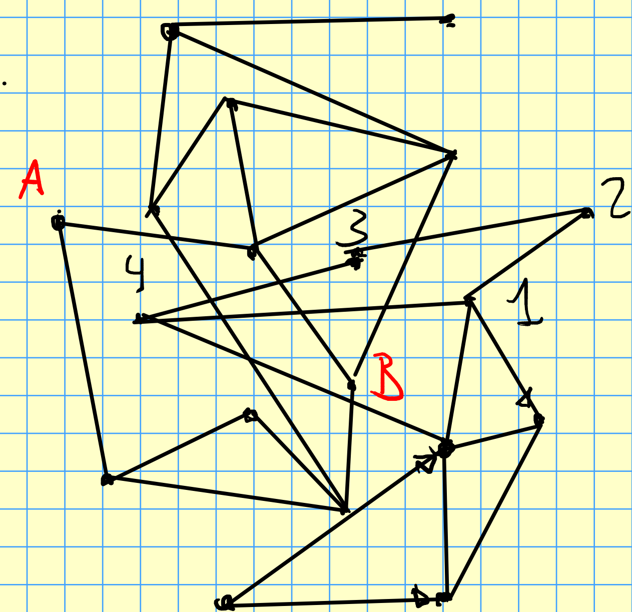
Unidad 4: Algoritmos en teoría de gráficas.

Unidad 1.

- Componentes conexas.
- Trayectorias
- Distancia
- Ciclos.
- Gráficas bipartitas.
- Independientes.
- Coloraciones.

Unidad 2.

Ordenamiento.



Ideas:

- Estructuras de datos para gráficas.
- Búsqueda en anchura
- Búsqueda en profundidad.
- Distancias mínimas.
- Ordenamientos topológicos.
- Cortes y flujos
- Problemas difíciles, P vs. NP, reducciones.

Listas de adyacencia
vs
Matrices de adyacencia.

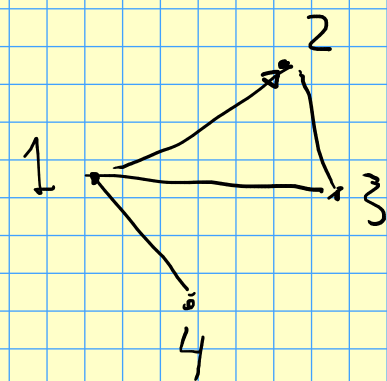
Listas de adyacencia.

V lista de vértices.

$V = [1, 2, \dots, n]$

Para cada j, construimos una lista

N_j = vecinos de j.



$\binom{n}{2} = \frac{n(n-1)}{2}$

$m = O(n^2)$

$V = [1, 2, 3, 4] \rightsquigarrow O(n)$

$$\left.\begin{aligned} N_1 &= [2,3,4] \\ N_2 &= [1,3] \\ N_3 &= [2,1] \\ N_4 &= [1] \end{aligned}\right\} \begin{array}{c} 2m \\ \parallel \\ O(m) \end{array}$$

$O(n+m)$ espacio.

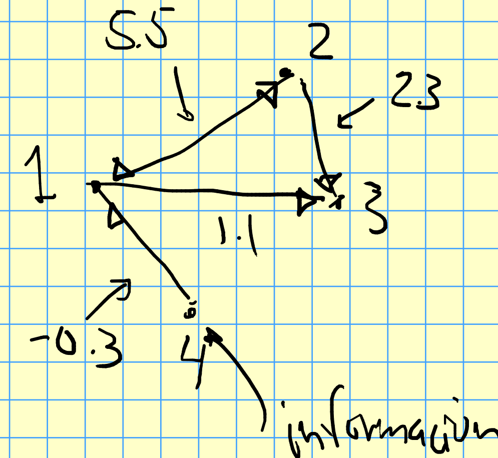
· Para leer el grado de un vértice, se puede implementar en tiempo $O(1)$.

· Leer los vecinos de un vértice v toma tiempo $O(\deg v)$.

Matrices de adyacencia.

Si tenemos n vértices, almacenamos la matriz M de $n \times n$ en donde la entrada i,j de M es

$$m_{ij} = \begin{cases} 0 & \text{si } i \text{ no es adyacente a } j \\ 1 & \text{si } i \text{ es adyacente a } j. \end{cases}$$



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

$M[3][2]$

Determinar adyacencia de dos vértices se hace en tiempo $O(1)$.

$O(n^2)$ espacio.