



UNIVERSIDAD NACIONAL DE CAJAMARCA
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA DE SISTEMAS



ALGORITMOS Y ESTRUCTURA DE DATOS II

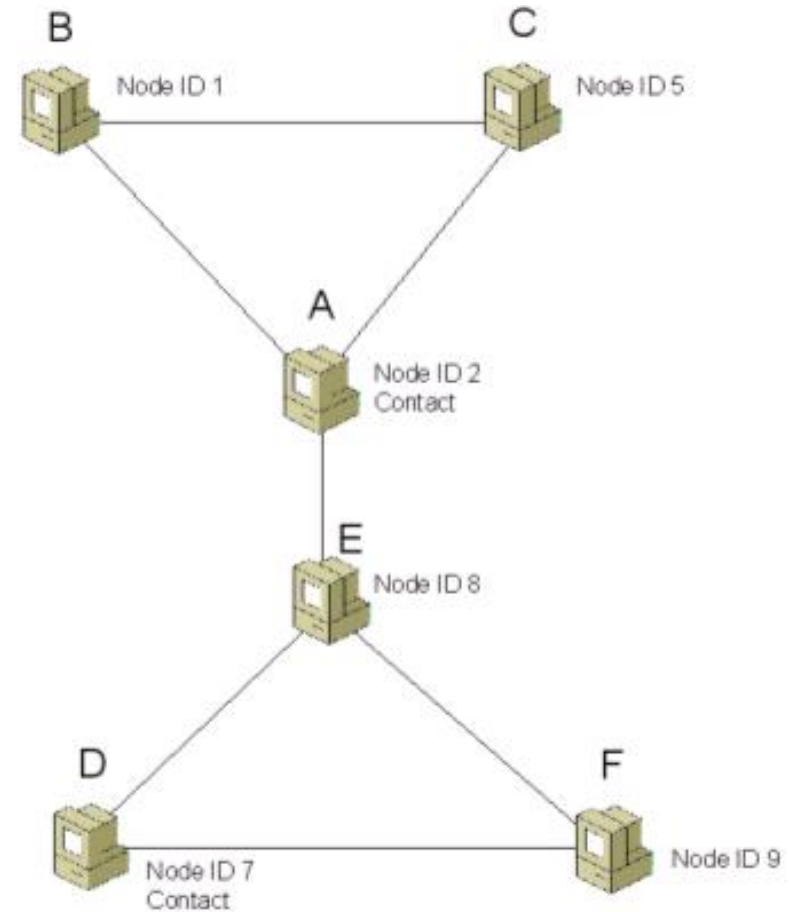
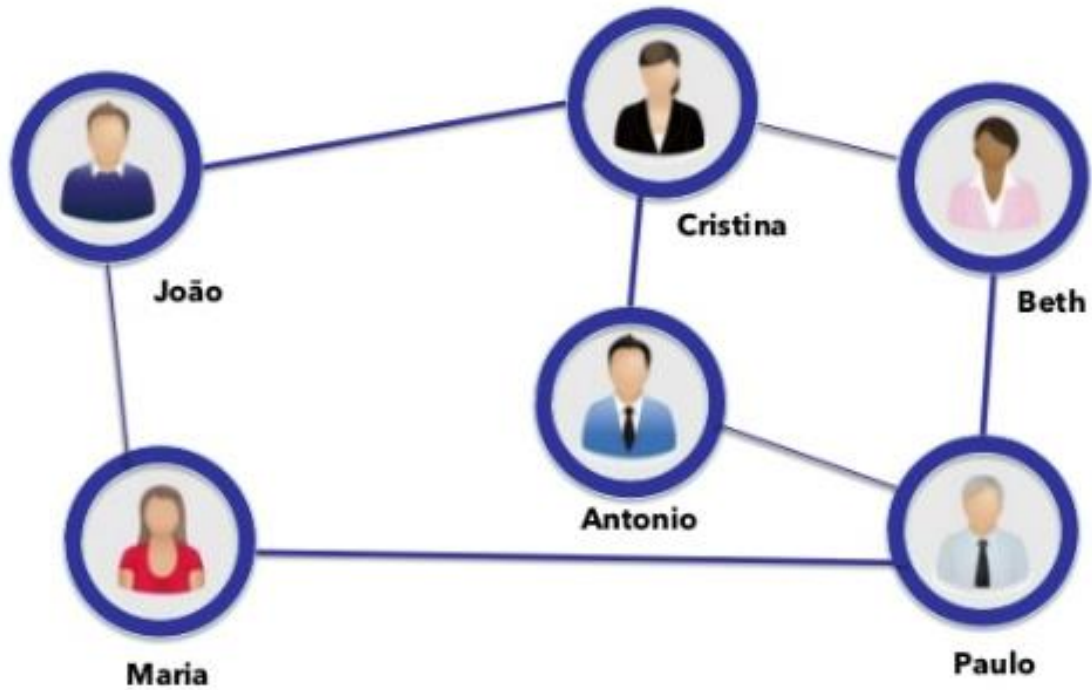
14: GRAFOS

INTRODUCCIÓN

Semana 14 - Grafos



¿QUÉ TIPOS DE ESTRUCTURAS OBSERVA?



RECORDEMOS



- ¿Qué es un árbol?
- ¿Qué tipos de árboles conoce?
- ¿Cuál es la diferencia entre un árbol binario y un árbol de búsqueda binaria?
- ¿De qué manera podemos recorrer un ABB?
- ¿Cuál de los recorridos muestra los elementos en orden
- Dibuje un árbol de expresión para la expresión $(9+4)*5+(4-(6-3))$



¿CÓMO PODRÍA...



- ...representar un grafo haciendo uso de un programa java?

LOGRO ESPERADO



- Al término de la sesión, el estudiante elabora un programa java que permita registrar y mostrar los datos de un grafo, haciendo uso de una matriz de adyacencia.



DESARROLLO DEL TEMA

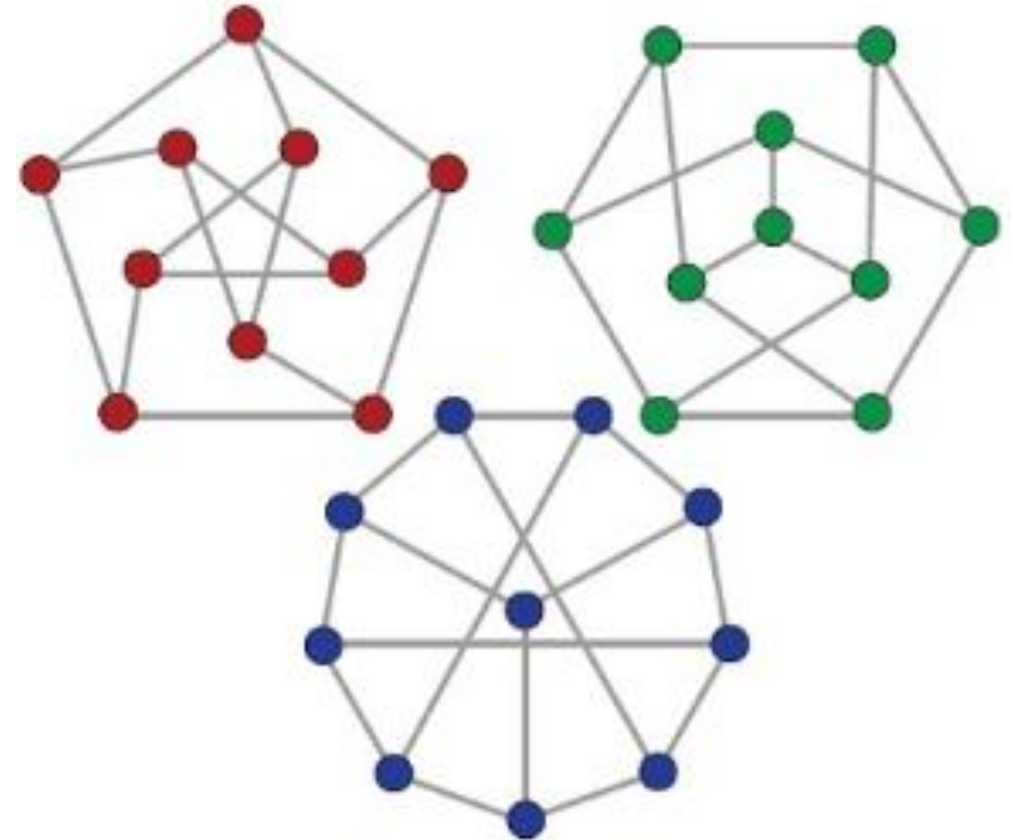
Estructura de Datos



DEFINICIÓN

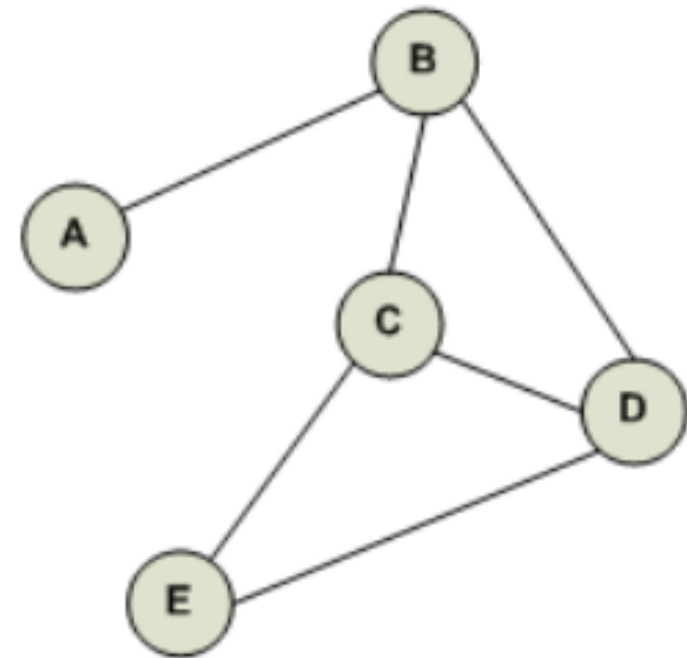


- Los grafos y la teoría de grafos son subdisciplinas propias tanto en el campo de las Matemáticas como en el de la Informática
- Un grafo en el ámbito de las ciencias de la computación es una estructura de datos, en concreto un tipo abstracto de datos (TAD), que consiste en un conjunto de nodos (también llamados vértices) y un conjunto de arcos (aristas) que establecen relaciones entre los nodos.



GRAFO NO DIRIGIDO

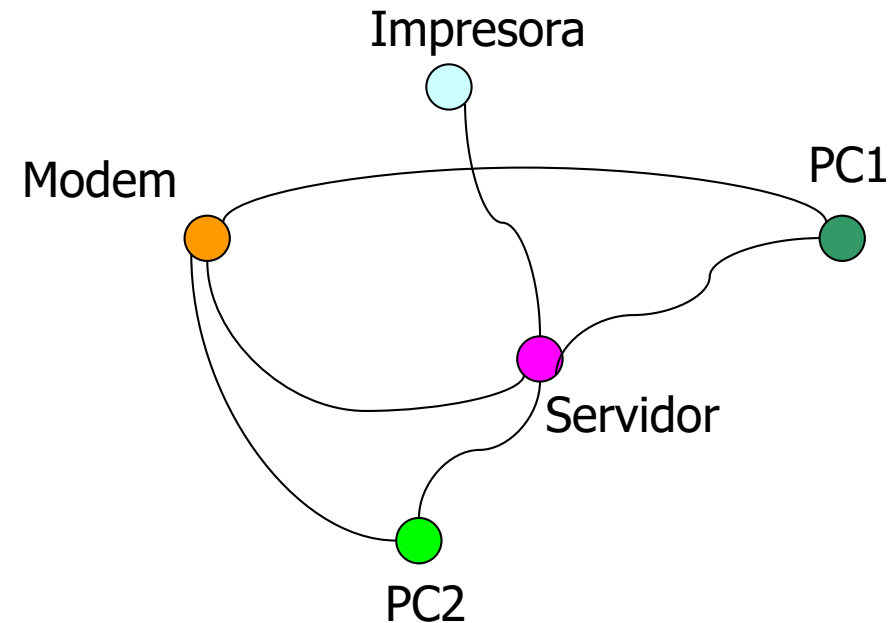
- En terminología de grafos, los **nodos se denominan vértices y las conexiones entre ellos se denominan aristas.**
- Los vértices se suelen designar mediante un nombre o una etiqueta.
 - Por ejemplo, podríamos etiquetar los vértices como A, B, C, D y E.
- A las aristas se las referencia indicando la pareja de vértices que conectan.
 - Por ejemplo, podríamos tener una arista (A, B), lo que significa que existe una arista entre el vértice A y el vértice B.



NODOS O VÉRTICES



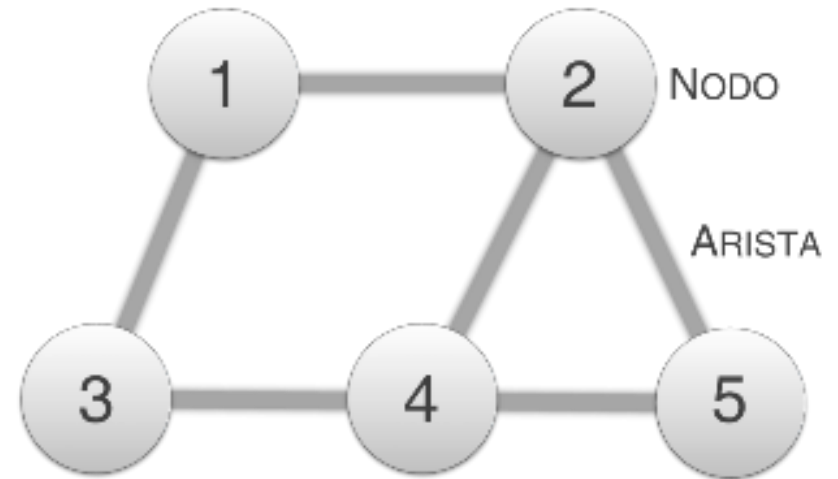
- Son los puntos o nodos con los que esta conformado un grafo.
- Llamaremos grado de un vértice al número de aristas de las que es extremo. Se dice que un vértice es par o impar según lo sea su grado.



ARISTAS

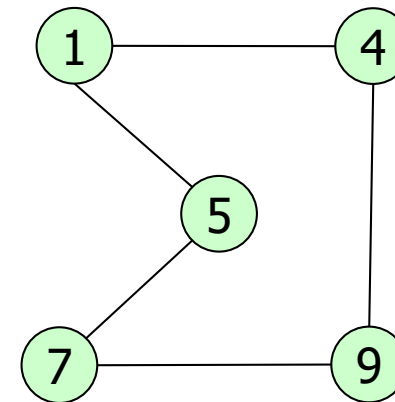


- Son las líneas con las que se unen os nodos de un grafo y con la que se construyen también caminos.



VÉRTICES Y ARISTAS

- Un grafo $G = (V, A)$
- V , el conjunto de vértices o nodos
 - Representan los objetos
- A , el conjunto de aristas (arcos)
 - Representan las relaciones



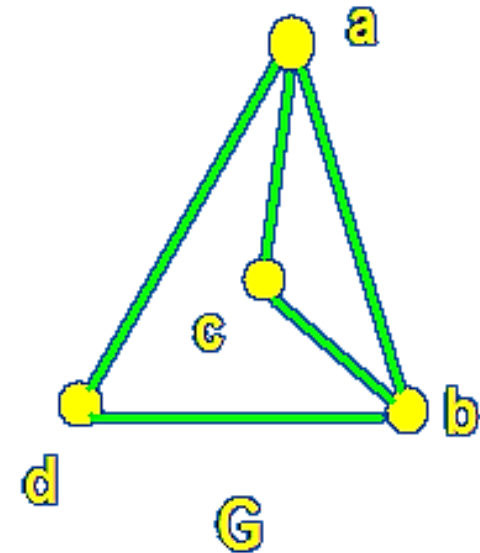
$$V = \{1, 4, 5, 7, 9\}$$

$$A = \{(1,4), (5,1), (7,9), (7,5), (4,9), (4,1), (1,5), (9,7), (5, 7), (9,4)\}$$

ENTONCES...

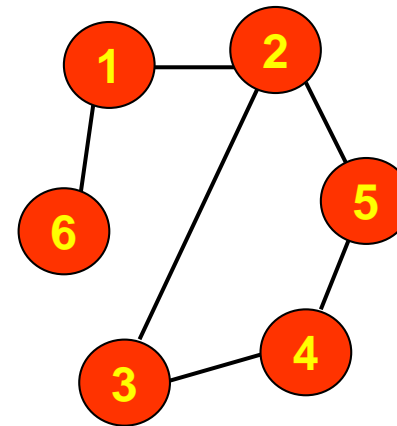
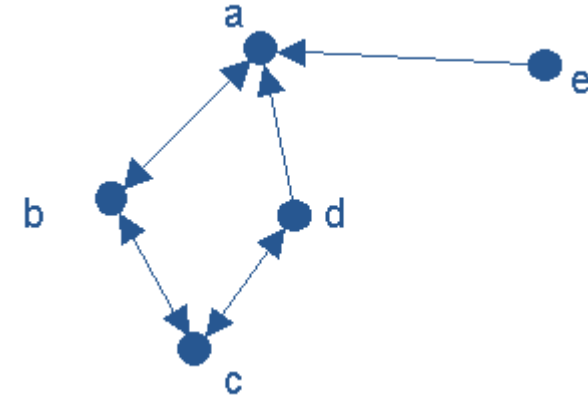


- Un grafo G es un par (V,A) donde:
 - $V = \{v_1, \dots, v_n\}$ es un conjunto de vértices
 - $A = \{a_1, \dots, a_m\}$ es un conjunto de aristas, con cada:
 - $a_k \in \{v_i, v_j\}$, con $v_i, v_j \in V, v_i \neq v_j$
- Los vértices se representan como puntos y las aristas como líneas entre vértices
- Ejemplo:
 - $G = (V,E)$
 - $V = \{a,b,c,d\}$
 - $E = \{\{a,b\}, \{b,c\}, \{a,c\}, \{a,d\}, \{d,b\}\}$



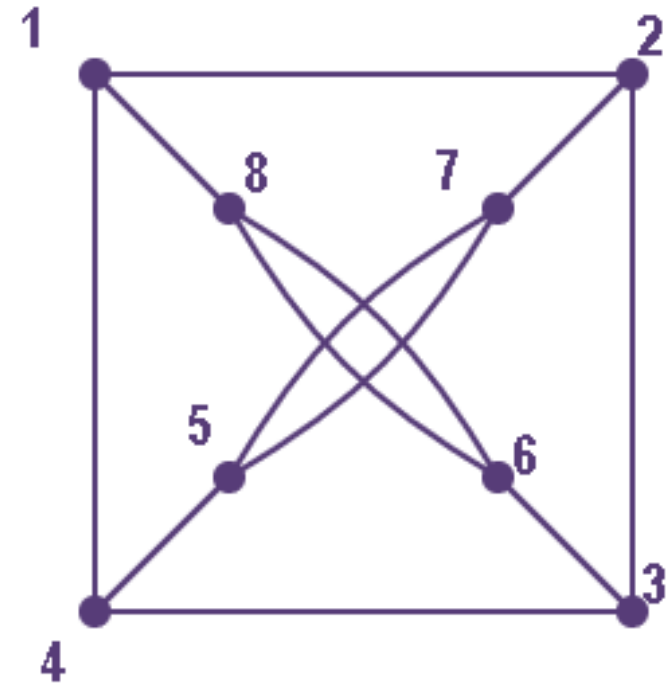
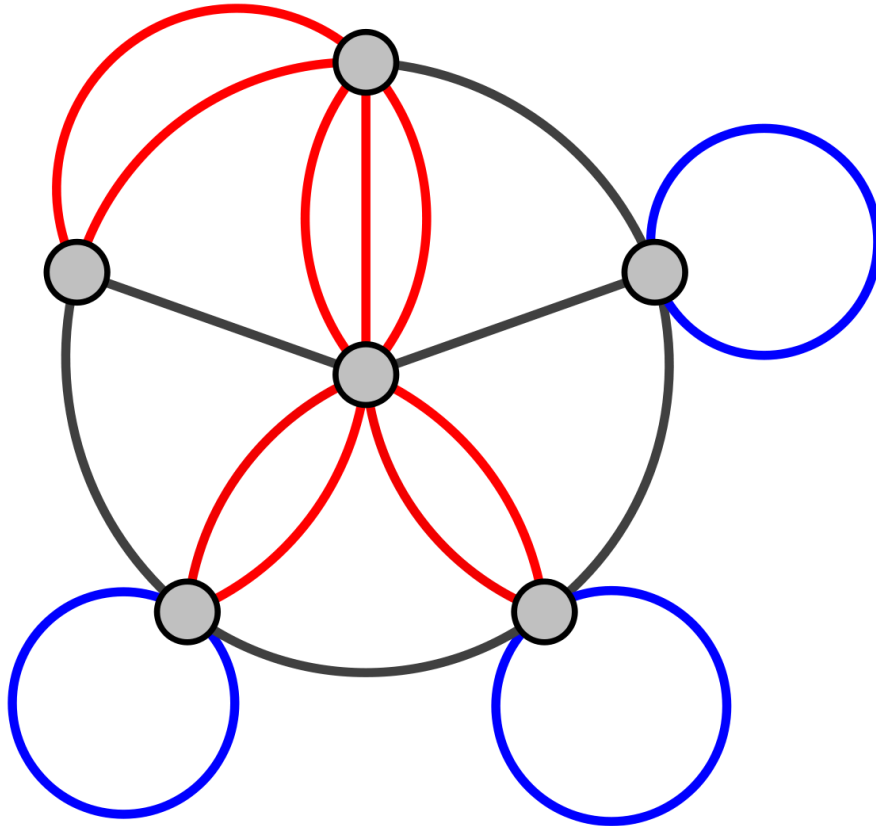
TIPOS DE GRAFOS: GRAFOS DIRIGIDOS

- Si el orden influye en la aristas se habla de **grafos dirigidos**:
 - En este caso a las aristas se les llama arcos y se representan como pares para indicar el orden:
 - $V = \{ a, b, c, d, e \}$
 - $A = \{ (e,a), (a,b), (b,a), (d,a), (c,d), (d,c), (b,c), (c,b) \}$
- **No dirigidos**: El par de vértices que representa un arco no está ordenado.



TIPOS DE GRAFOS: MULTIGRAFOS

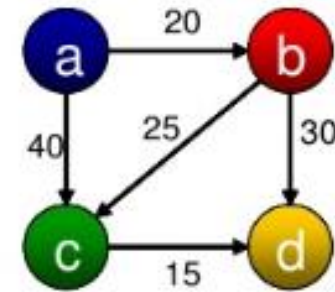
- Si se permite que haya más de una arista se habla de multigrafos



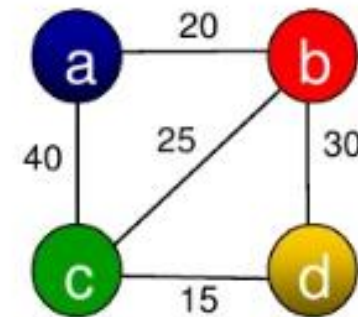
Un multigrafo con múltiples aristas (en rojo) y tres bucles (en azul). No todos los autores permiten multigrafos con bucles.

TIPOS DE GRAFOS: GRAFOS VALORADOS

- Cuando las aristas tienen un valor numérico asociado se llama de grafos valorados:
- Al valor numérico asociado se le llama coste de la arista o factor de peso



Grafo Dirigido con Costo

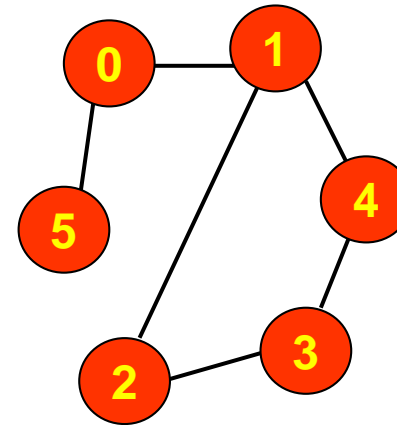


Grafo No Dirigido con Costo

REPRESENTACIÓN DE GRAFOS

MATRIZ DE ADYACENCIA

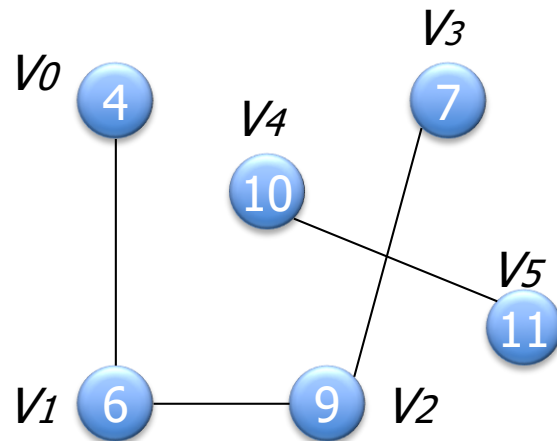
- Para representar los grafos a menudo se utiliza la llamada matriz de adyacencia
- Se construye imaginando que en las filas y las columnas corresponden a los vértices. Se pone un 0 para indicar que 2 vértices no son adyacentes, y un 1 para indicar que sí lo son:



	0	1	2	3	4	5
0	0	1	0	0	0	1
1	1	0	1	0	1	0
2	0	1	0	1	0	0
3	0	0	1	0	1	0
4	0	1	0	1	0	0
5	1	0	0	0	0	0

MATRIZ DE ADYACENCIA

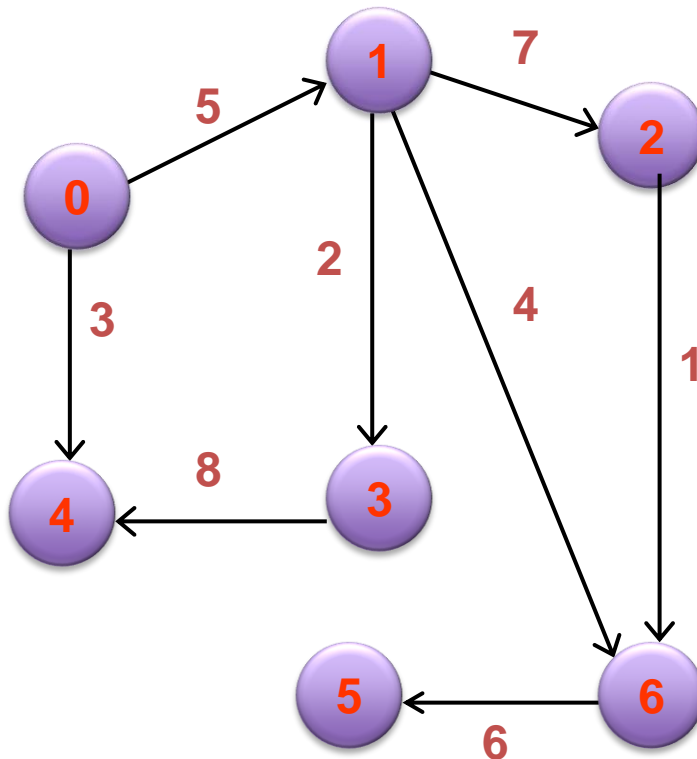
- Si el grafo fuese valorado, en vez de 1, se coloca el factor de peso
- En un grafo dirigido, se supone que la fila representa el vértice origen, y la columna el vértice destino del arco



	V_0	V_1	V_2	V_3	V_4	V_5
V_0	0	1	0	0	0	0
V_1	1	0	1	0	0	0
V_2	0	1	0	1	0	0
V_3	0	0	1	0	0	0
V_4	0	0	0	0	0	1
V_5	0	0	0	0	1	0

MATRIZ DE COSTOS

- Es una matriz en la cual se almacenan los costos.



- ∞ es el símbolo para un número inmensamente grande en computadoras se pueden utilizar el MAX_INT, o el limite de los números flotantes.

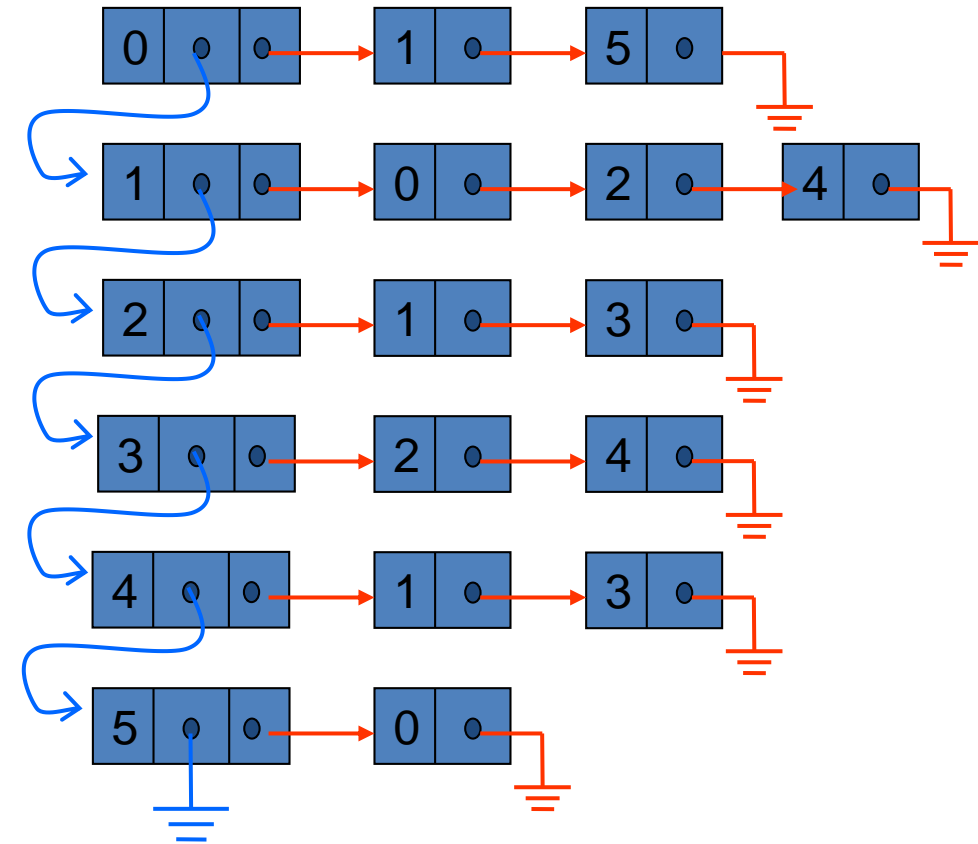
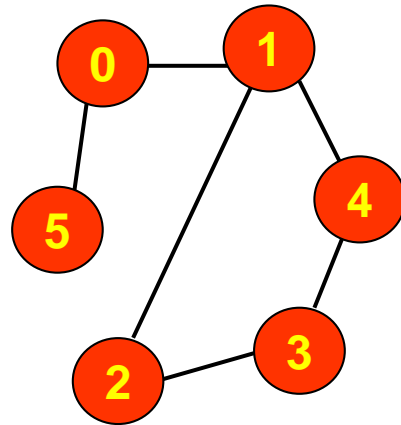
	0	1	2	3	4	5	6
0	∞	5	∞	∞	3	∞	∞
1	∞	∞	7	2	∞	∞	4
2	∞	∞	∞	∞	∞	∞	1
3	∞	∞	∞	∞	8	∞	∞
4	∞	∞	∞	∞	∞	∞	∞
5	∞	∞	∞	∞	∞	∞	∞
6	∞	∞	∞	∞	∞	6	∞

REPRESENTACIÓN DE GRAFOS



LISTA DE ADYACENCIA

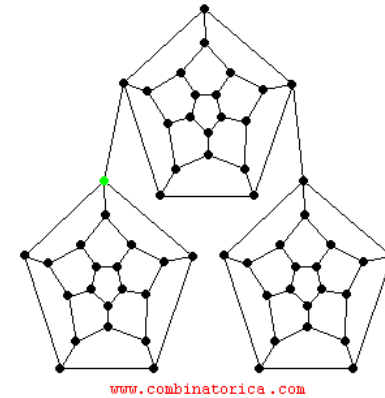
- Es una lista de los vértices, donde cada vértice apunta a una lista de los vértices adyacentes.



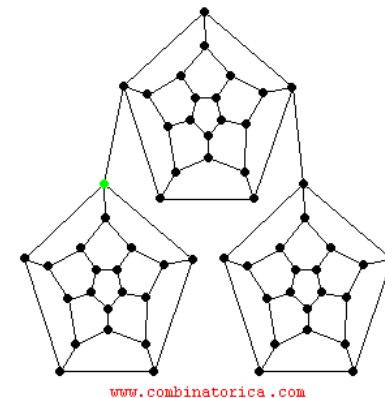
RECORRIDO DE UN GRAFO

- Recorrer un grafo significa tratar de alcanzar todos los nodos a partir de un nodo (inicio) hacia otro nodo (fin)
- Tipos de recorrido:
 - Recorrido en profundidad (DFS Depth First Search). Equivalente a recorrer en preorden en un árbol.
 - Recorrido en anchura (BFS Breadth First Search). Equivalente al recorrido por niveles en un árbol.

Breadth-First Search



Depth-First Search

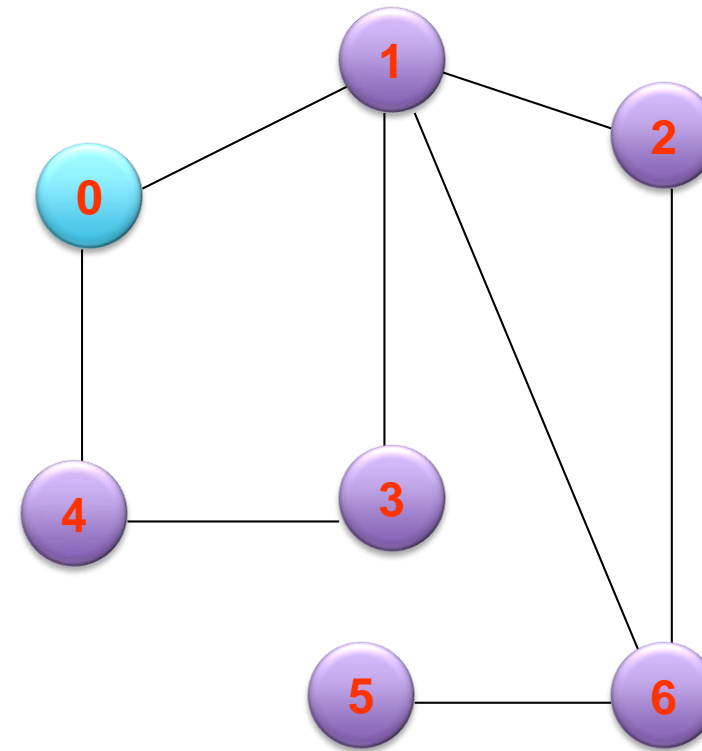


RECORRIDO EN PROFUNDIDAD (DFS: DEPTH FIRST SEARCH BÚSQUEDA PRIMERO EN PROFUNDIDAD)



- Busca los caminos que parten desde un nodo hasta que ya no es posible avanzar más, en donde vuelve atrás en busca de caminos alternativos, que no se recorrieron previamente.
- Este algoritmo es el mismo tanto para grafos no dirigidos como para grafos dirigidos.

- El recorrido será:
 - 0 1 2 6 5 3 4



RECORRIDO EN PROFUNDIDAD (DFS: DEPTH FIRST SEARCH BÚSQUEDA PRIMERO EN PROFUNDIDAD)



BUSQUEDA EN PROFUNDIDAD DFS

- Se comienza con un nodo u .
- Por cada nodo se expande de manera recursiva en un camino concreto.
- Cuando no es posible visitar más nodos por ese camino, regresa.
- Se repite el proceso con cada uno de los nodos hermanos del nodo ya procesado.



- método DFS(nodo_origen):
- marcamos origen como visitado
- para cada vertice v adyacente a origen en el grafo:
- si v no ha sido visitado:
- marcamos como visitado v
- llamamos recursivamente DFS(v)

RECORRIDO EN PROFUNDIDAD



```
void dfs(int m[][LIM], int n, int nodos[]){
    int visitado[LIM];
    for(int i=0; i<n; i++){    // marca a todos como no visitados
        visitado[i]=0;
    }

    int ordenVisita = 0;
    for (int i=0; i<n; i++){    // p cada nodo
        if (visitado[i]==0){
            visitar(m, n, i, visitado, ordenVisita, nodos);
        }
    }
}
```

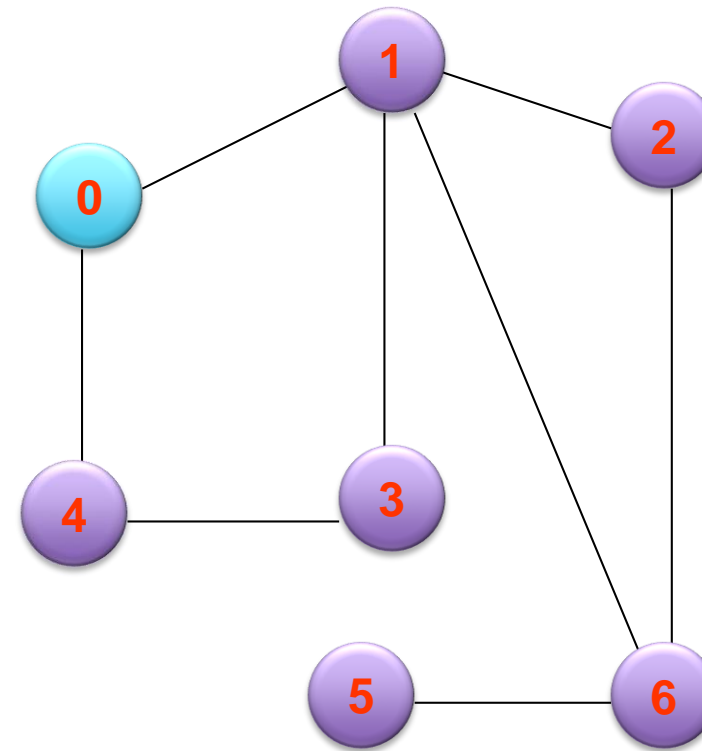
```
void visitar(int m[][LIM], int n, int k, int visitado[], int &ordenVisita, int nodos[]){
    nodos[ordenVisita] = k;
    ordenVisita++;
    visitado[k]=ordenVisita;    // lo marco como visitado en ese orden
    for (int i=0; i<n; i++){
        if (m[k][i] && visitado[i]==0){
            visitar(m, n, i, visitado, ordenVisita, nodos);
        }
    }
}
```


RECORRIDO EN AMPLITUD O ANCHURA (BFS: BREADTH FIRST SEARCH BÚSQUEDA PRIMERO EN ANCHURA)



- Recorre un grafo, a partir de un nodo dado, en niveles, es decir, primero los que están a un arco de distancia del nodo de salida, después los que están a dos arcos de distancia, y así sucesivamente hasta alcanzar todos los nodos a los que se pudiese llegar desde el nodo salida.
- Este algoritmo es el mismo tanto para grafos no dirigidos como para grafos dirigidos.

- El recorrido será:
 - 0 1 4 2 3 6 5



RECORRIDO EN AMPLITUD O ANCHURA (BFS: BREADTH FIRST SEARCH BÚSQUEDA PRIMERO EN ANCHURA)



metodo BFS(grafo, nodo_origen):

creamos una cola Q

agregamos el nodo_origen a la cola Q

marcamos nodo_origen como visitado

mientras Q no esté vacío:

 sacamos un elemento de la cola Q (lo llamamos v)

 para cada vertice w adyacente a v:

 si w no ha sido marcado como visitado

 marcamos w como visitado

 insertamos w en la cola Q



BUSQUEDA EN ANCHURA BFS

- Se comienza con un nodo u .
- Se exploran todos los nodos v adyacentes a u .
- Para cada uno de los nodos v , se exploran sus respectivos adyacentes (que no hayan sido visitados antes).
- Se continua de esta manera hasta recorrer todo el grafo.

RECORRIDO EN AMPLITUD O ANCHURA (BFS: BREADTH FIRST SEARCH BÚSQUEDA PRIMERO EN ANCHURA)



```
void bfs(int m[][LIM], int n, int nodos[], int origen){
    Cola colita;
    inicializa(colita);

    int visitado[LIM] = {0}; //inicializa los ndos como no visitados
    int ordenVisita = 0;

    visitado[origen] = 1; //se marca el origen como visitado
    encolar(colita,origen);

    while(!estaVacia(colita)){
        int v = desencolar(colita);
        nodos[ordenVisita] = v;
        ordenVisita++;

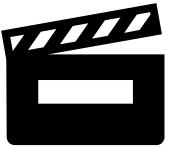
        for(int i= 0; i<n; i++){
            if(m[v][i] == 1 && visitado[i] == 0){
                visitado[i] = 1;
                encolar(colita,i);
            }
        }
    }
}
```

MATERIAL DE CONSULTA DE EXPOSICIONES



- ✓ **FERNÁNDEZ HERRERA, JORDAN**
- ✓ **MALIMBA MINCHAN, VICTOR MANUEL**
- ✓ **TAMAY NUÑEZ, JOSÉ UBELSER**

SAUCEDO RAICO FERNANDO
TAFUR VALDIVIA ROBERTO JHULIÑO
TORRES HOYOS JHON CRISTHIAN



Revisar a partir del minuto 12:30''



Estructuras de Datos en Java 4ed - Mark Allen Weiss

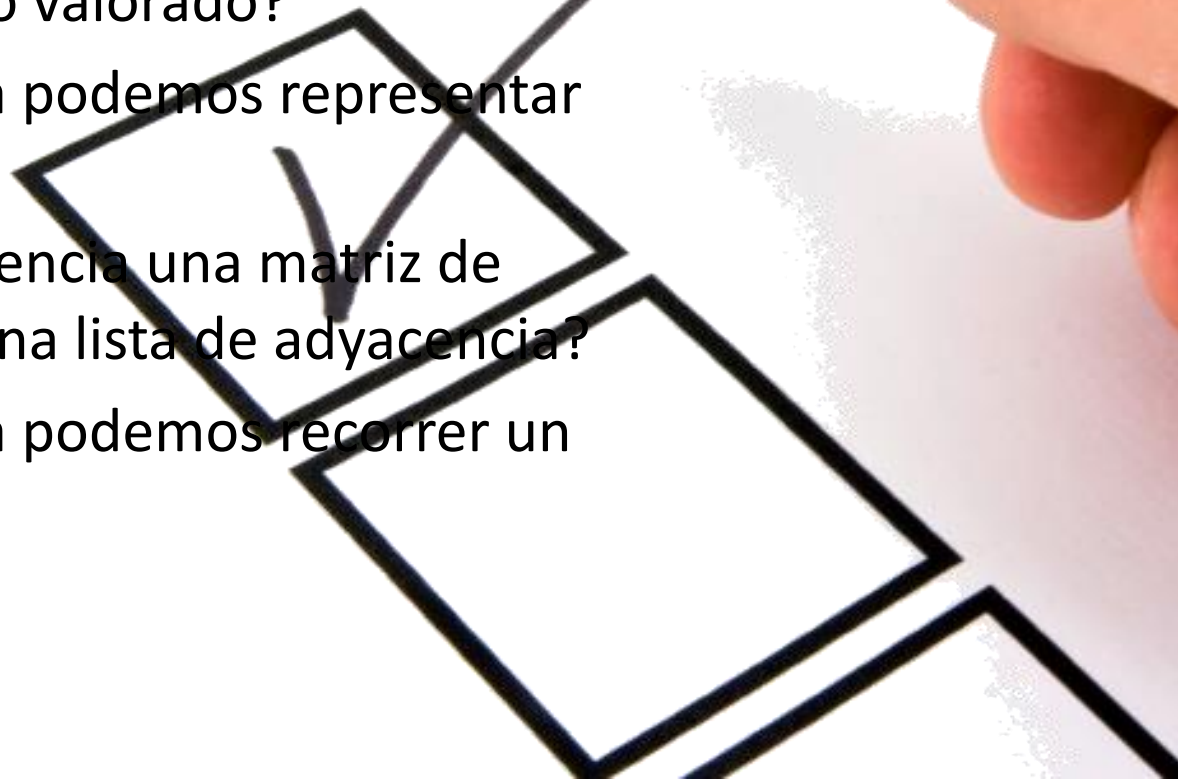
EVALUACIÓN DEL TEMA DESARROLLADO

Reflexionemos!



RECORDANDO LO APRENDIDO

- ¿Qué es un grafo?
- ¿Qué es un vértice y qué es un arista?
- ¿Cómo podemos clasificar a los grafos?
- ¿Qué es un grafo valorado?
- ¿De qué manera podemos representar un grafo?
- ¿En qué se diferencia una matriz de adyacencia de una lista de adyacencia?
- ¿De qué manera podemos recorrer un grafo?



EJERCICIOS DE APLICACIÓN



EJERCICIOS DE APLICACIÓN



- Describa qué alternativas se tiene para determinar si un elemento se encuentra en un grafo.
- Escriba un programa que determine si un elemento forma parte del grafo.