Martínez

Universidad Autónoma Metropolitana Unidad Azcapotzalco
Departamento de Sistemas



2 de junio de 2021

Buckminster Fuller

Los años oscuros todavía reinan sobre la humanidad, y la **profundidad** y la persistencia de esta dominación están apenas ahora volviéndose claras. Esta prisión de los años oscuros no tiene barrotes, ni cadenas, ni candados. En lugar de eso, está cerrada por la desorientación y construida de desinformación.

George MacDonald

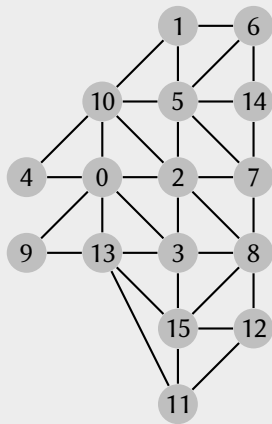
Me encuentro más allá de mi **profundidad**. Siempre estoy más allá de mi profundidad, flotando en un mar infinito; pero la **profundidad** del mar me conoce.

Jacob Boehme

Aquí no hay ni luz ni oscuridad, ni luz ni fuego, ni bien ni mal. Aquí no hay ni altura ni **profundidad**, ni grande ni pequeña, ni ancha ni delgada. Aquí está todo y nada.

Gráficas

Caminos y conexidad



Camino Secuencia de vértices en la que cada dos consecutivos están unidos por una arista.

Camino simple Camino en el que no se repiten vértices.

Gráfica conexa Existen caminos entre cada dos vértices.

Componente conexa Si una gráfica no es conexa, entonces tiene varias componentes.

Circuito Camino cerrado (inicio = fin).

Ciclo Circuito en el que no se repiten vértices.

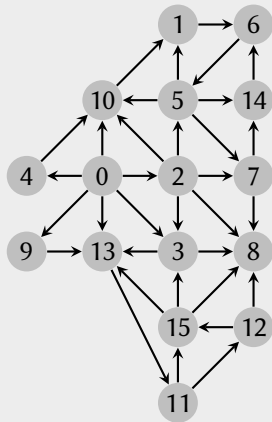
Bosque Gráfica que no tiene ciclos.

Árbol Bosque conexo.

Árbol abarcador Abarca todos los vértices.

Digráficas

Caminos y conexidad fuerte



Camino Secuencia de vértices en la que existe un arco de cada vértice al siguiente.

Camino simple Camino en el que no se repiten vértices.

Digráfica fuertemente conexa Existen caminos de cada vértice a todos los demás.

Componente fuertemente conexa Si una digráfica no es fuertemente conexa, entonces tiene varias componentes fuertemente conexas.

Circuito Camino cerrado (inicio = fin).

Ciclo Circuito en el que no se repiten vértices.

Digráfica acíclica Digráfica que no tiene ciclos.

Recorridos de gráficas

Cuando se trabaja con una gráfica G , uno se encuentra preguntas comunes:

- 1 ¿Existe algún camino de u a v ?
- 2 ¿Será que G es conexa?
- 3 Si no ¿cuáles son sus componentes conexas?
- 4 Si sí ¿se puede destruir fácilmente la conexidad?
- 5 ¿Tiene G ciclos?

Para responder a estas preguntas se debe recorrer G de forma algorítmica, de modo que se visiten todos sus vértices y aristas (de preferencia una sola vez). Estos algoritmos pueden ser recursivos o iterativos y dependen un poco de la representación de G .

Búsqueda en profundidad recursiva

Un primer algoritmo, llamado **búsqueda en profundidad** (DFS por las siglas en inglés de *depth first search*), se puede describir fácilmente de forma recursiva:

- 1 Se marcan todos los vértices como **no vistos**.
- 2 Mientras haya algún vértice u no visto:
 - 1 Se marca a u como **ya visto**.
 - 2 Se visita recursivamente a todos los vecinos no vistos de u .

Búsqueda en profundidad recursiva

Implementación con matrices de adyacencia

Usaremos un arreglo en el que `visto[u]` será el `orden` de visita de `u` (0 si no se ha visto).

```
void busquedaProfundidad(grafica g) {  
    int orden = 0;  
    int *visto = (int *) calloc(g.n, sizeof(int));  
    for (int u = 0; u < g.n; u++)  
        if (visto[u] == 0)  
            profundidad(g, u, visto, &orden);  
    free(visto);  
}
```

Búsqueda en profundidad recursiva

Implementación con matrices de adyacencia

Esta función recorre cada renglón de la matriz de adyacencia, por lo que hace n^2 pasos.

```
void profundidad(grafica g, int u, int visto[], int *orden) {  
    visto[u] = ++(*orden);  
    for (int v = 0; v < g.n; v++)  
        if (g.a[u][v] == 1 && visto[v] == 0)  
            profundidad(g, v, visto, orden);  
}
```


Búsqueda en profundidad recursiva

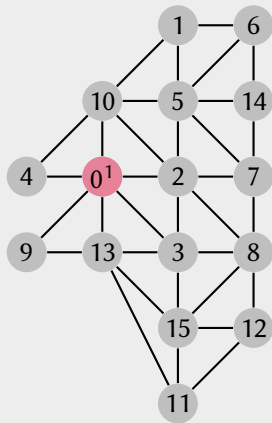
Implementación con listas de adyacencia

Esta función recorre cada lista de adyacencia, por lo que hace $n + m$ pasos.

```
void profundidadLA(graficaLA g, int u, int visto[], int *orden) {  
    visto[u] = ++(*orden);  
    for (nodo *v = g.a[u]; v != NULL; v = v->sig)  
        if (visto[v->a] == 0)  
            profundidadLA(g, v->a, visto, orden);  
}
```

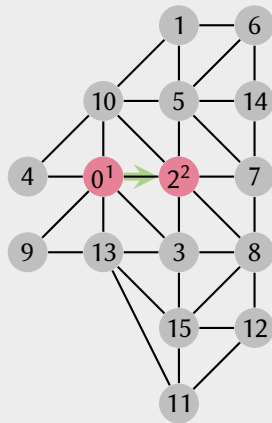
Búsqueda en profundidad recursiva

Ejemplo de gráfica



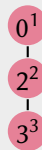
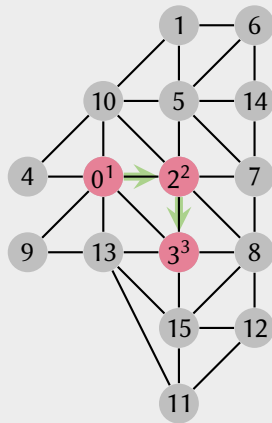
0¹

Ejemplo de gráfica



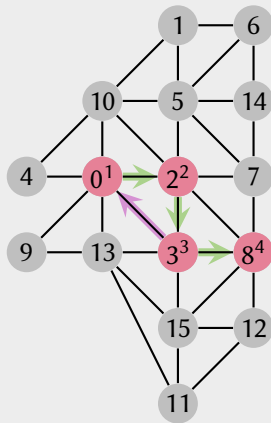
Búsqueda en profundidad recursiva

Ejemplo de gráfica



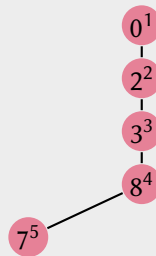
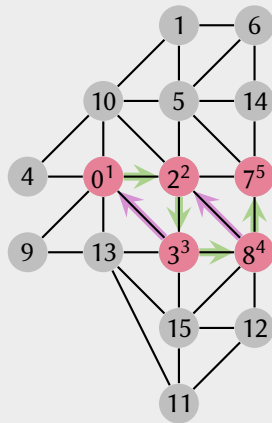
Búsqueda en profundidad recursiva

Ejemplo de gráfica



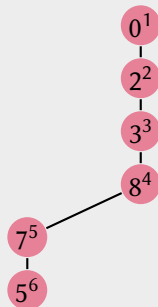
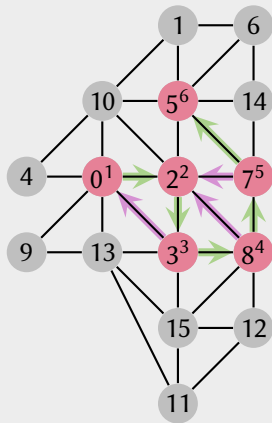
Búsqueda en profundidad recursiva

Ejemplo de gráfica



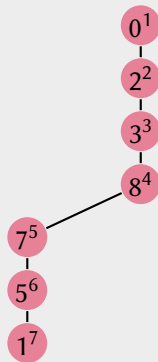
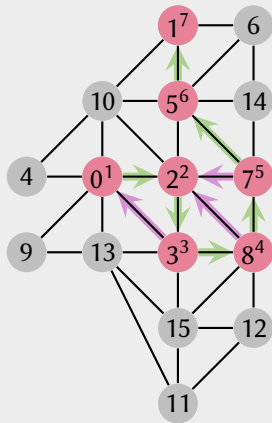
Búsqueda en profundidad recursiva

Ejemplo de gráfica



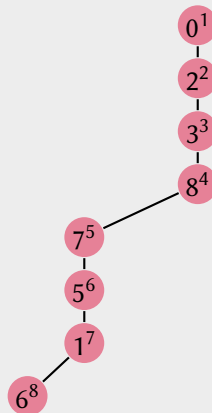
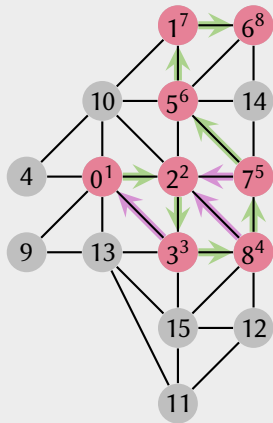
Búsqueda en profundidad recursiva

Ejemplo de gráfica



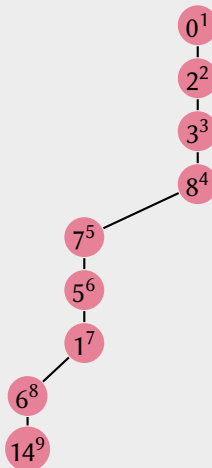
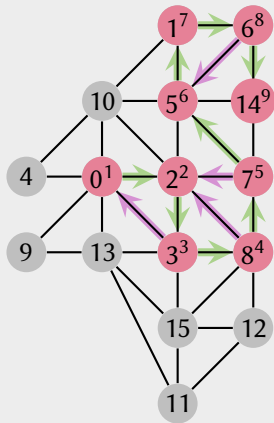
Búsqueda en profundidad recursiva

Ejemplo de gráfica



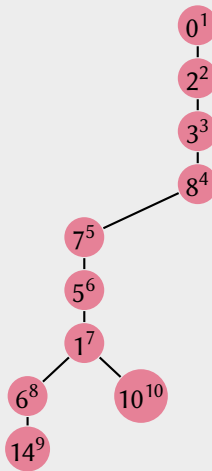
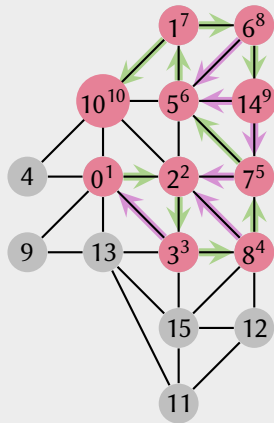
Búsqueda en profundidad recursiva

Ejemplo de gráfica



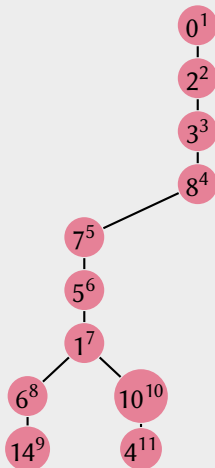
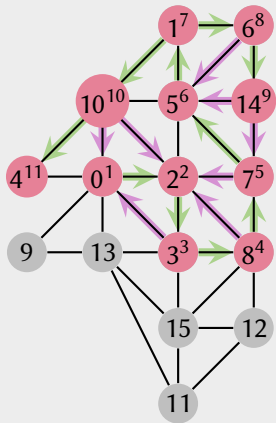
Búsqueda en profundidad recursiva

Ejemplo de gráfica



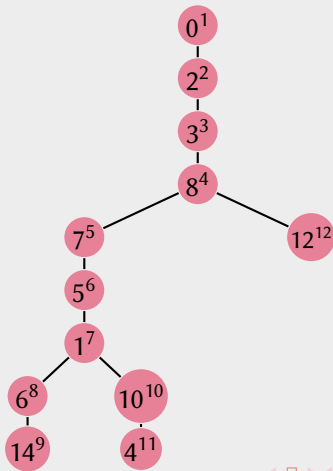
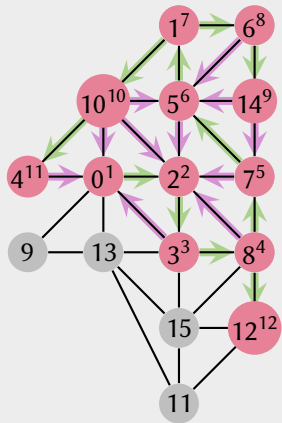
Búsqueda en profundidad recursiva

Ejemplo de gráfica

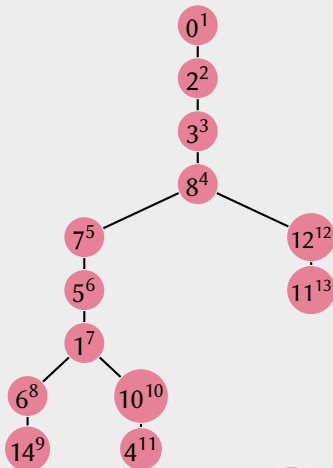
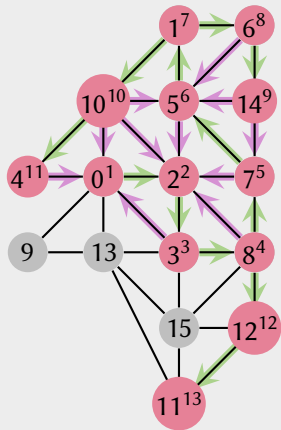


Búsqueda en profundidad recursiva

Ejemplo de gráfica

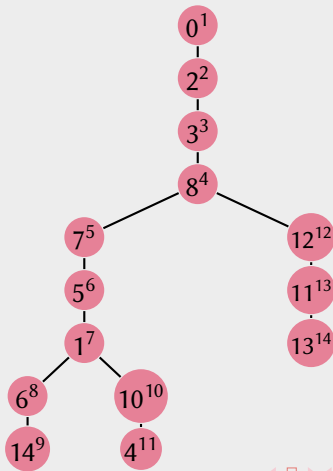
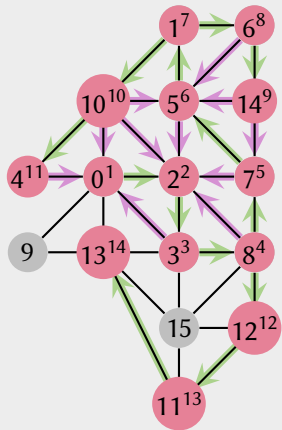


Ejemplo de gráfica



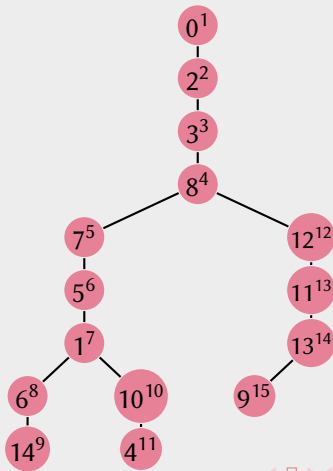
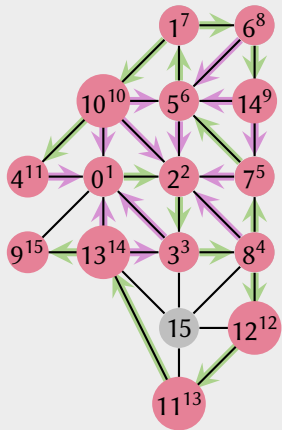
Búsqueda en profundidad recursiva

Ejemplo de gráfica



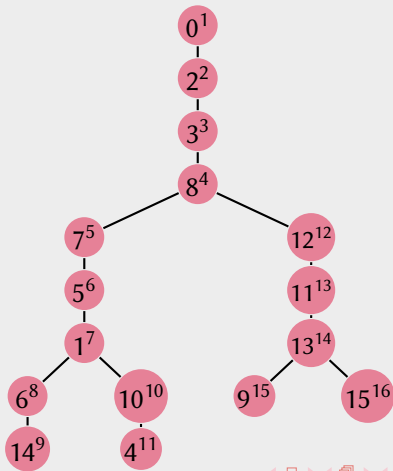
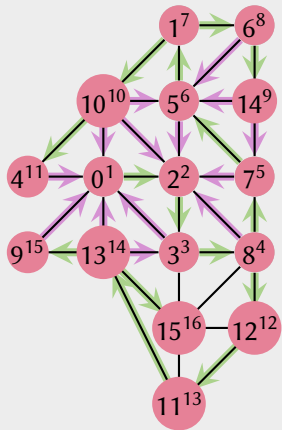
Búsqueda en profundidad recursiva

Ejemplo de gráfica



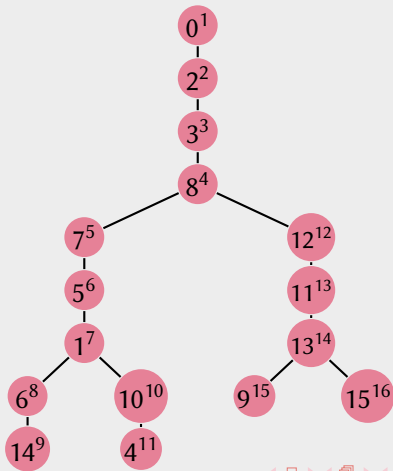
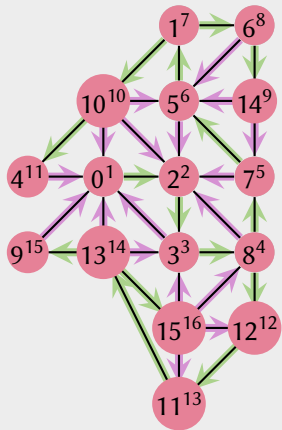
Búsqueda en profundidad recursiva

Ejemplo de gráfica



Búsqueda en profundidad recursiva

Ejemplo de gráfica



Búsqueda en profundidad no recursiva

La **búsqueda en profundidad** también se puede describir fácilmente sin usar recursión. Esto se logra usando una pila inicialmente vacía.

- 1 Se marcan todos los vértices como **no vistos**.
- 2 Mientras haya algún vértice u no visto:
 - 1 Se apila u .
 - 2 Mientras la pila no esté vacía:
 - 1 Se desapila u .
 - 2 Se marca a u como **ya visto**.
 - 3 Se apila a todos los vecinos no vistos de u .

Lo peor que puede pasar es que todos los vértices de G estén simultáneamente en la pila. Es suficiente usar una pila estática de tamaño n (funciones que terminan en **pilaAE**).

Búsqueda en profundidad no recursiva

Implementación con matrices de adyacencia

```
void profundidad(grafica g, int u, int visto[], int *orden) {
    pilaAE s = creaPilaAE(g.n);
    apilaAE(&s, u);
    while (!esVacíaPilaAE(&s)) {
        desapilaAE(&s, &u);
        visto[u] = ++(*orden);
        for (int v = 0; v < g.n; v++)
            if (g.a[u][v] == 1 && visto[v] == 0) {
                apilaAE(&s, v);
                visto[v] = -1;
            }
    }
    destruyePilaAE(&s);
}
```

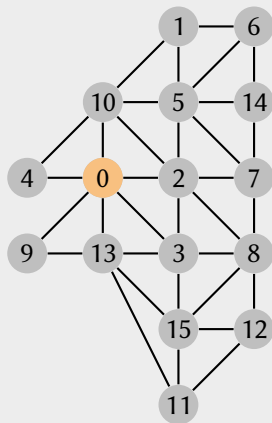
Búsqueda en profundidad no recursiva

Implementación con listas de adyacencia

```
void profundidadLA(graficaLA g, int u, int visto[], int *orden) {
    pilaAE s = creaPila(g.n);
    apilaAE(&s, u);
    while (!esVaciaPilaAE(&s)) {
        desapilaAE(&s, &u);
        visto[u] = ++(*orden);
        for (nodo *v = g.a[u]; v != NULL; v = v->sig)
            if (visto[v->a] == 0) {
                apilaAE(&s, v->a);
                visto[v->a] = -1;
            }
    }
    destruyePilaAE(&s);
}
```

Búsqueda en profundidad no recursiva

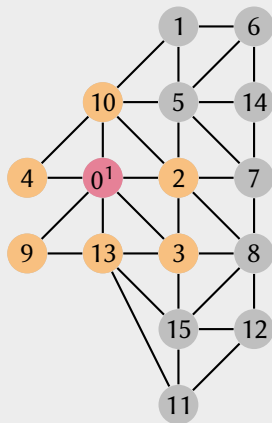
Ejemplo de gráfica



0
pilaAE s

Búsqueda en profundidad no recursiva

Ejemplo de gráfica

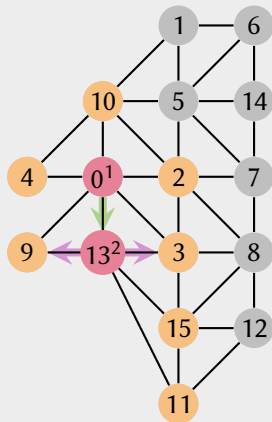


13
10
9
4
3
2
pilaAE s

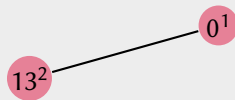
0¹

Búsqueda en profundidad no recursiva

Ejemplo de gráfica

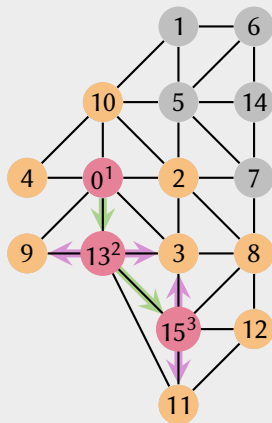


15
11
10
9
4
3
2
pilaAE s

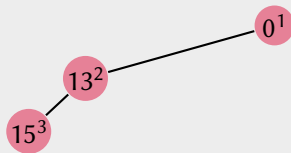


Búsqueda en profundidad no recursiva

Ejemplo de gráfica

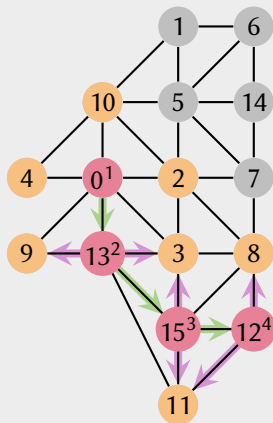


12
8
11
10
9
4
3
2
pilaAE s

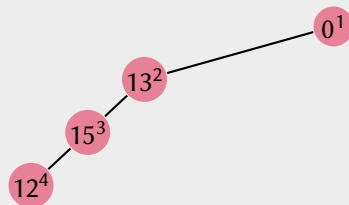


Búsqueda en profundidad no recursiva

Ejemplo de gráfica

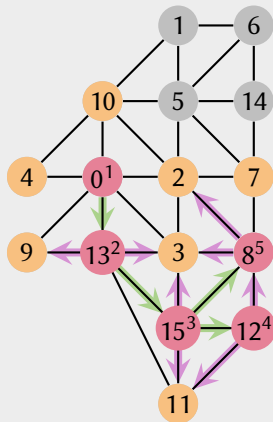


8
11
10
9
4
3
2
pilaAE s

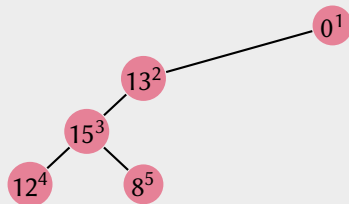


Búsqueda en profundidad no recursiva

Ejemplo de gráfica

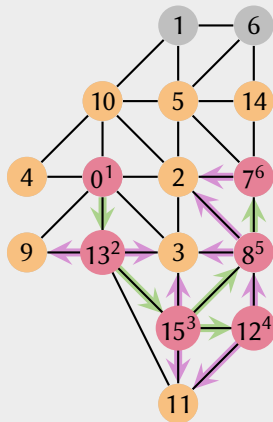


7
11
10
9
4
3
2
pilaAE s

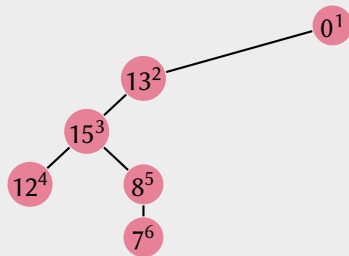


Búsqueda en profundidad no recursiva

Ejemplo de gráfica

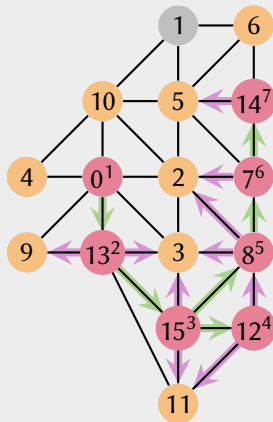


14
5
11
10
9
4
3
2
pilaAE s

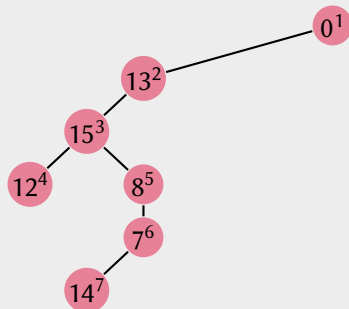


Búsqueda en profundidad no recursiva

Ejemplo de gráfica

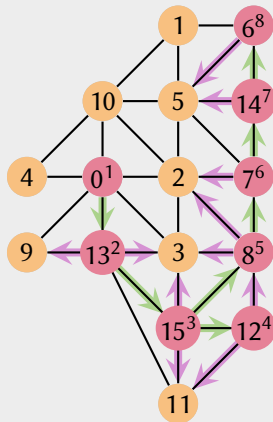


6
5
11
10
9
4
3
2
pilaAE s

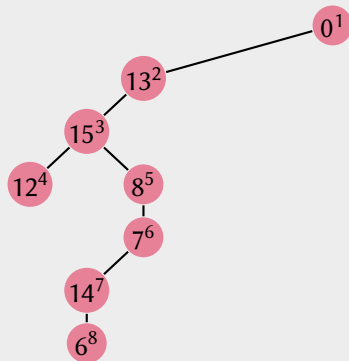


Búsqueda en profundidad no recursiva

Ejemplo de gráfica

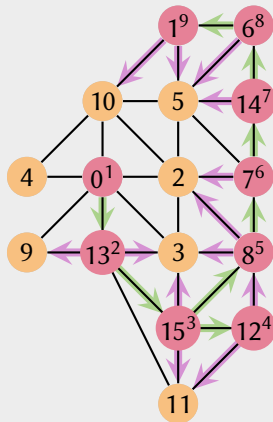


1
5
11
10
9
4
3
2
pilaAE s

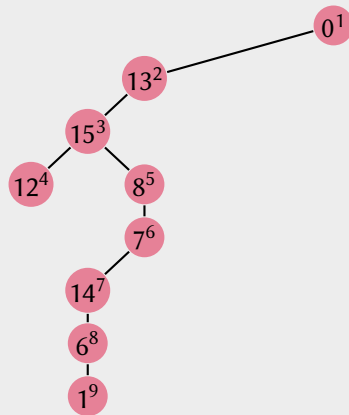


Búsqueda en profundidad no recursiva

Ejemplo de gráfica

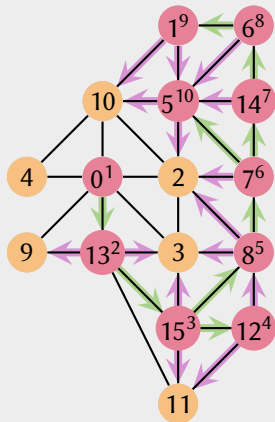


5
11
10
9
4
3
2
pilaAE s

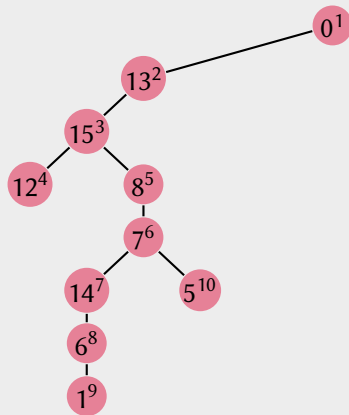


Búsqueda en profundidad no recursiva

Ejemplo de gráfica

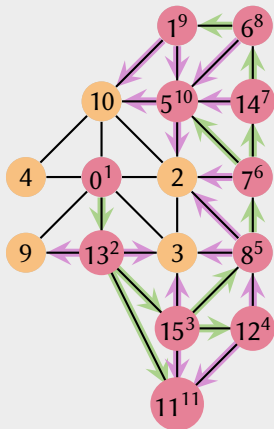


11
10
9
4
3
2
pilaAE s

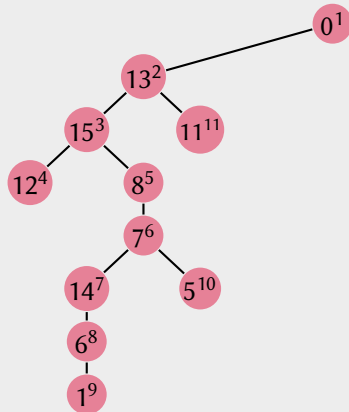


Búsqueda en profundidad no recursiva

Ejemplo de gráfica

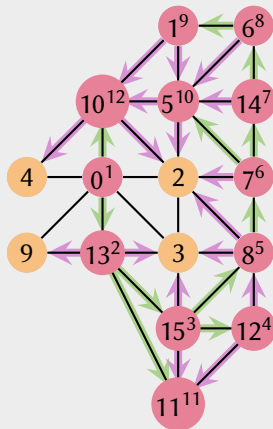


10
9
4
3
2
pilaAE s

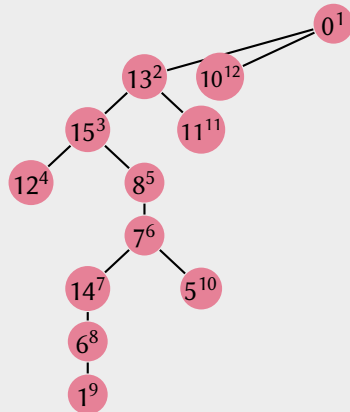


Búsqueda en profundidad no recursiva

Ejemplo de gráfica

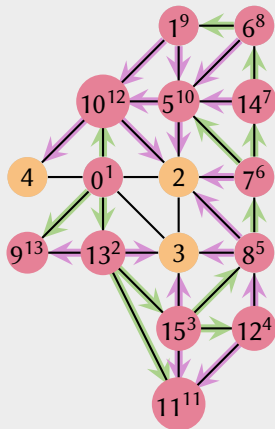


9
4
3
2
pilaAE s

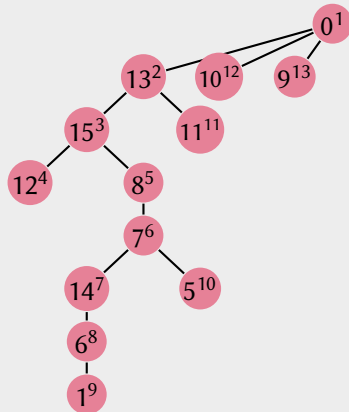


Búsqueda en profundidad no recursiva

Ejemplo de gráfica

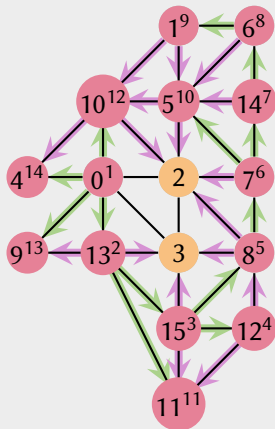


4
3
2
pilaAE s

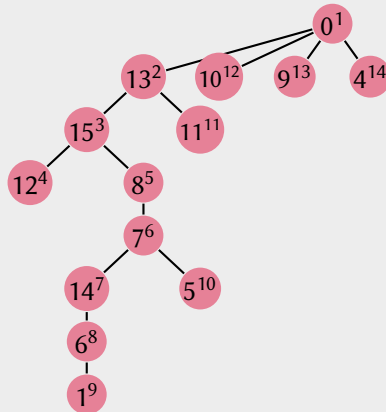


Búsqueda en profundidad no recursiva

Ejemplo de gráfica

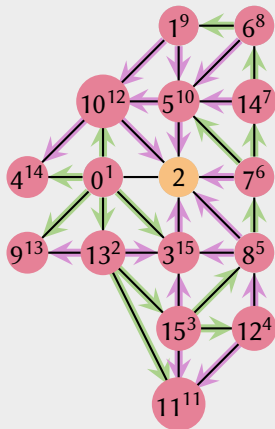


3
2
pilaAE s

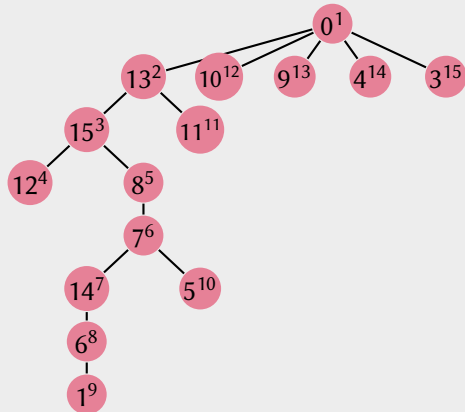


Búsqueda en profundidad no recursiva

Ejemplo de gráfica

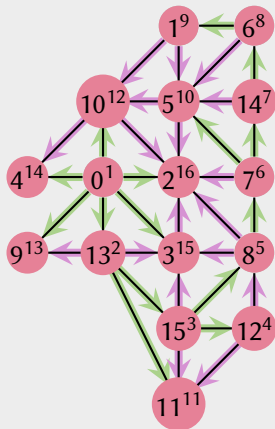


2
pilaAE s

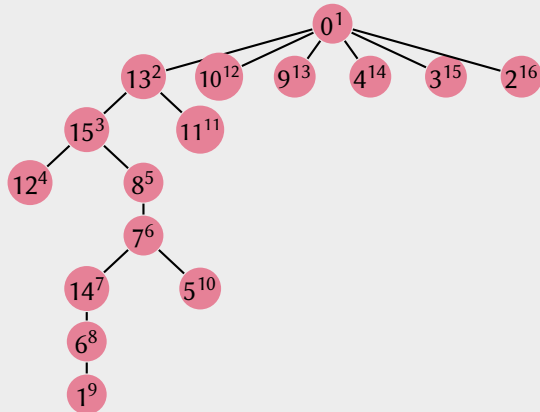


Búsqueda en profundidad no recursiva

Ejemplo de gráfica



pilaAE s



Búsqueda en profundidad

Observaciones

Este método es una generalización del recorrido en preorden de un árbol.

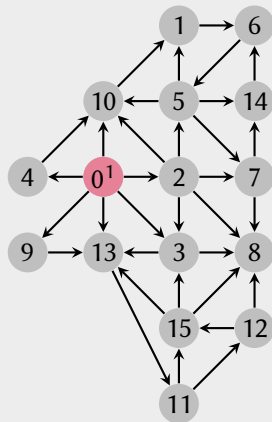
- 1 Cada vértice se visita una vez y cada arista se visita dos veces.
- 2 El tiempo de ejecución depende de la representación (matriz n^2 , listas $n + m$).
- 3 El orden de visita depende de si se usó la versión recursiva o la no recursiva.
- 4 Las aristas que visitan por primera vez vértices no vistos forman un **bosque de búsqueda en profundidad**. Las demás aristas apuntan hacia arriba en ese bosque.

Estos algoritmos se usan sin cambios en digráficas, aunque cambia el comportamiento.

- 1 Cada vértice y cada arco se visitan una vez.
- 2 Los arcos que no pertenecen al bosque de búsqueda en profundidad pueden apuntar hacia arriba, hacia abajo o ser transversales.

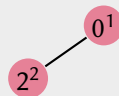
Búsqueda en profundidad recursiva

Ejemplo de digráfica



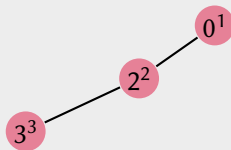
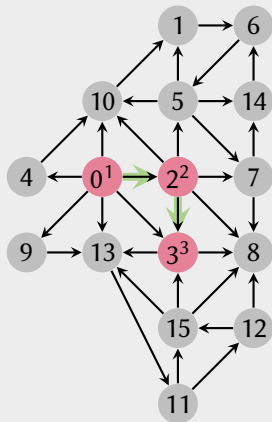
0¹

Ejemplo de digráfica



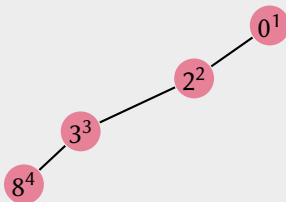
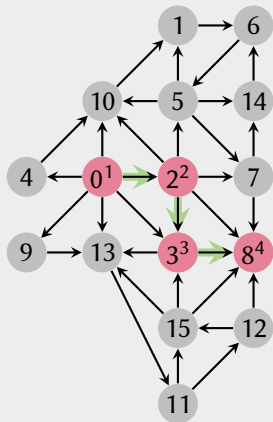
Búsqueda en profundidad recursiva

Ejemplo de digráfica



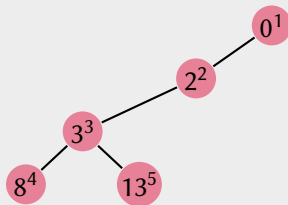
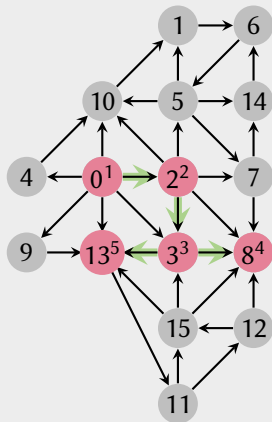
Búsqueda en profundidad recursiva

Ejemplo de digráfica



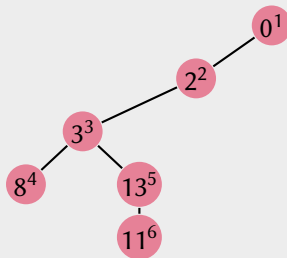
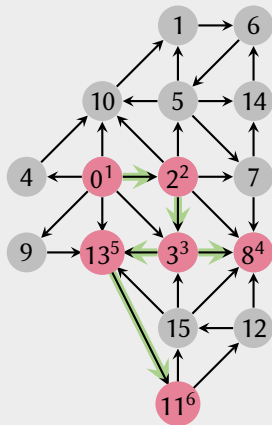
Búsqueda en profundidad recursiva

Ejemplo de digráfica



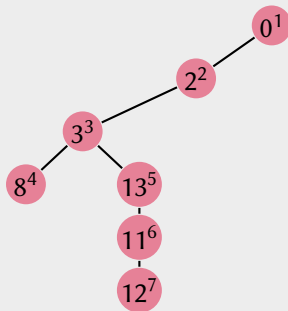
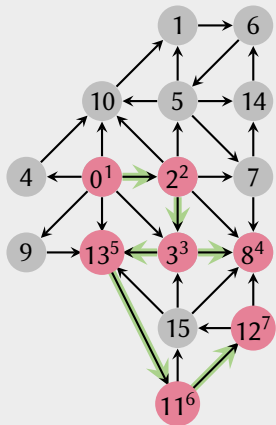
Búsqueda en profundidad recursiva

Ejemplo de digráfica



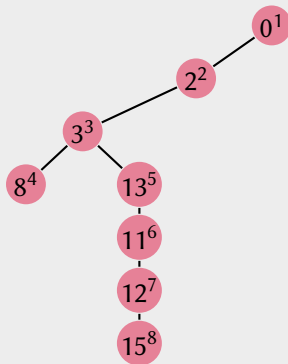
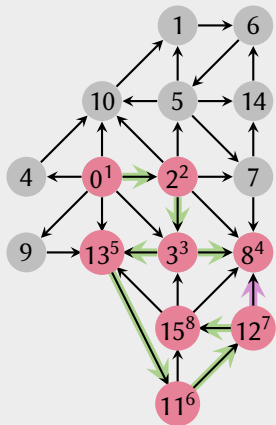
Búsqueda en profundidad recursiva

Ejemplo de digráfica



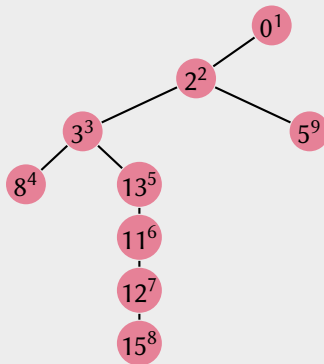
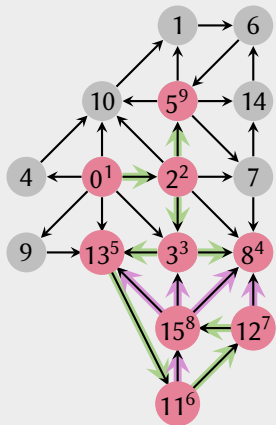
Búsqueda en profundidad recursiva

Ejemplo de digráfica

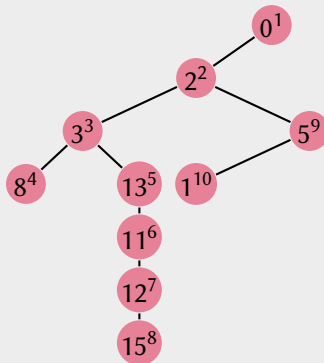
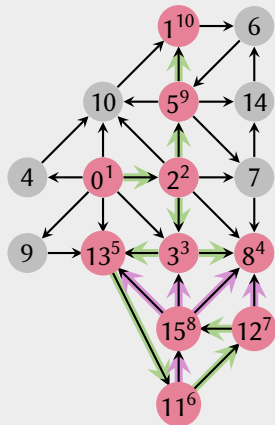


Búsqueda en profundidad recursiva

Ejemplo de digráfica

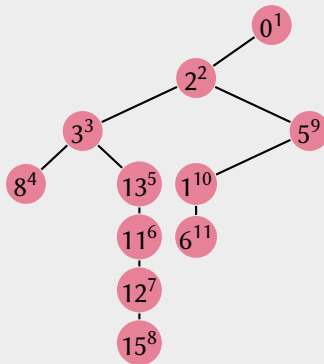
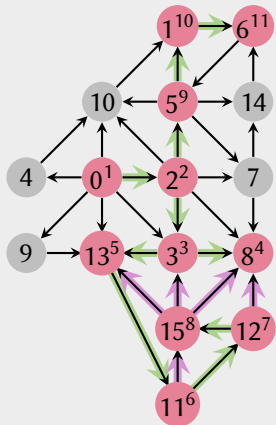


Ejemplo de digráfica



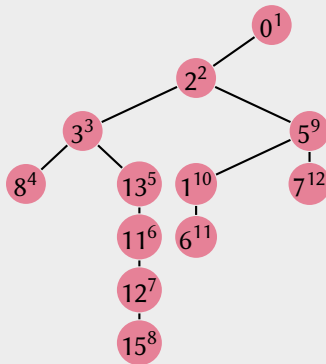
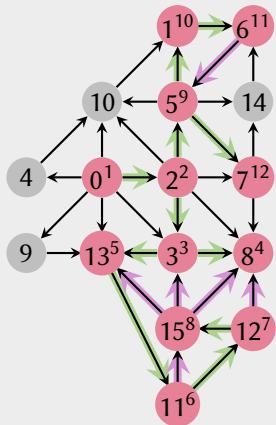
Búsqueda en profundidad recursiva

Ejemplo de digráfica



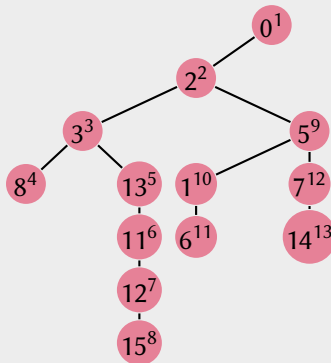
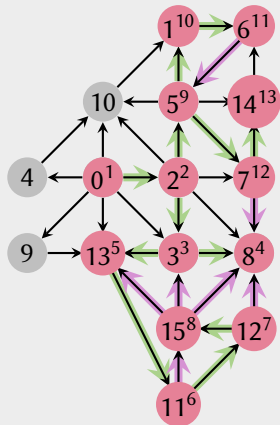
Búsqueda en profundidad recursiva

Ejemplo de digráfica



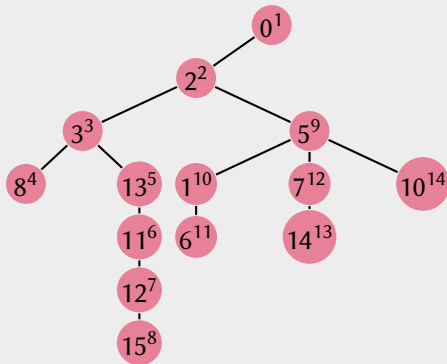
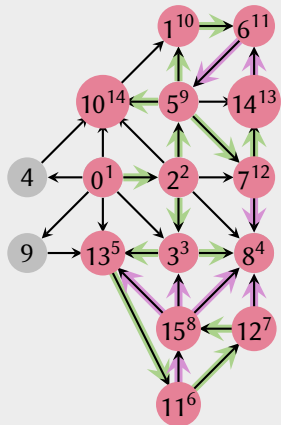
Búsqueda en profundidad recursiva

Ejemplo de digráfica



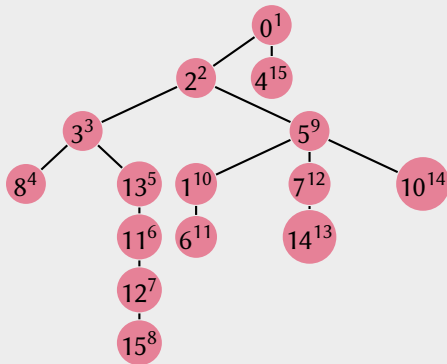
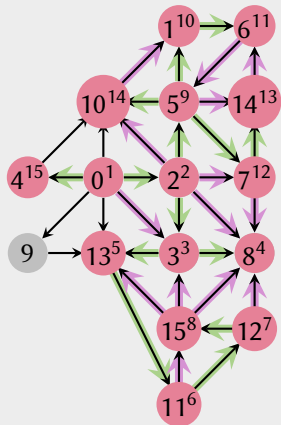
Búsqueda en profundidad recursiva

Ejemplo de digráfica



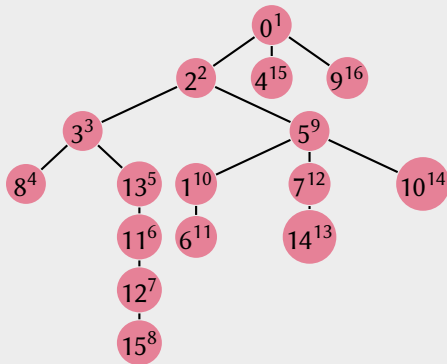
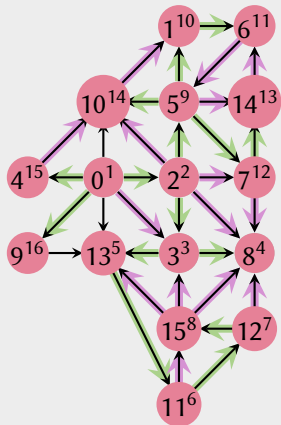
Búsqueda en profundidad recursiva

Ejemplo de digráfica



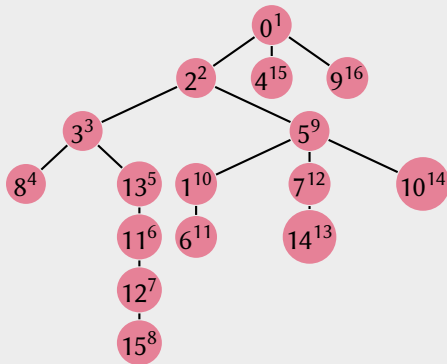
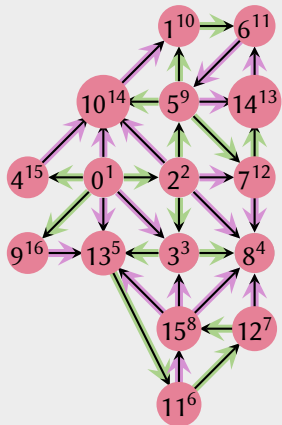
Búsqueda en profundidad recursiva

Ejemplo de digráfica



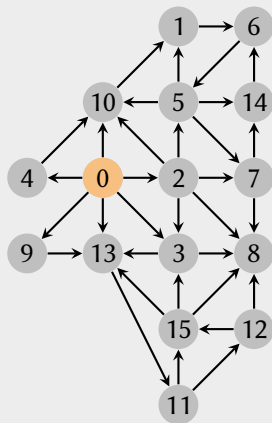
Búsqueda en profundidad recursiva

Ejemplo de digráfica



Búsqueda en profundidad no recursiva

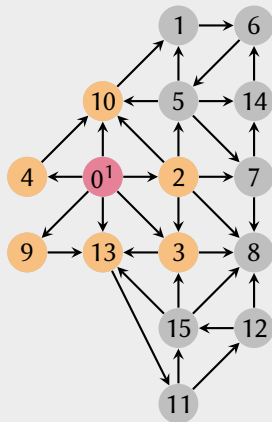
Ejemplo de digráfica



0
pilaAE s

Búsqueda en profundidad no recursiva

Ejemplo de digráfica

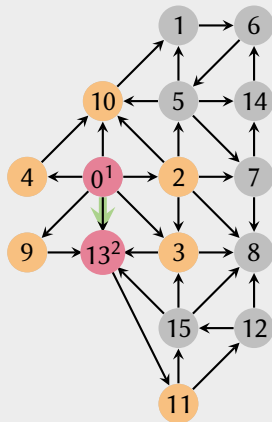


13
10
9
4
3
2
pilaAE s

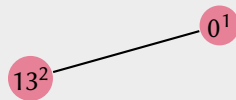
0¹

Búsqueda en profundidad no recursiva

Ejemplo de digráfica

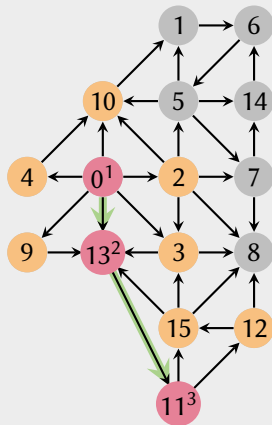


11
10
9
4
3
2
pilaAE s

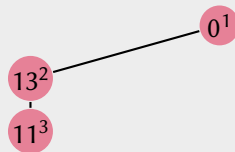


Búsqueda en profundidad no recursiva

Ejemplo de digráfica

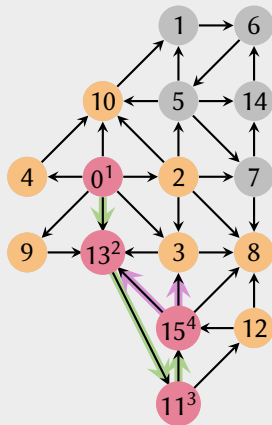


15
12
10
9
4
3
2
pilaAE s

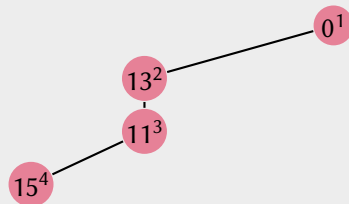


Búsqueda en profundidad no recursiva

Ejemplo de digráfica

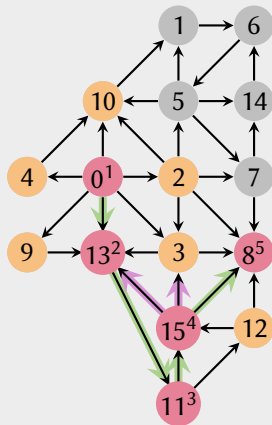


8
12
10
9
4
3
2
pilaAE s

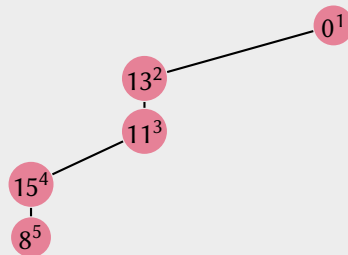


Búsqueda en profundidad no recursiva

Ejemplo de digráfica

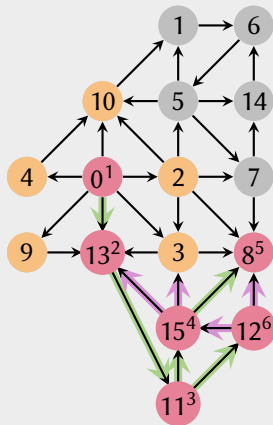


12
10
9
4
3
2
pilaAE s

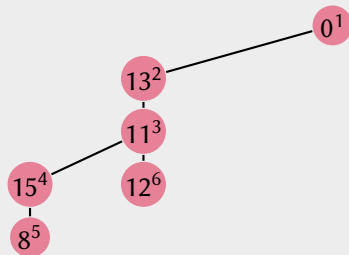


Búsqueda en profundidad no recursiva

Ejemplo de digráfica

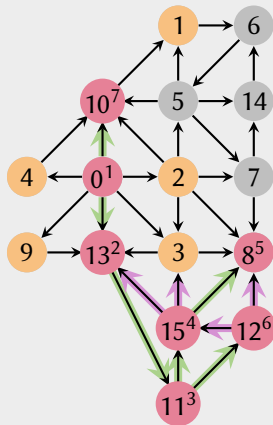


10
9
4
3
2
pilaAE s

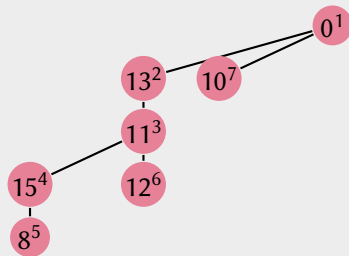


Búsqueda en profundidad no recursiva

Ejemplo de digráfica

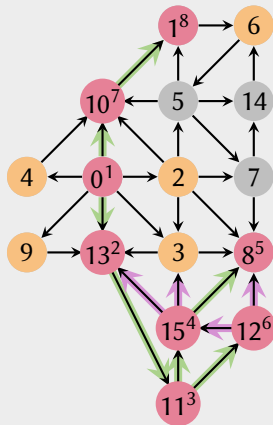


1
9
4
3
2
pilaAE s

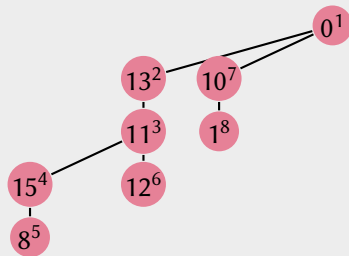


Búsqueda en profundidad no recursiva

Ejemplo de digráfica

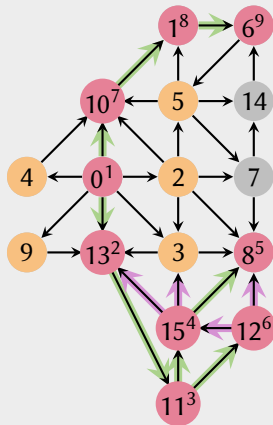


6
9
4
3
2
pilaAE s

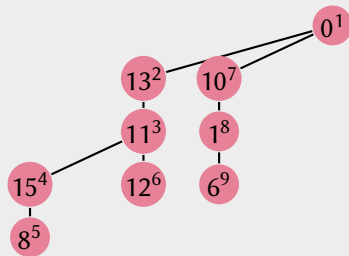


Búsqueda en profundidad no recursiva

Ejemplo de digráfica

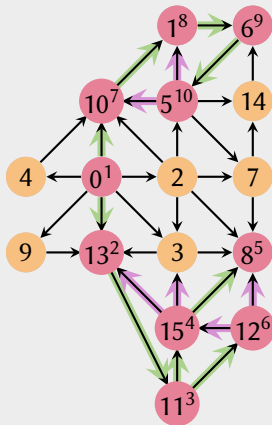


5
9
4
3
2
pilaAE s

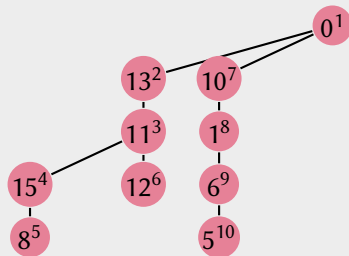


Búsqueda en profundidad no recursiva

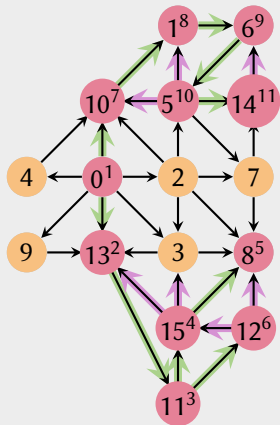
Ejemplo de digráfica



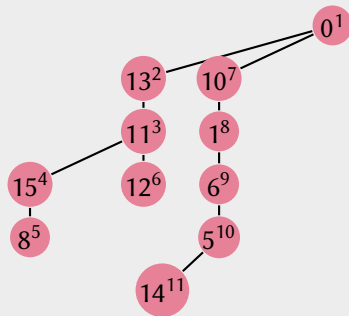
14
7
9
4
3
2
pilaAE s



Ejemplo de digráfica

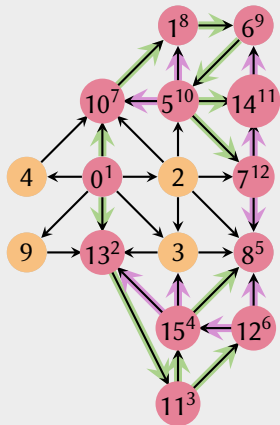


7
9
4
3
2
pilaAE s

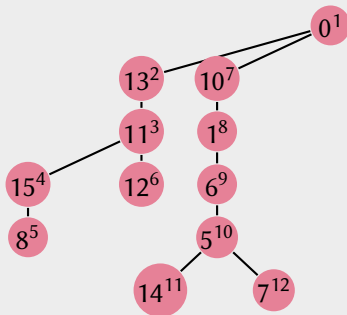


Búsqueda en profundidad no recursiva

Ejemplo de digráfica

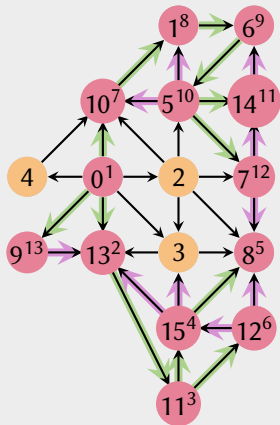


9
4
3
2
pilaAE s

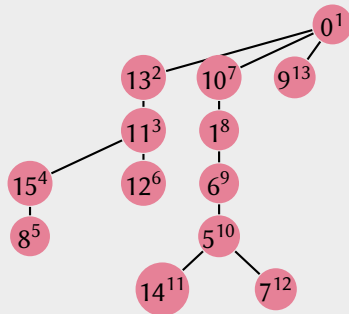


Búsqueda en profundidad no recursiva

Ejemplo de digráfica

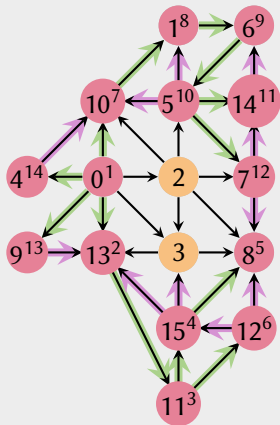


4
3
2
pilaAE s

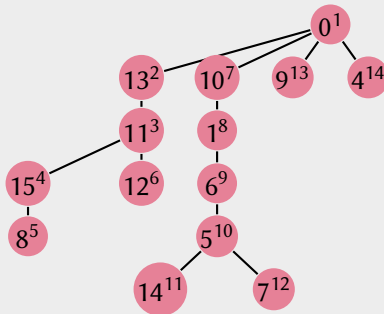


Búsqueda en profundidad no recursiva

Ejemplo de digráfica

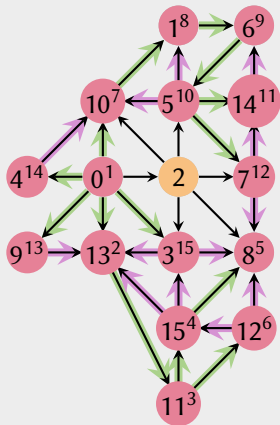


3
2
pilaAE s

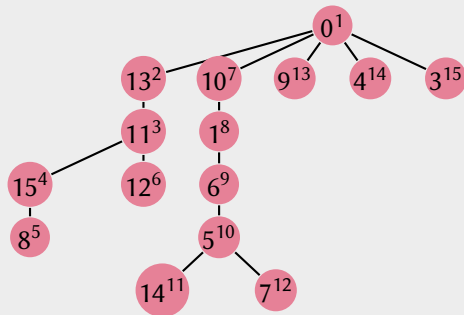


Búsqueda en profundidad no recursiva

Ejemplo de digráfica

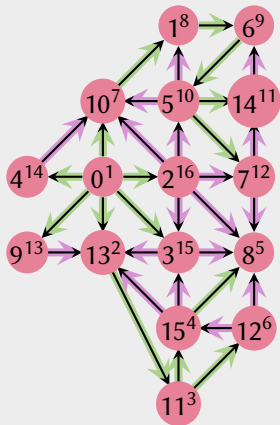


2
pilaAE s



Búsqueda en profundidad no recursiva

Ejemplo de digráfica



pilaAE s

