



UNIVERSITAT DE
BARCELONA



Presentació Assignatura

Algorísmica Avançada | Enginyeria Informàtica

Santi Seguí I 2021-2022



Professor Agregat al departament de Matemàtiques i Informàtica
de la Universitat de Barcelona.

Doctorat amb *Visió per Computador i Machine Learning*
Graduat amb Informàtica

Santi Seguí Email: santi.segui@ub.edu



Pere Gilabert
Estudiant Doctorat

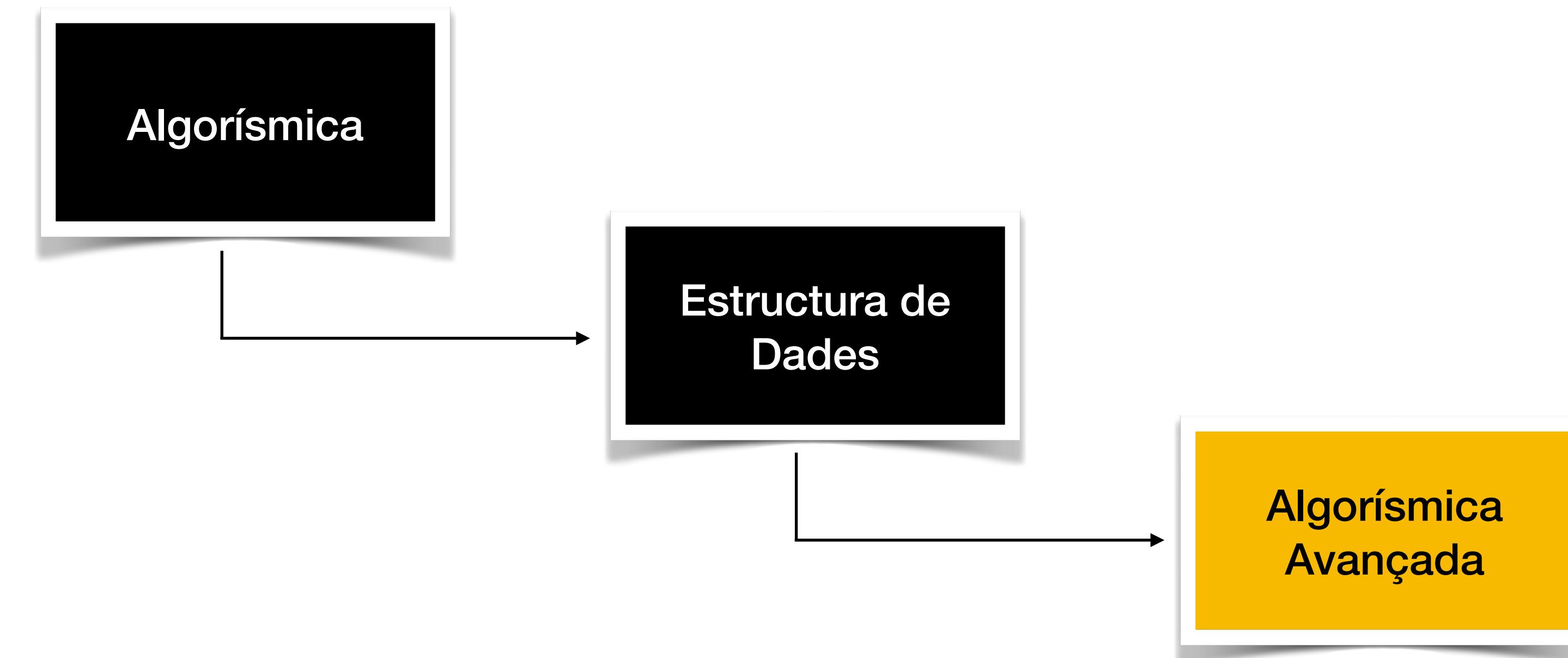
pere.gilabert@ub.edu

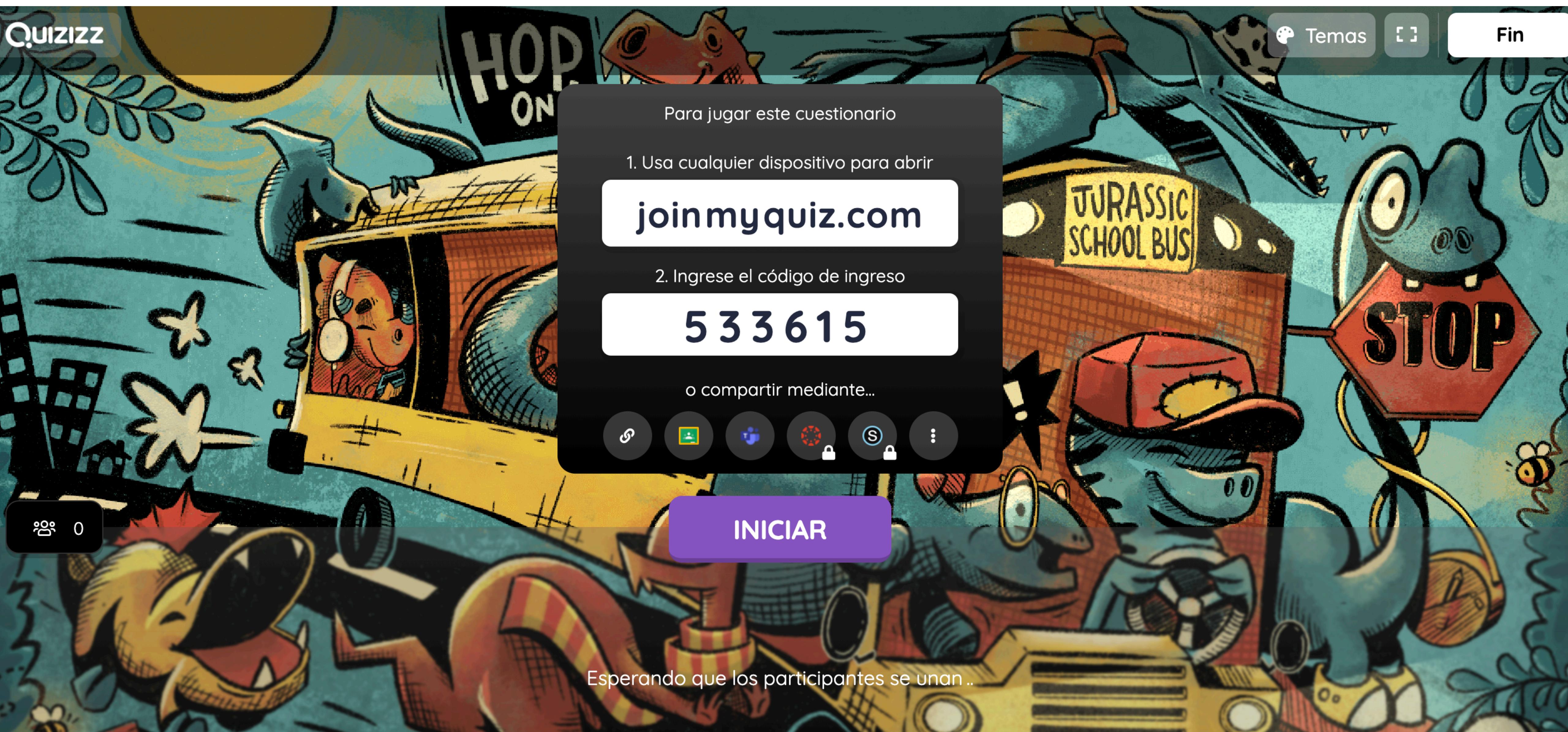


Enric Florit
Professor Associat

xxxx@xxxx.ub.edu

Algorísmica Avançada





Para jugar este cuestionario

1. Usa cualquier dispositivo para abrir

joinmyquiz.com

2. Ingrese el código de ingreso

533615

o compartir mediante...



INICIAR

Esperando que los participantes se unan ..

Quina és la complexitat del següent algoritme:

```
def main(c):
    a=1
    b=4
    d=max(c)
    for i in c:
        a=a+b
    return a
```

$O(2c + 4)$

$O(2c)$

$O(c)$

$O(2c + 3)$

Quines respostes són correctes?

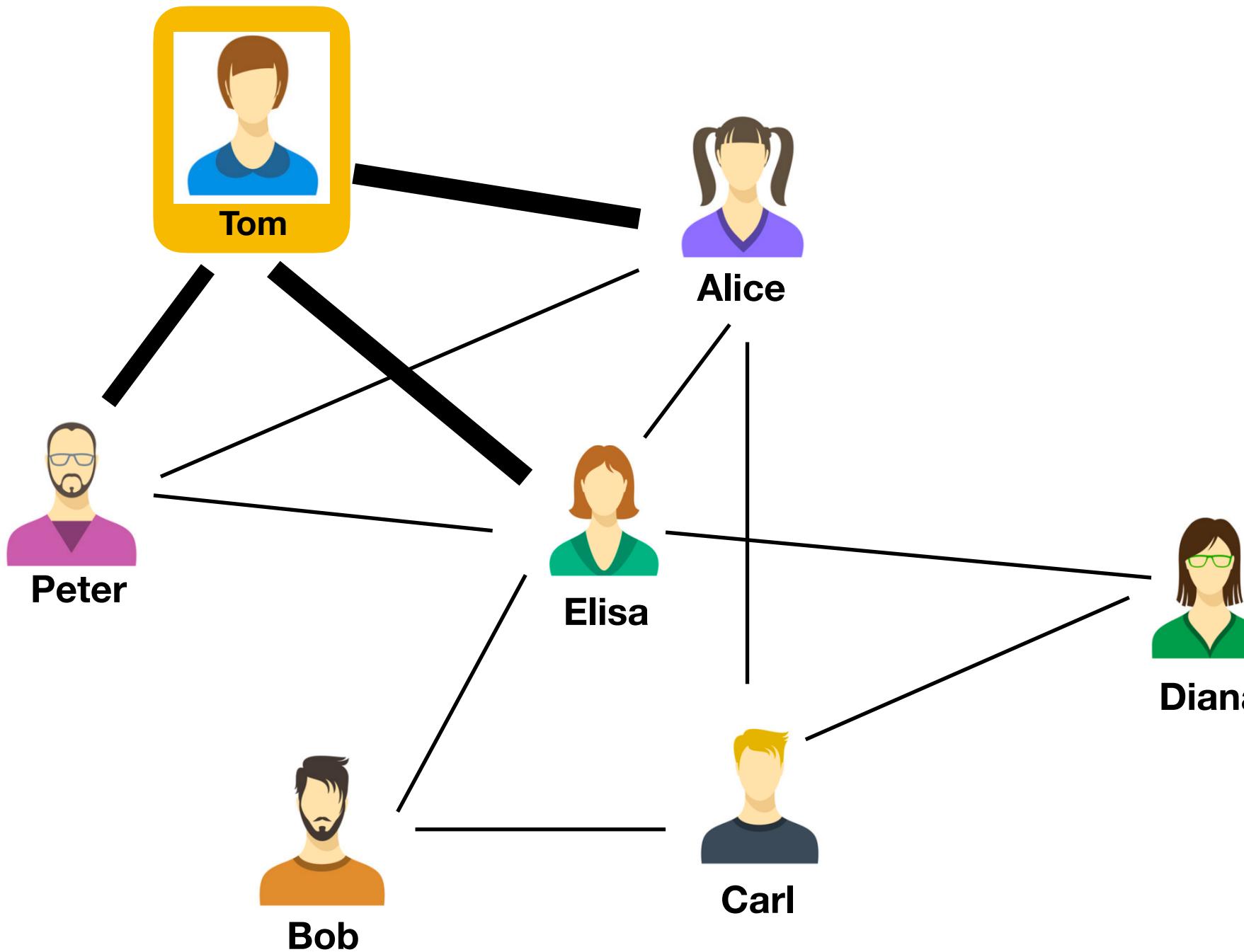
$O(n) < O(n \log n) < O(n^2) < O(n!)$

No podem saber si un algoritme amb complexitat $O(n!)$ és millor que un amb complexitat $O(2^n)$ sense saber el valor de n .

$O(n) < O(2n)$

$O(2^n) < O(n^2)$

Podem suggerir “amics” al TOM?.
Recomanarem aquells que estiguin distància 2 (amics d’amics). Quin algoritme podem utilitzar?

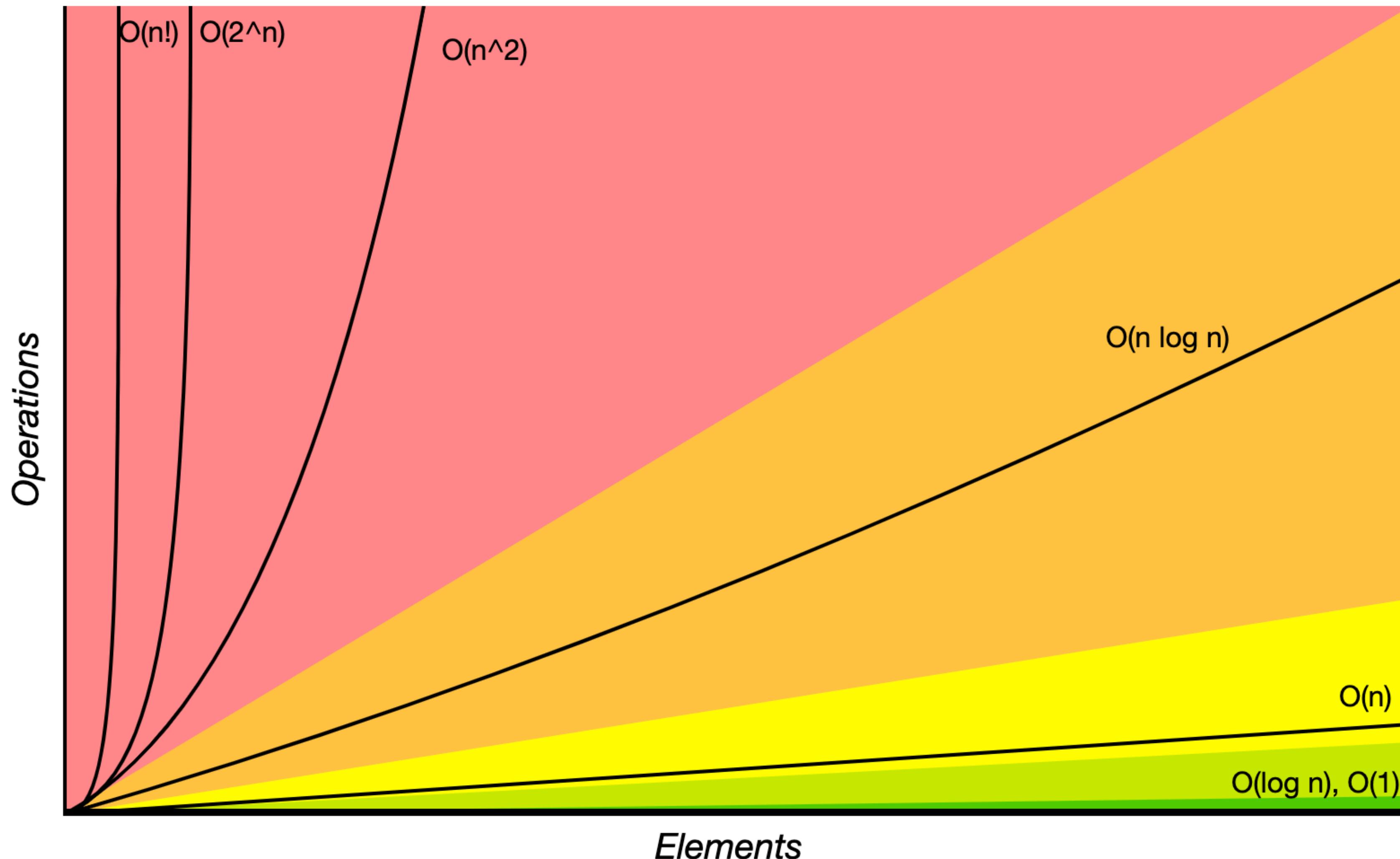


BFS

DFS

Big-O Complexity Chart

Horrible Bad Fair Good Excellent



Notació Asimptòtica

$O(n)$, $\Theta(n)$, $o(n)$ i $\Omega(n)$

Notació Asimptòtica

$O(n)$

conjunt de les funcions que creixen més a poc a poc o igual que n

$\Theta(n)$

conjunt de les funcions que creixen igual que n

$o(n)$

conjunt de les funcions que creixen més a poc a poc que n

$\Omega(n)$

conjunt de les funcions que creixen més de pressa o igual que n

Notació Asimptòtica

```
def main(c):
    a=1
    b=4
    d=max(c)
    for i in c:
        a=a+b
    return a
```

Notació Asimptòtica

```
def main(c):
    a=1
    b=4
    d=max(c)
    for i in c:
        a=a+b
    return a
```

1

1

c

$c^*(1)$

1

1

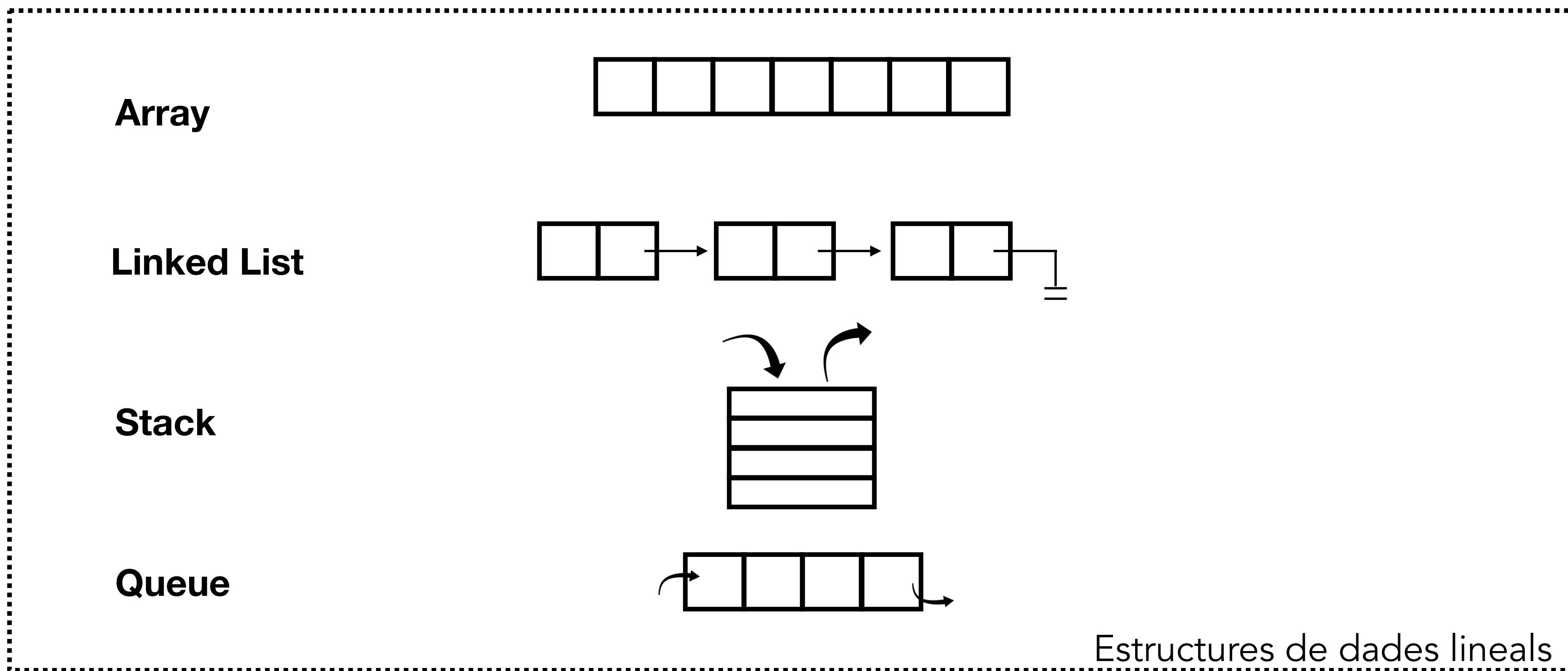
Total = $1+1+c+c+1=2c+3$

$\Theta(c)$

Què és un graf?

Què és un graf?

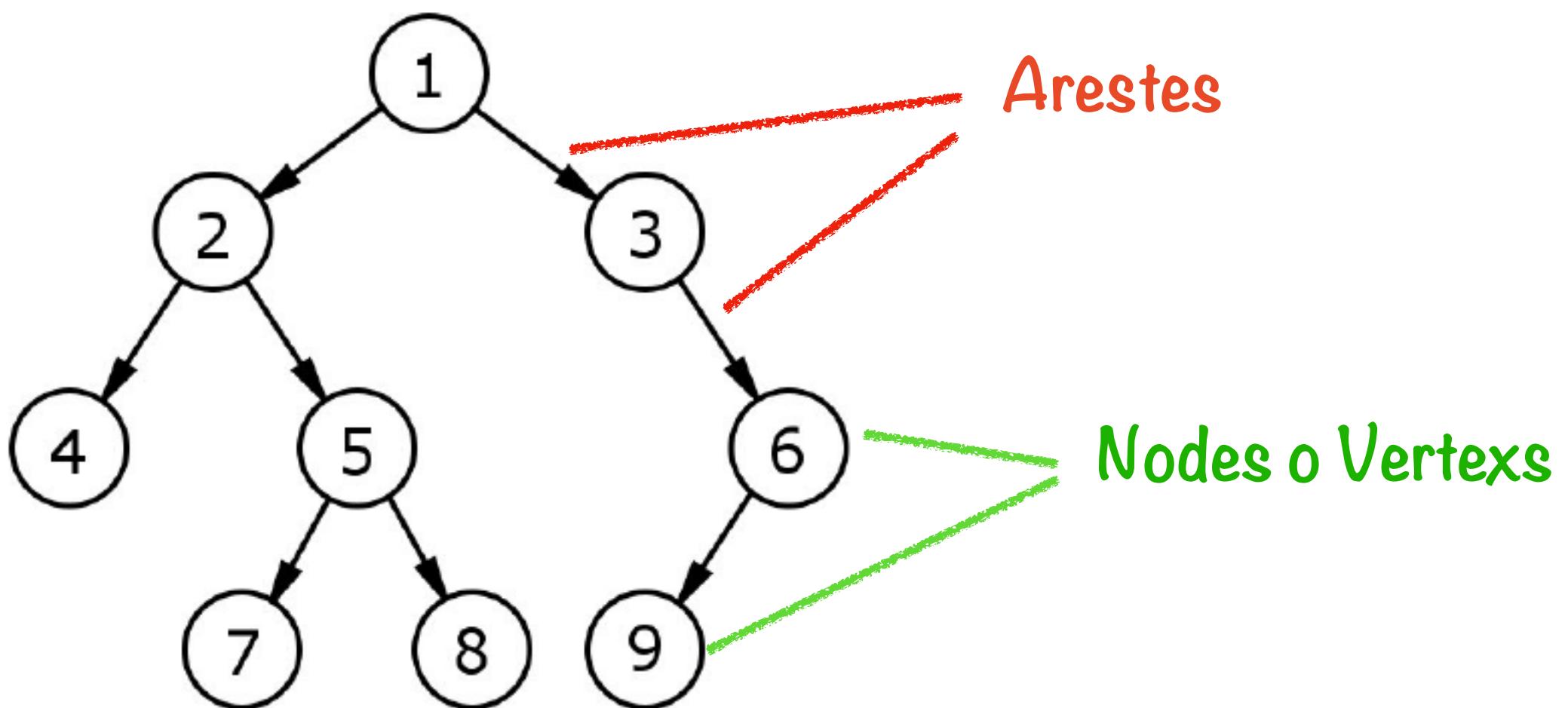
- El graf és una estructura de dades **NO LINEAL**



Què és un graf?

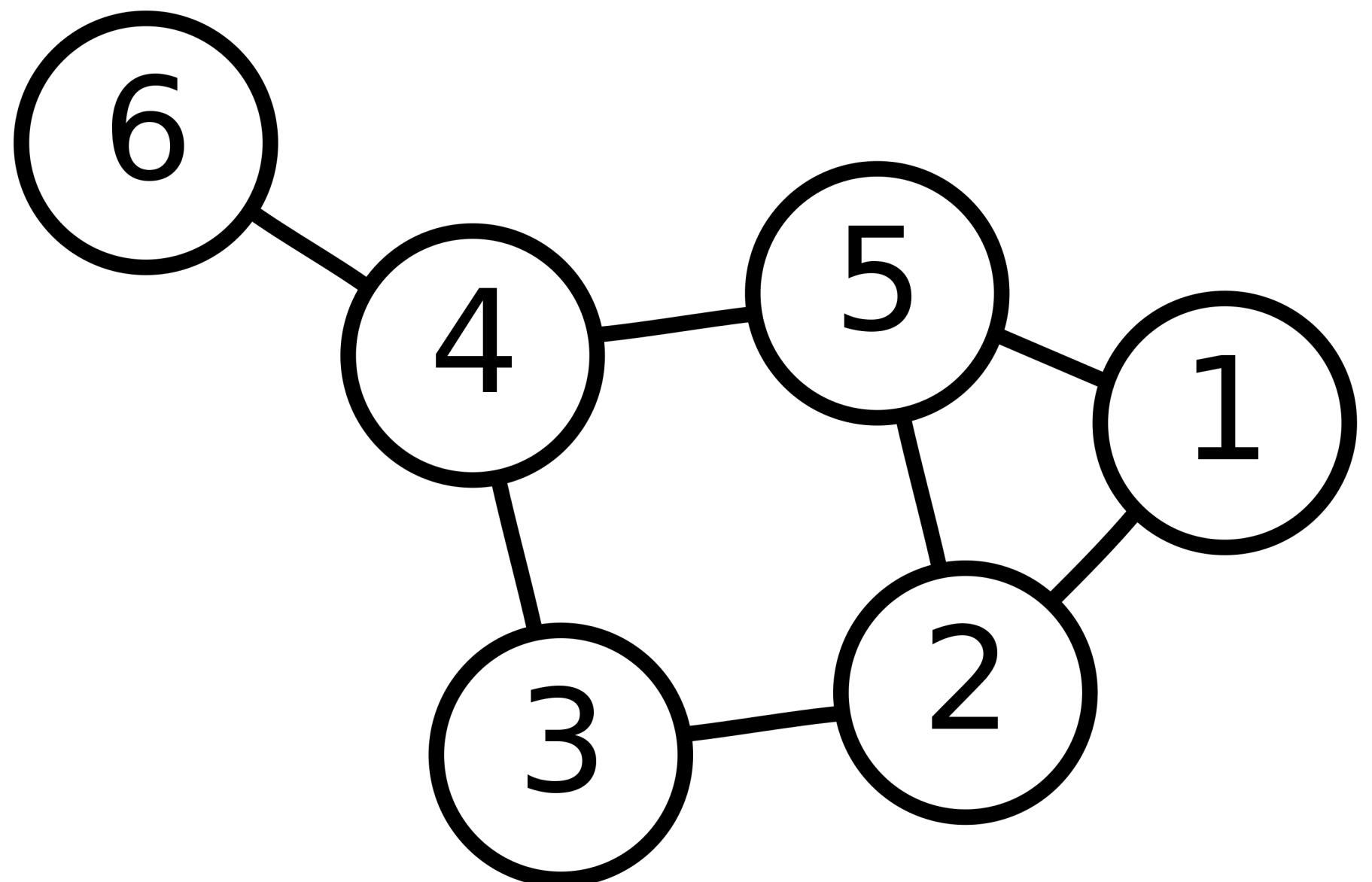
- El graf és una estructura de dades **NO LINEAL**

Quina estructura de dades **NO LINEAL** coneixeu?



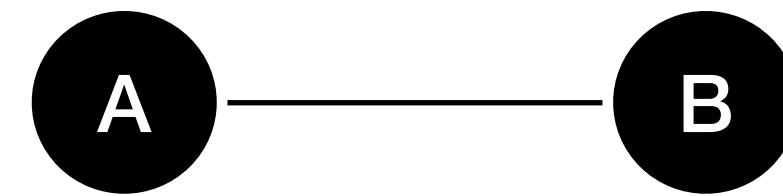
Què és un graf?

- El graf és una estructura de dades **NO LINEAL**

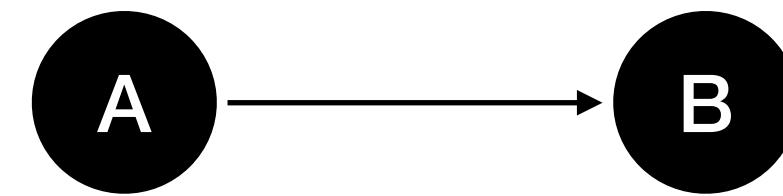


Què és un graf?

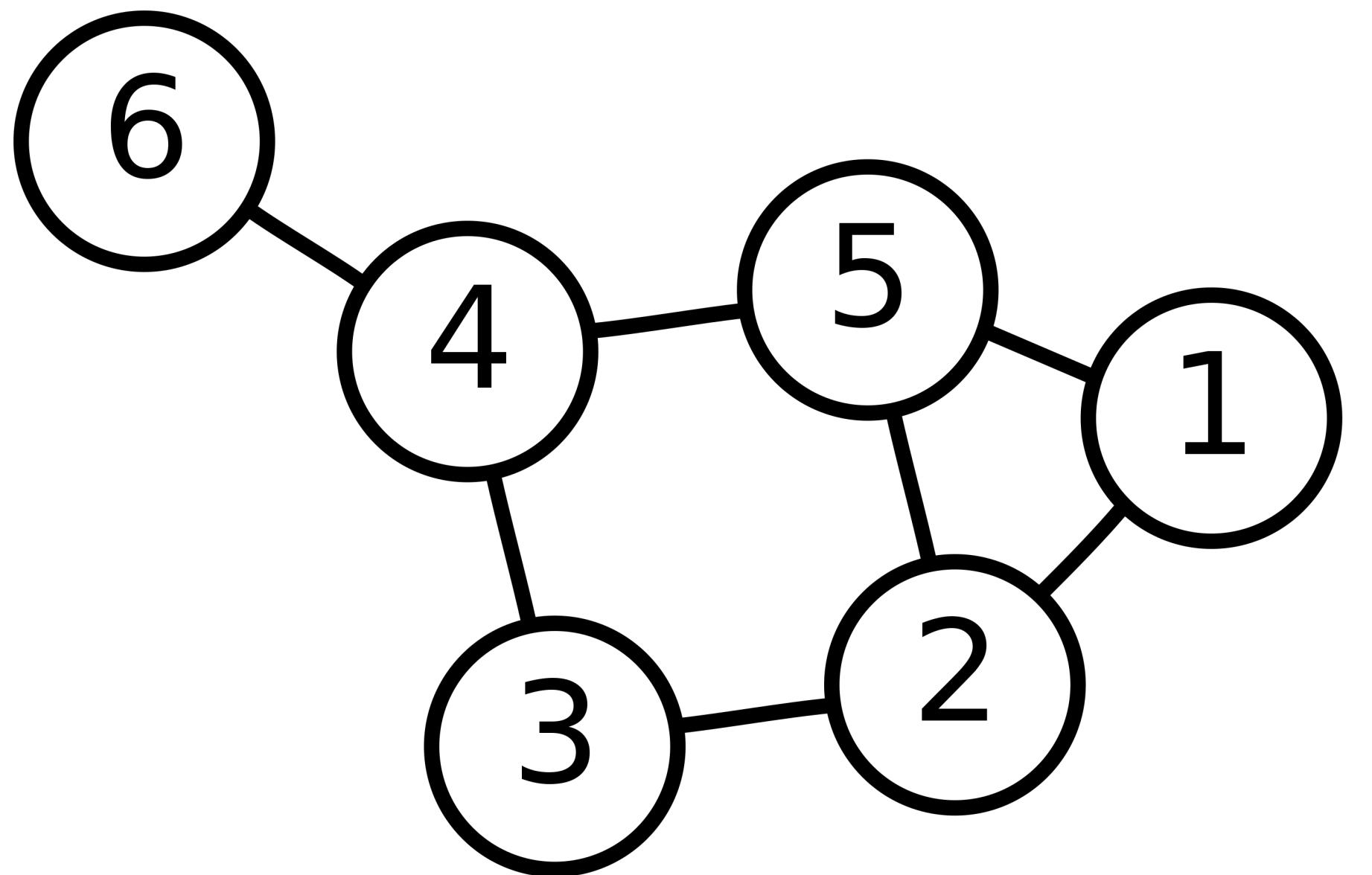
- Un **graf** és un conjunt V de vèrtexs o nodes i un conjunt E de d'arestes o arcs relacionats entre aquests vèrtexs. Els grafs tenen una estructura topològica que ens permet establir relacions espacials entre els vèrtexs que estem estudiant.
- Un **graf** és un conjunt ordenats $G = (V, E)$ on:
 - V és el conjunt de vèrtex, i
 - E és un conjunt d'arestes que relacionen aquests vèrtexs

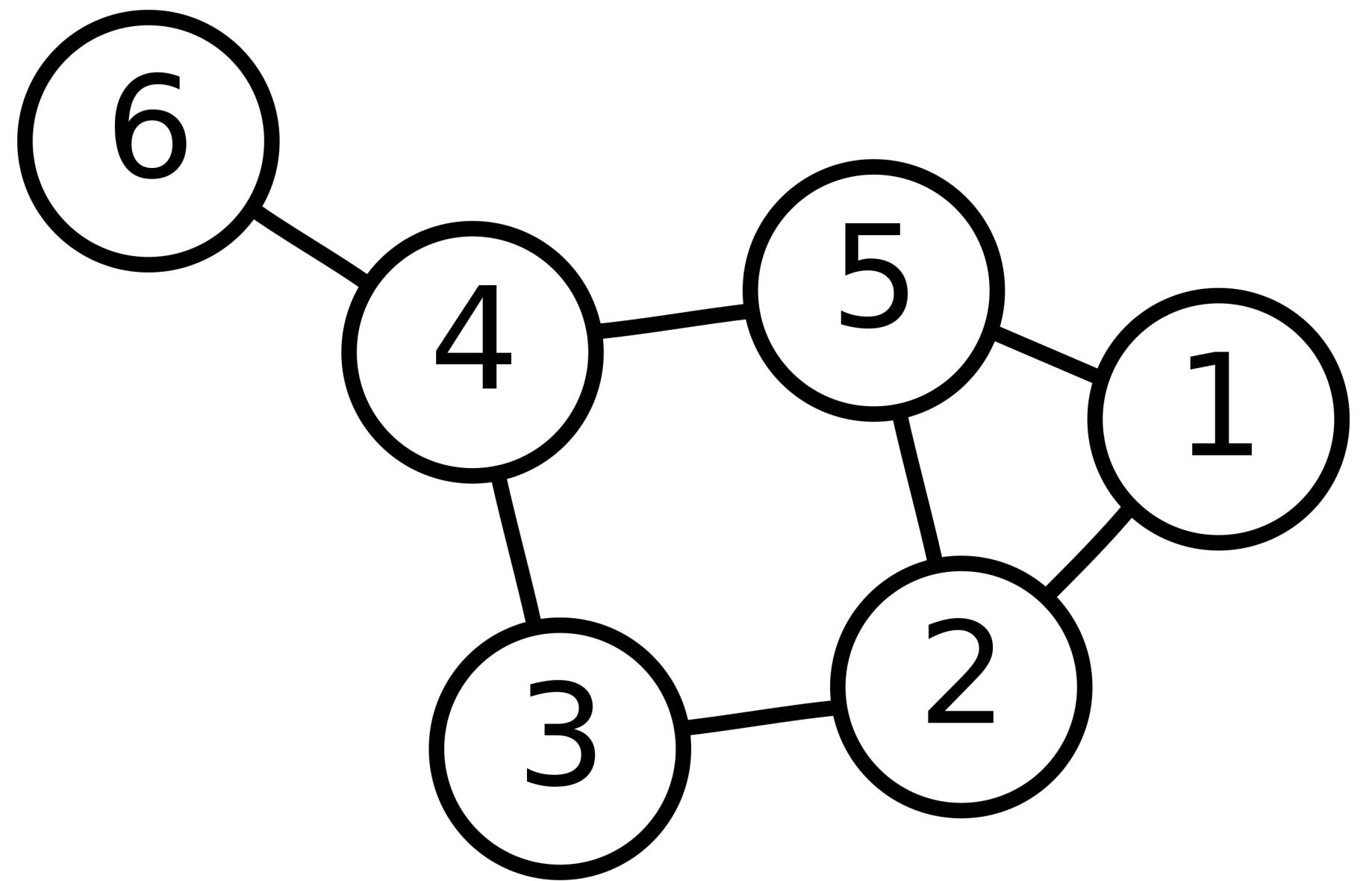


Graf no dirigit



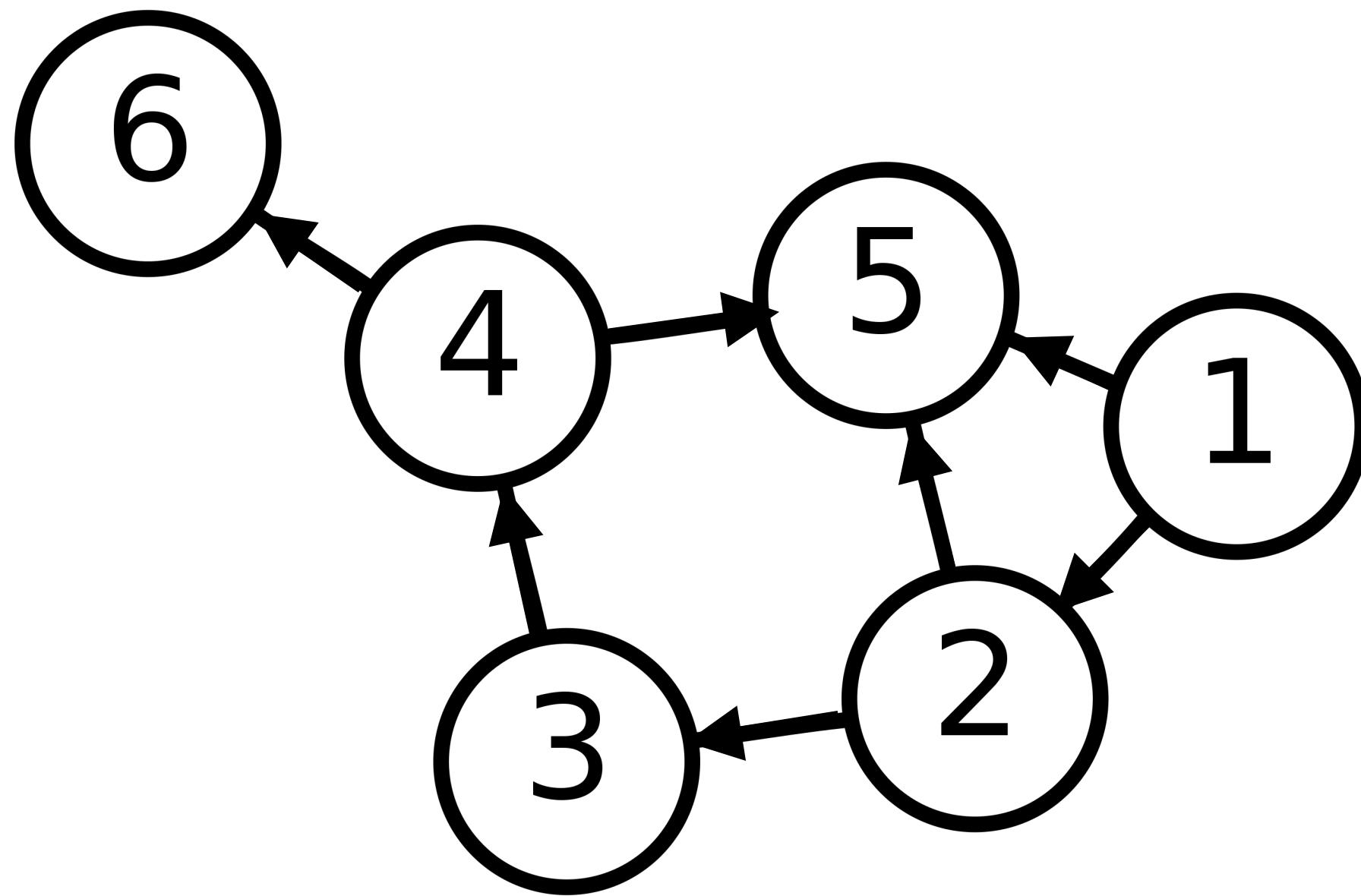
graf dirigit





$$V = \{1, 2, 3, 4, 5, 6\}$$

$$E = \{\{1, 2\}, \{1, 5\}, \{2, 3\}, \{2, 5\}, \{3, 4\}, \{4, 5\}, \{4, 6\}\}$$

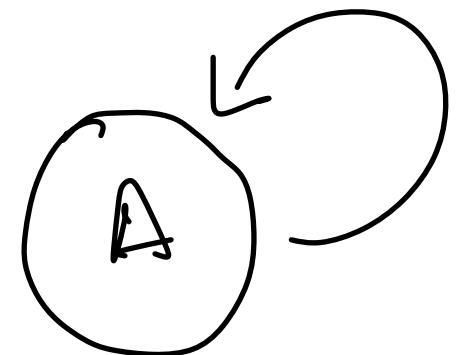


$$V = \{1, 2, 3, 4, 5, 6\}$$

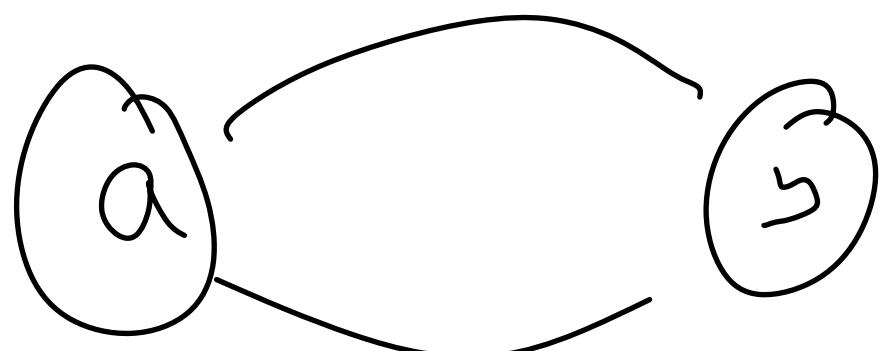
$$E = \{(1, 2), (1, 5), (2, 3), (2, 5), (3, 4), (4, 5), (4, 6)\}$$

Graf simple

- Tenim un bucle (o **loop**) si tenim un node connectat amb si mateix, i.e. tenim una aresta que comença i acaba al mateix node.

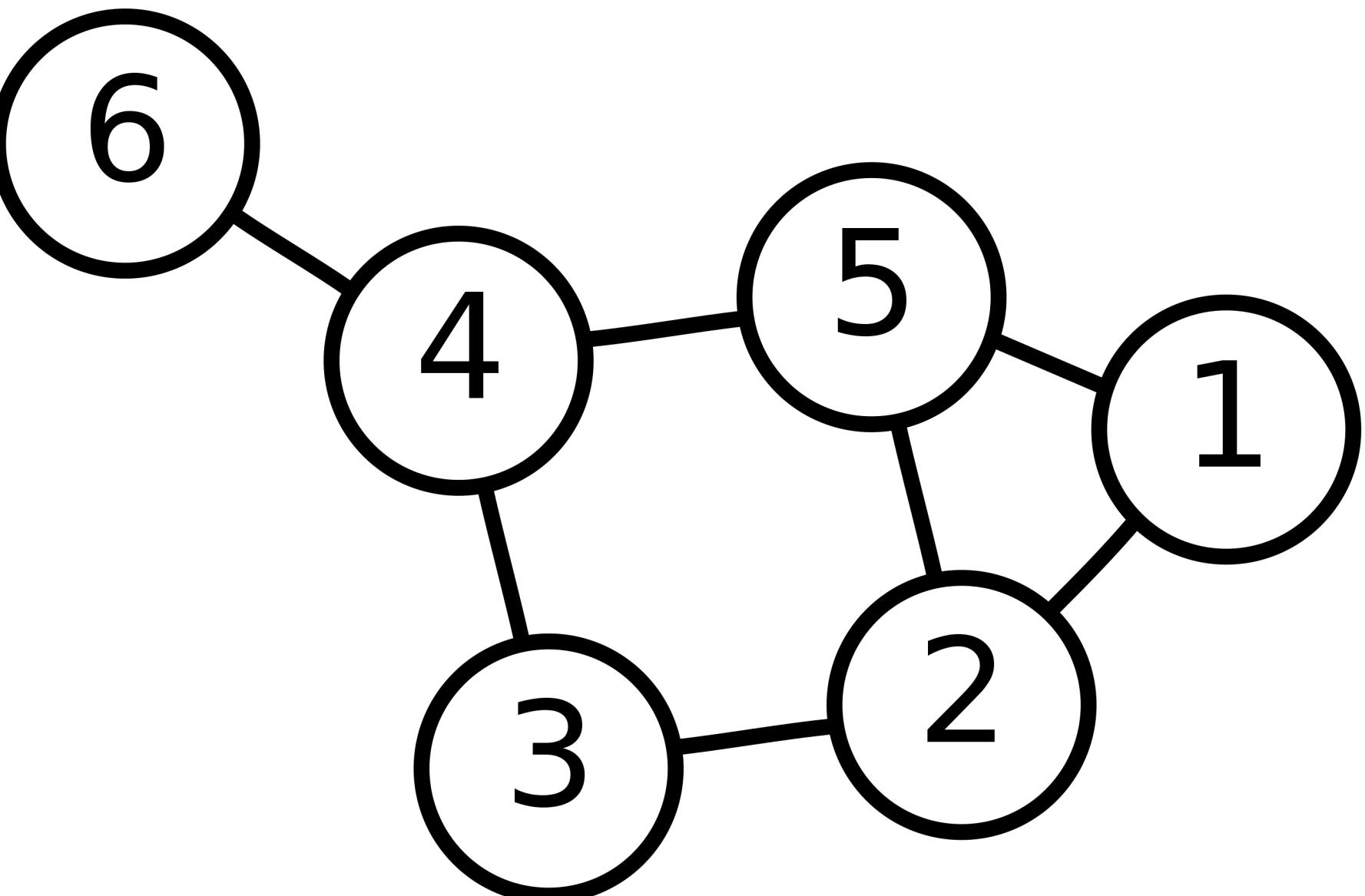


- Múltiples arestes entre dos nodes



- Diem que el graf es **simple** si es el graf és no dirigit, no té cap **bucle** i no més d'una aresta entre dos vèrtexs diferents qualsevol

- Es això un graf simple?

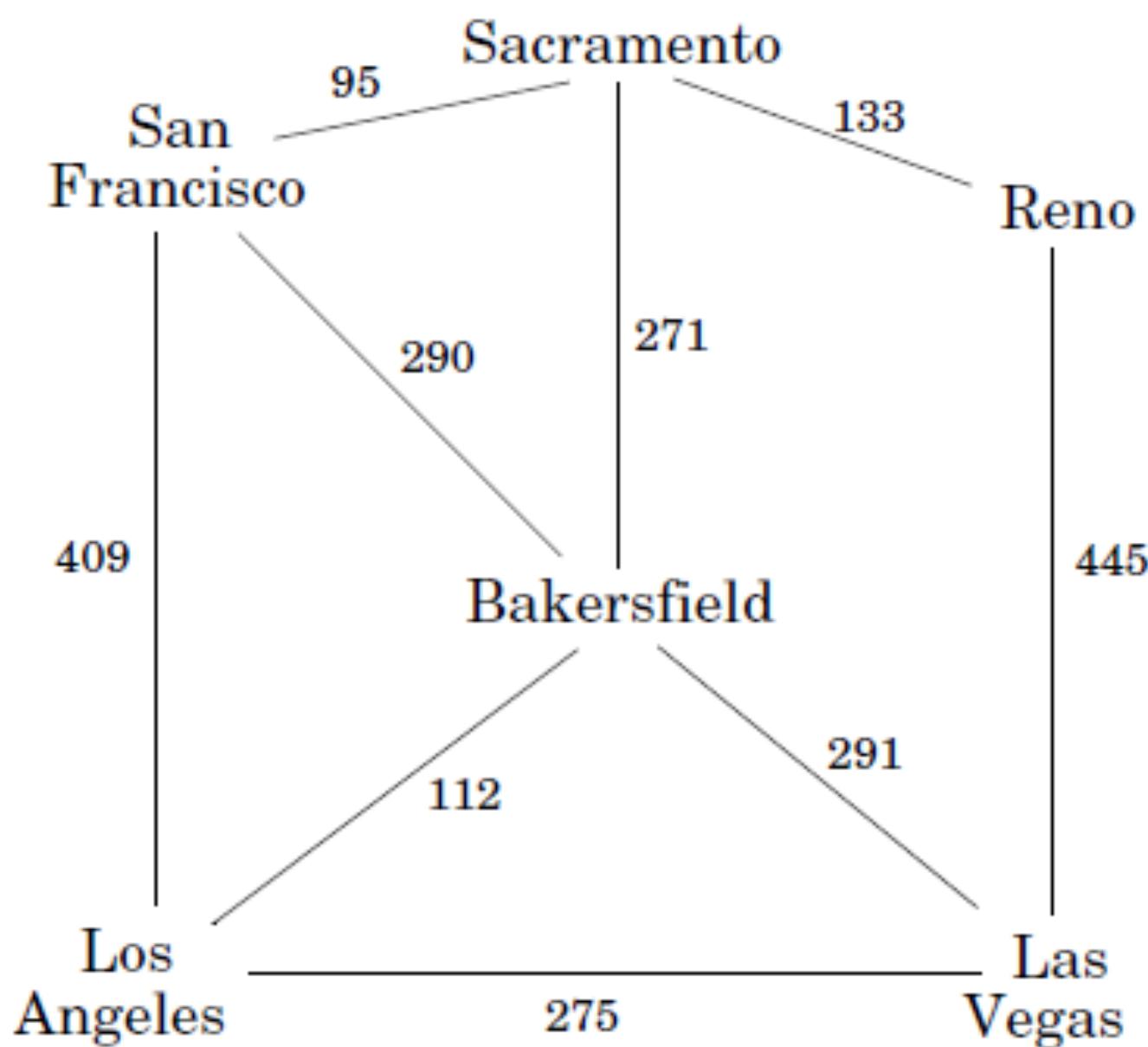


Grafs amb pesos

Algorismes sobre grafs

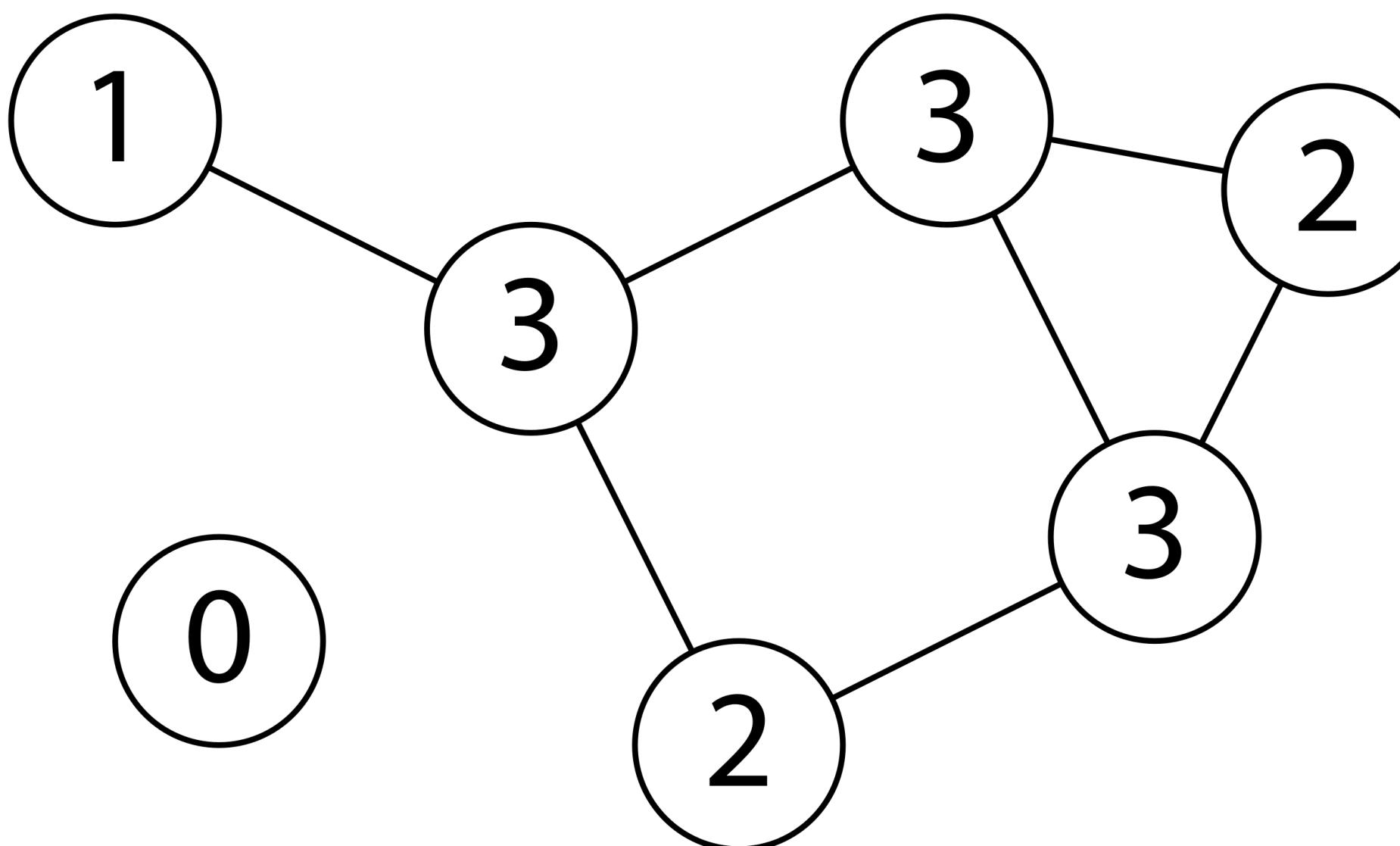
- **Pesos a les arestes**

Exemple amb distàncies



Propietats dels grafs

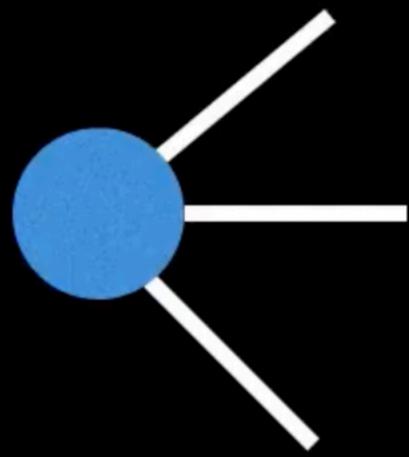
- El **grau** d'un vèrtex és el nombre d'arestes que hi incideixen



Un graf amb vèrtexs etiquetats segons el seu grau. El vèrtex aïllat s'etiqueta amb 0, ja que no és adjacent a cap altre vèrtex.

Que és el grau d'un node?

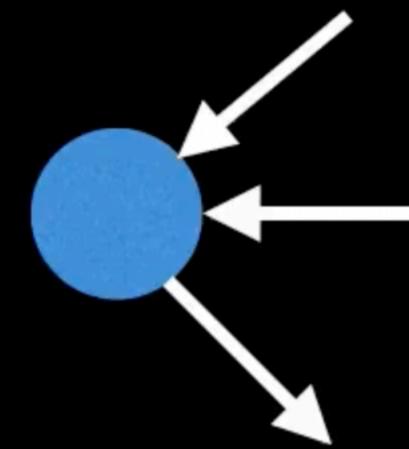
Graf no dirigit



Grau (degree) del node = 3

El grau d'un node és el nombre d'arestes incidents a ell

Graf dirigit



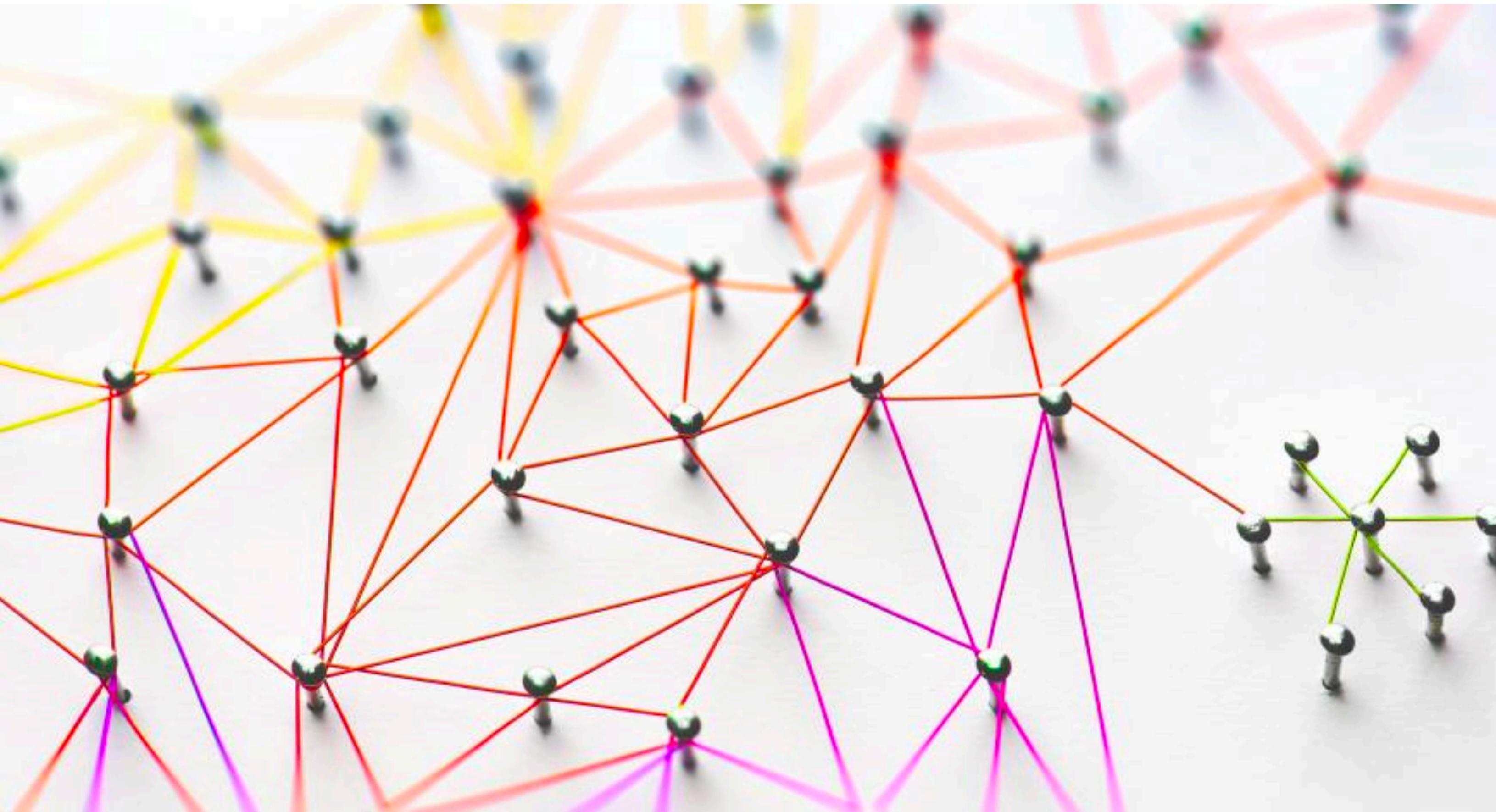
Grau d'entrada (In degree) = 1

Grau de sortida (Out degree) = 3

El grau d'entrada (indegree) es el nombre d'arestes d'entrada i el grau de sortida (outdegree) és el nombre d'arestes de sortida

Exemples

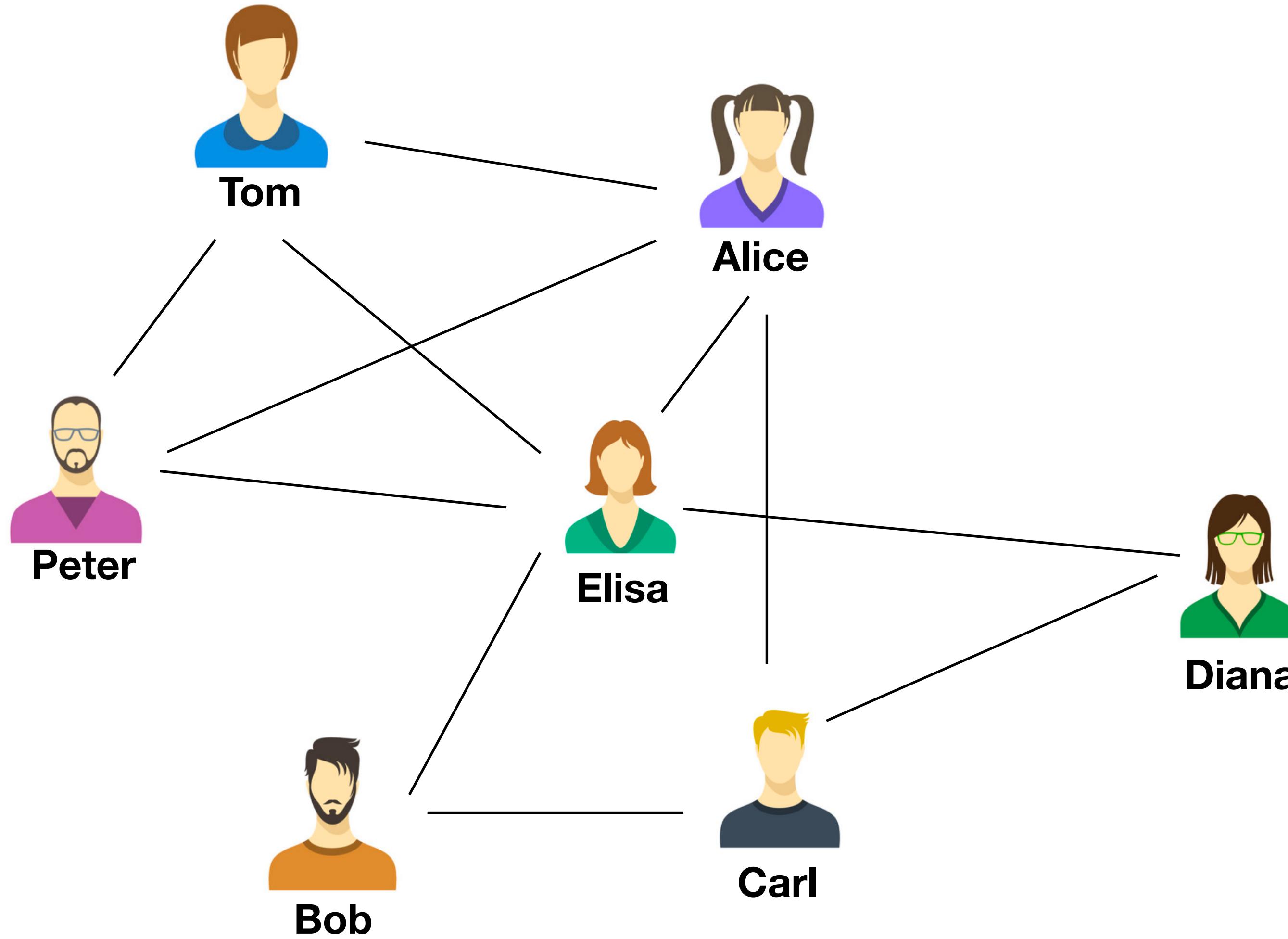
Principalment treballarem amb grafs



Xarxes socials

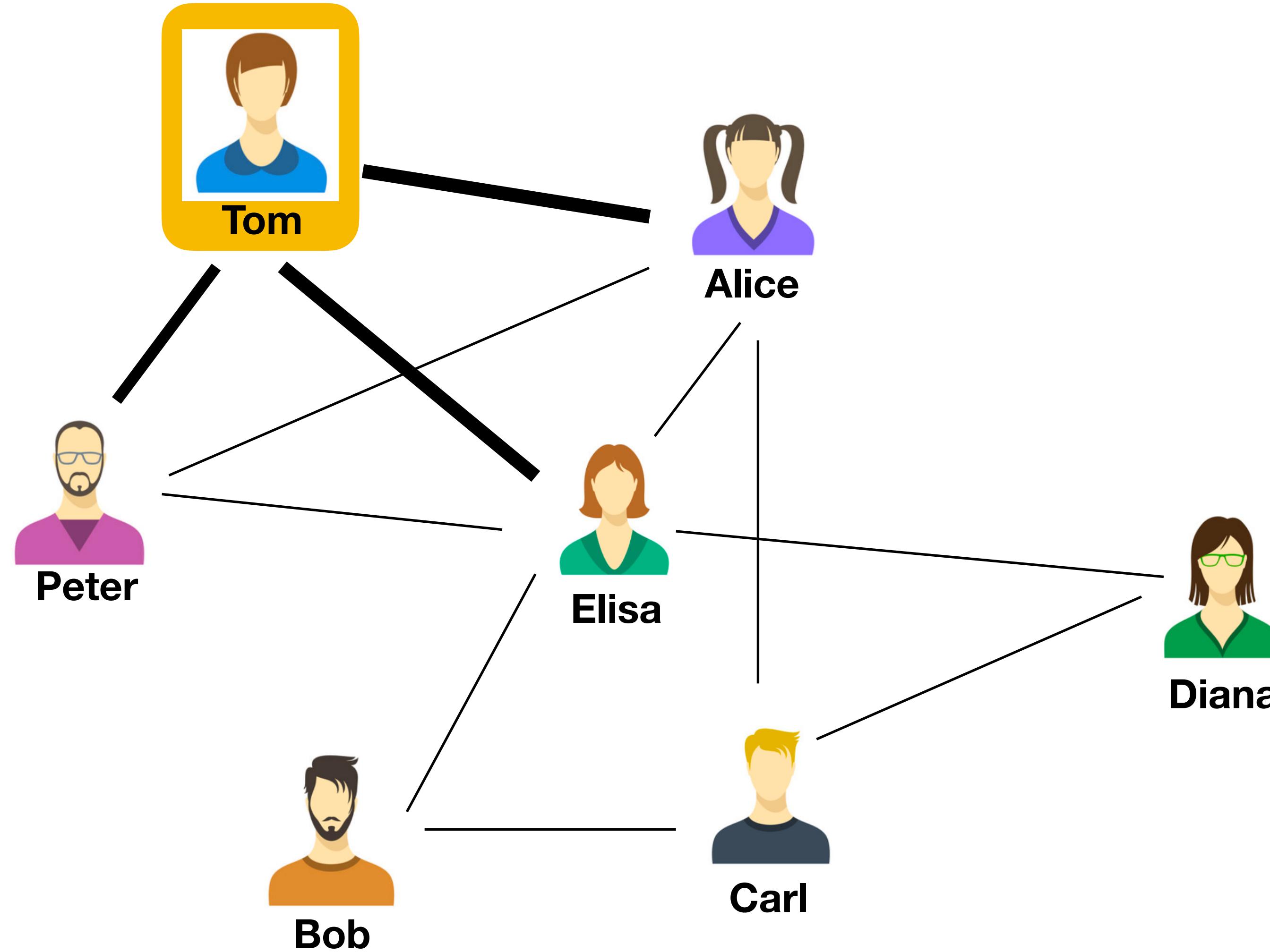


Social Network

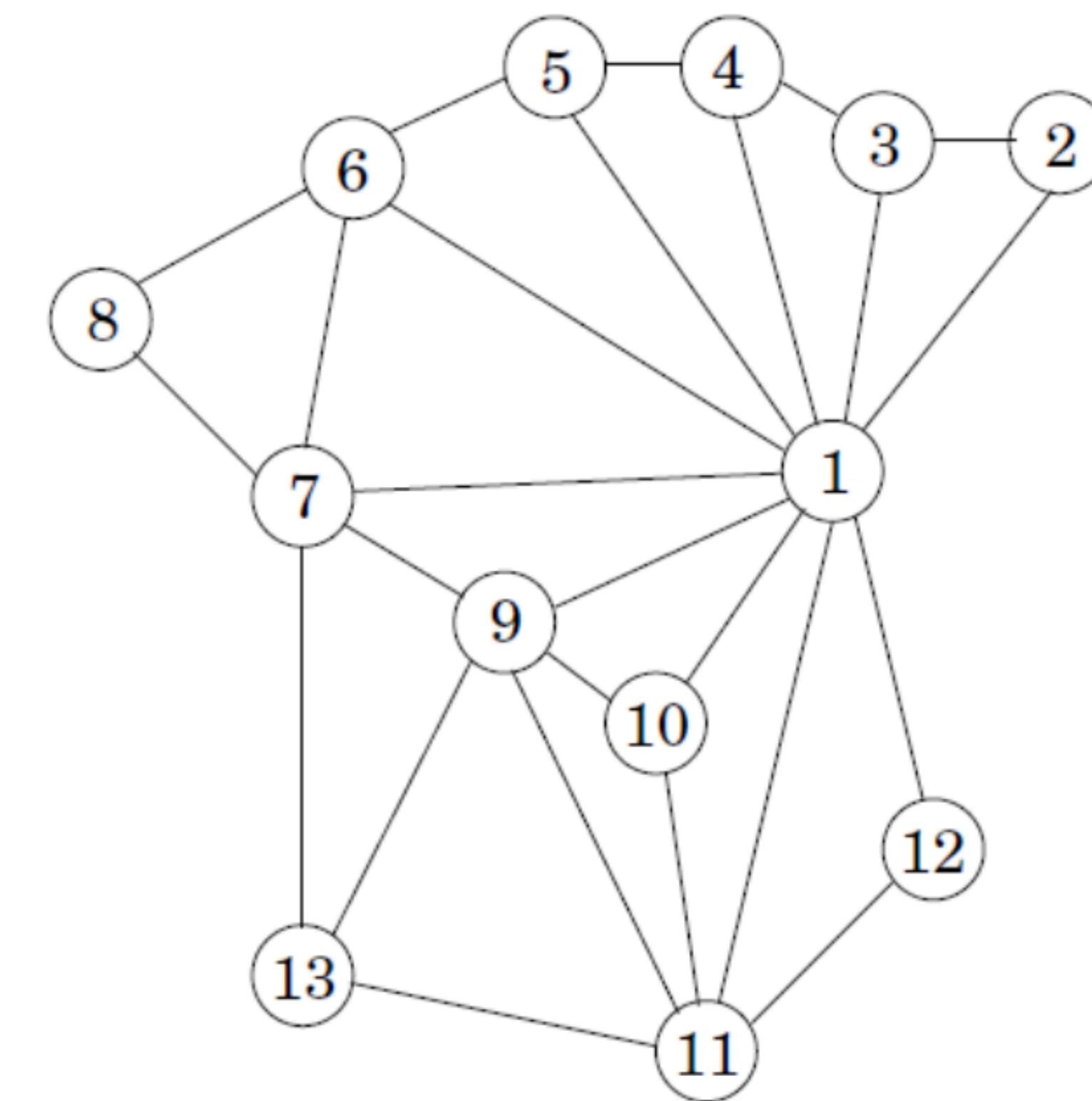


Social Network

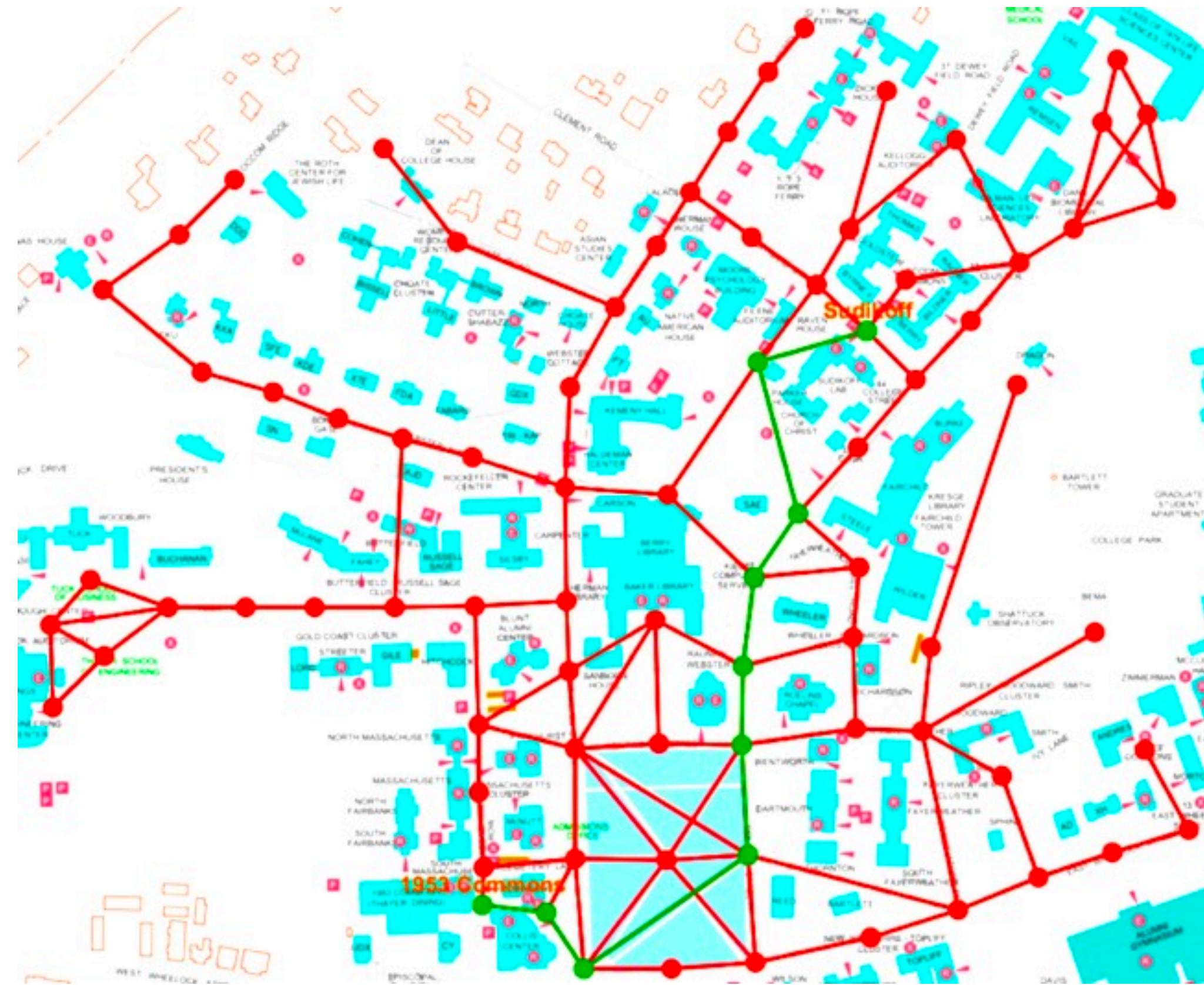
can you suggest some friends to Tom?



- Per a què serveix un graf?

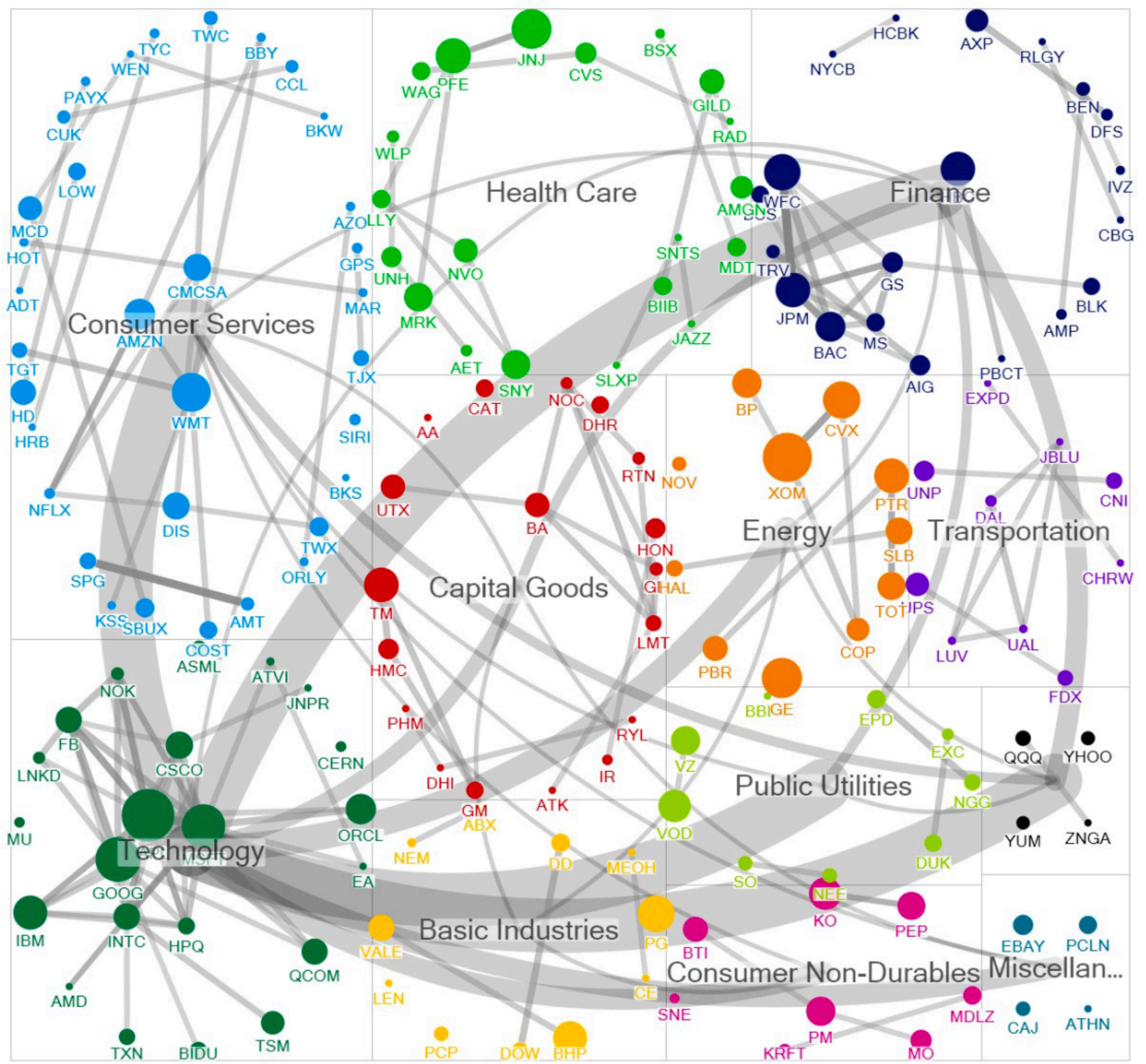


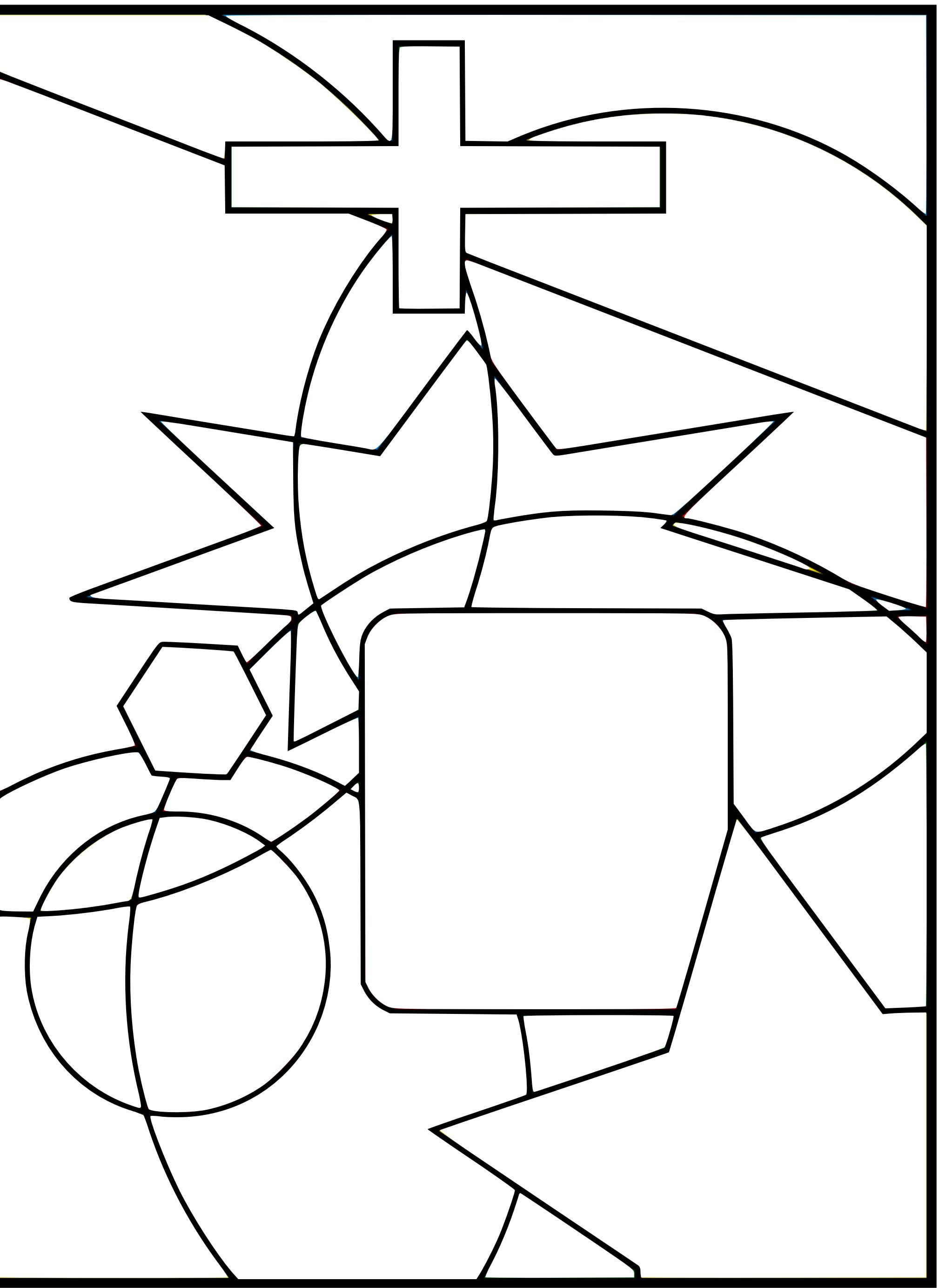
Mapes

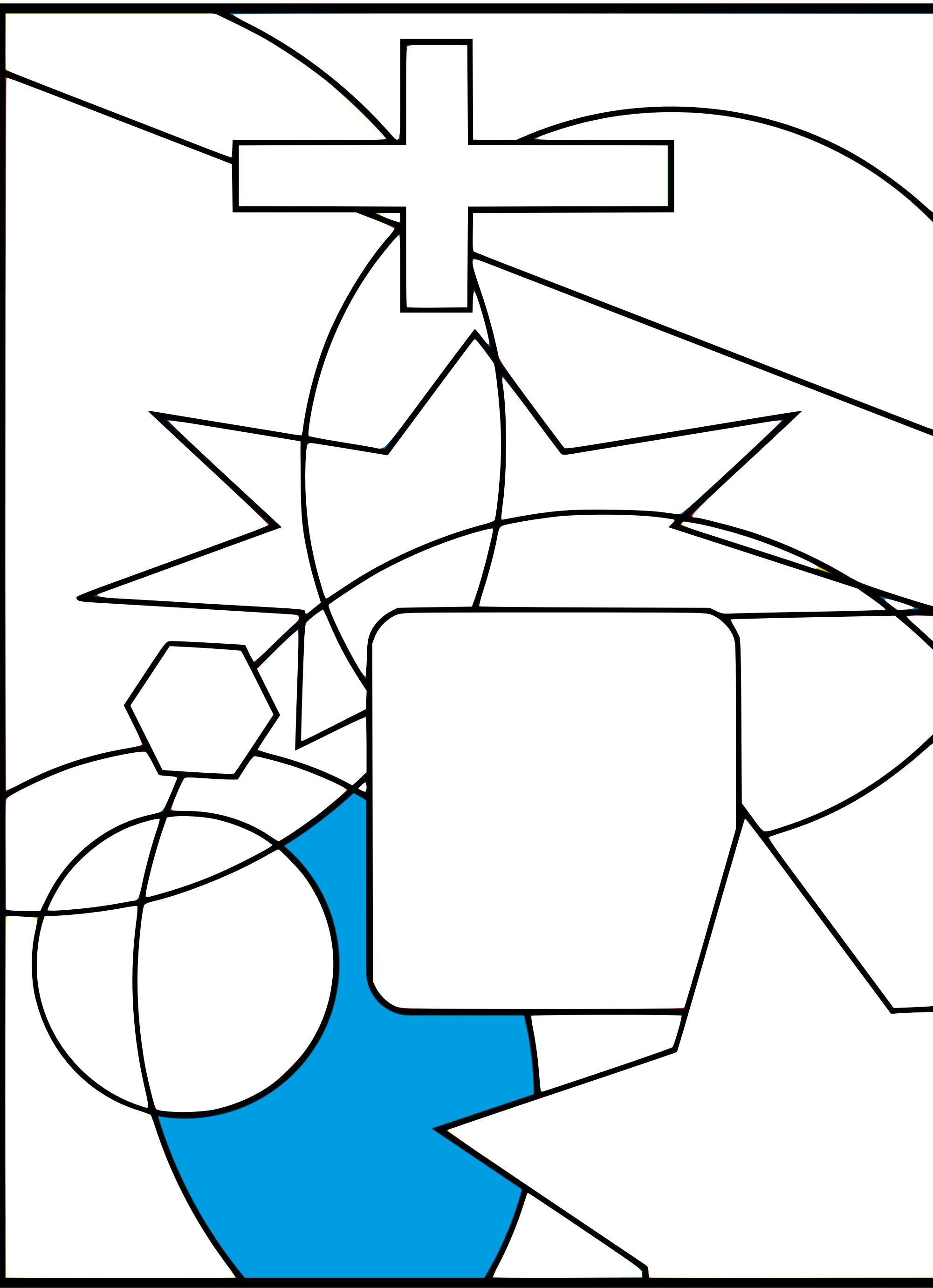


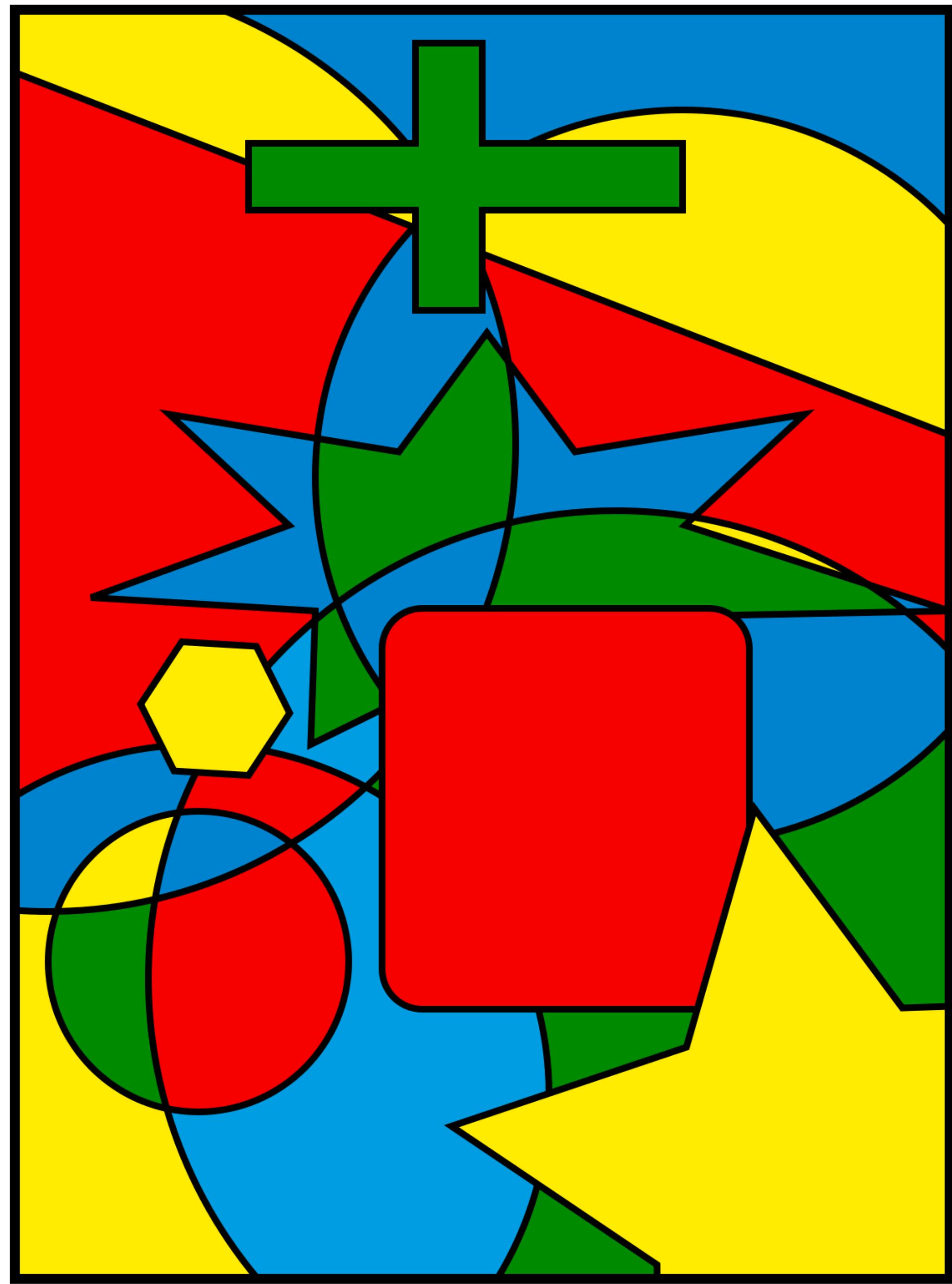
Com podem visitar tots els PDI's fent el menor número de kilòmetres?



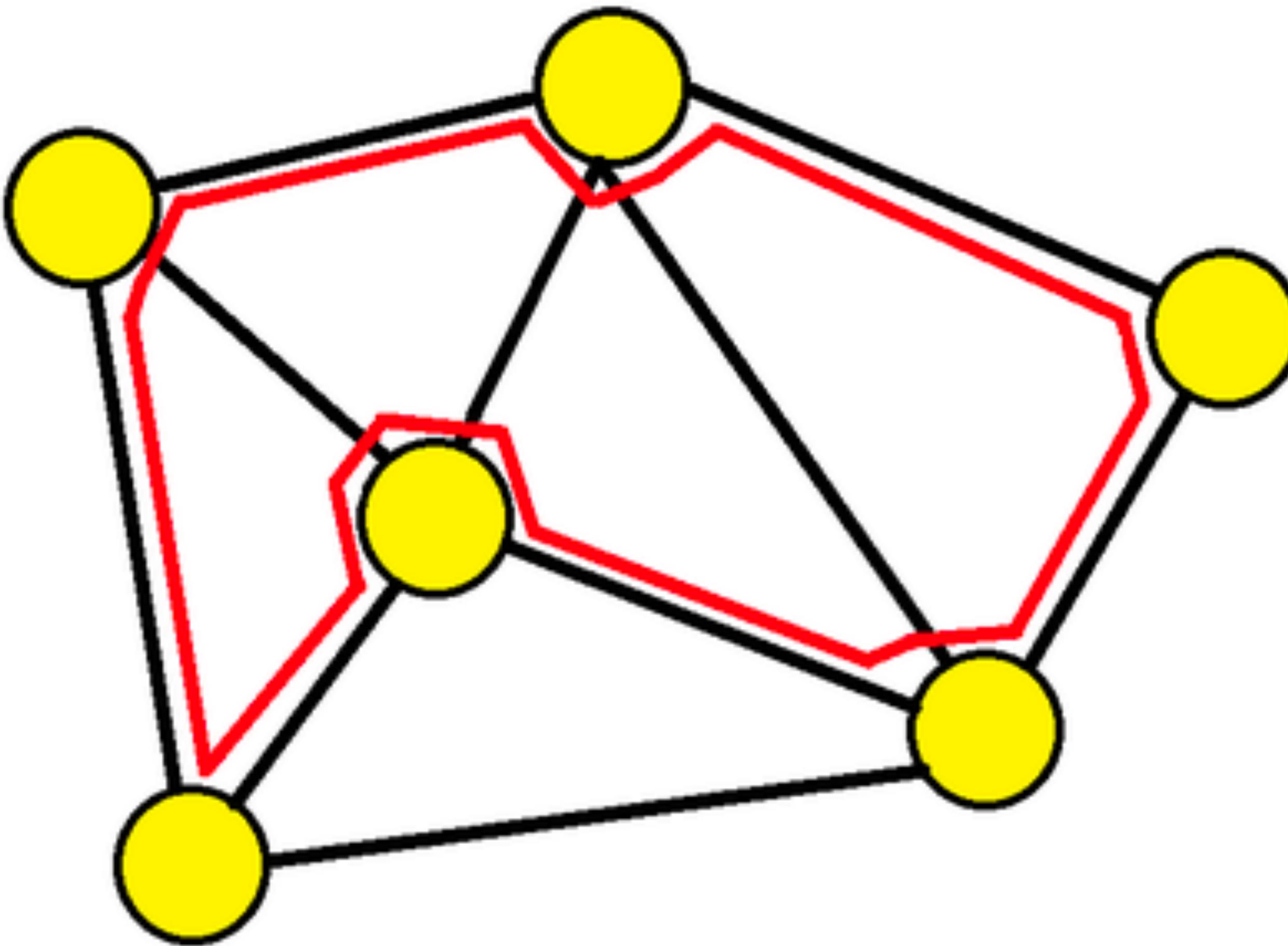


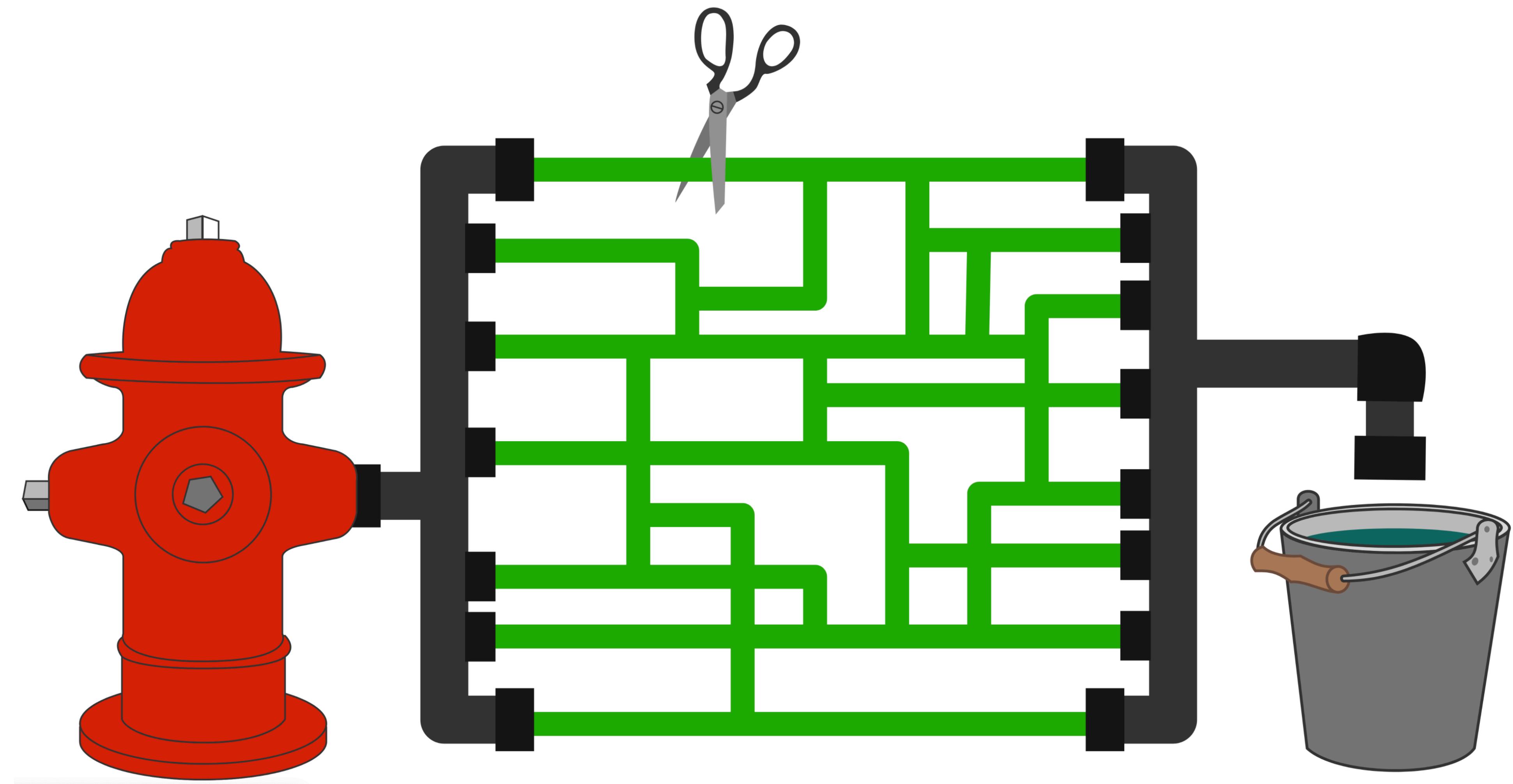






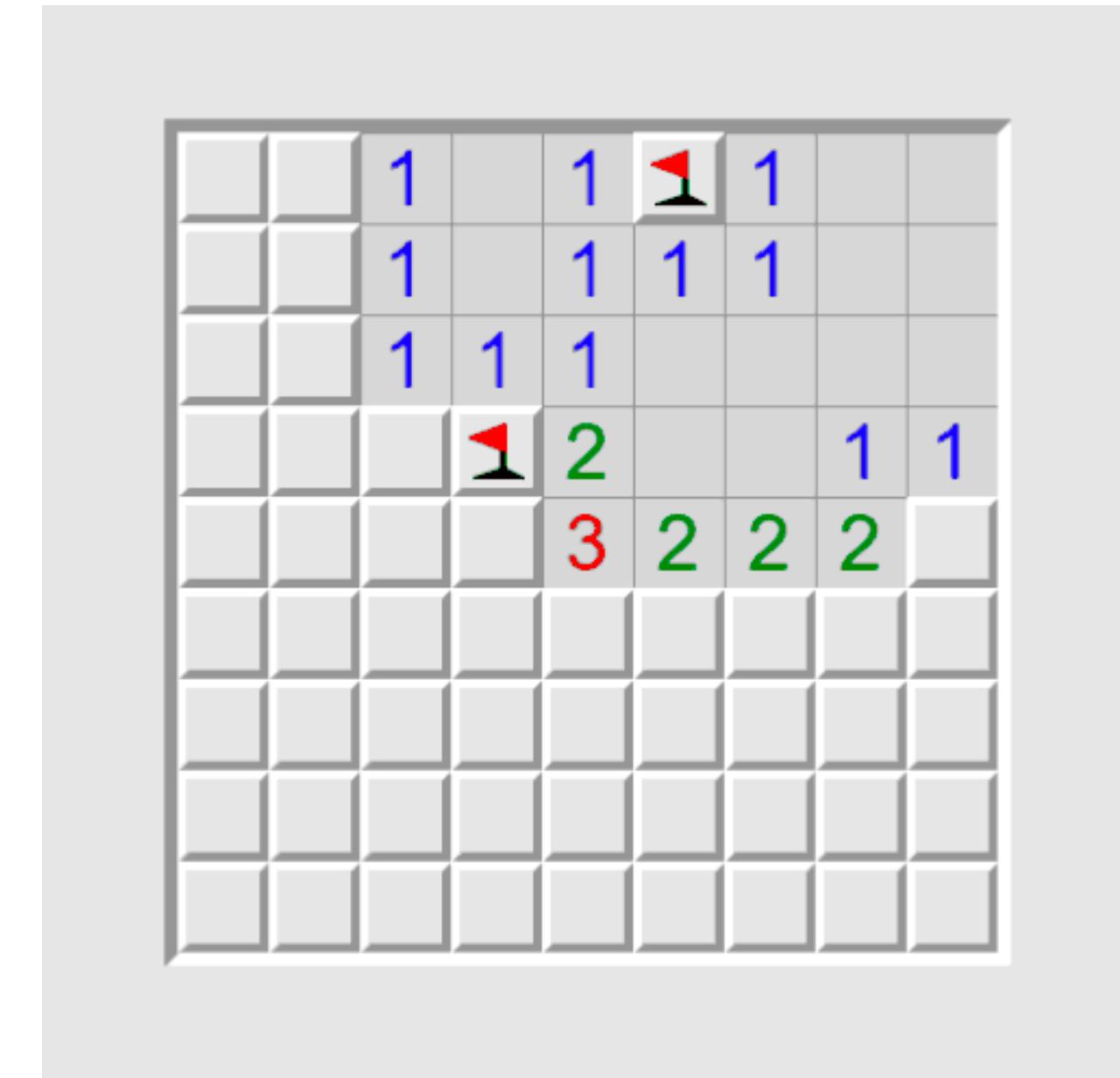
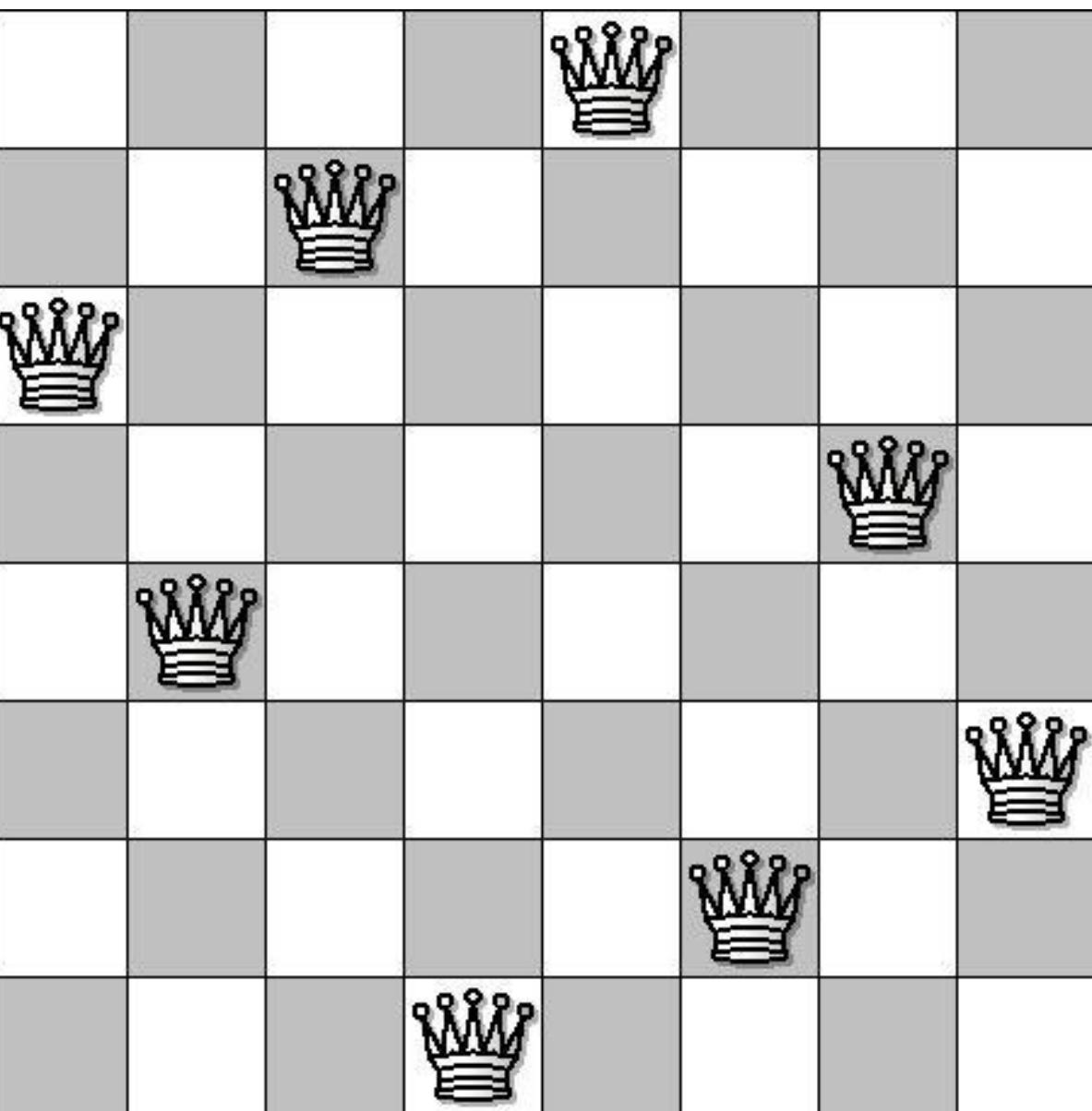
Hi ha un cicle que estrictament ens permet passar un cop per cada node?





JOCS

8								
	3	6						
7		9	2					
5			7					
	4	5	7					
	1			3				
1				6	8			
8	5			1				
9			4					



Aplicacions amb grafs

Graph	Vertex	Edge
Network Communication	telephone, computer	fiber optic cable
circuit	gate, register, processor	wire
financial	stock, currency	transactions
transportation	street intersection, airport	highway, airway route
game	board position	legal move
social relationship	person, actor	friendship, movie cast
molecule	atom	bond
neural network	neuron	synapse

Algunes preguntes interessants!

- **Camí (Path):** Hi ha cap camí entre els nodes s i t ?
- **Camí més curt:** Quin és el camí més curt entre els nodes s i t ?
- **Cicle:** Hi ha cap cicle en el graf?
- **Camí d'Euler:** Hi ha camí que em permeti visitar cada node passant exactament un cop per cada aresta?
- **Camí Hamiltonià:** Hi ha un camí en el graf que passa exactament un cop per cada vèrtex?
- **Connectivitat.** Hi ha cap manera de connectar tots els nodes
- **MST.** Quina és la millor manera de connectar tots els nodes d'un graf?
- **Bioconnectivitat.** Hi ha cap node que la seva eliminació disconnecti el graf.
- **Planaritat.** Podeu dibuixar el gràfic en el pla sense vores de creuament
- **Isomorfisme gràfic.** Les dues llistes adjacència representen el mateix gràfic?
- **Flux màxim:** Quin és el flux màxim que podem transportar d'un node s a un node t

Quins d'aquests problemes
són fàcils? difícil? intractable?

Planificació Organització

Algorísmica Avançada

Grafs

Què son els grafs?

Com es representen?

DFS /BFS/ Component Connexes

Com trobar camí més curts entre dos nodes (Dijkstra/BellmanFord)

Problema Flux Màxim

Circuits i camins eulerians

Algorítmes greedy

Un **algorisme greedy** és un algorisme que, per resoldre un problema d'optimització, fa una seqüència d'eleccions, prenent en cada pas un òptim local, amb l'esperança (no sempre complerta) d'arribar a un òptim global. L'algorisme voraç no torna mai enrere per re-avaluar les eleccions ja preses.

Kruskal / Prim

Problema de la Motxilla/Viatger de Comerç

Programació Dinàmica

Problemes d'optimització de tasques per tal d'arribar a una solució òptima

Floyd Wharshall

Problema de la Motxilla, Levenshtein

Ponts de Hanoi

...

Enumeratius

Backtracking, Ramificació i Poda

Sudoku, 8-Reines,

Problema de la Motxilla

...

...

Objectius del curs

- Anàlisi i disseny avançat d'algorismes, tant des d'un punt de vista teòric com aplicat

Què esperem dels estudiants?

- Els estudiants han de participar de forma activa durant les classes de teoria (1.5 hores a la setmana). Durant les hores presencials de pràctiques i problemes (1.5 i 1.5 hores cada setmana excepte les setmanes marcades com NO classe) hauran de resoldre de forma individual una sèrie de problemes. Les hores no presencials de l'assignatura (4 hores a la setmana) les han de dedicar a l'estudi de la teoria i a la preparació de les pràctiques.

Contactes:

Teoria Online

- [Santi Seguí](mailto:santi.segui@ub.edu) Email: santi.segui@ub.edu

Sessions Teoriopràctiques

- [Santi Seguí](mailto:santi.segui@ub.edu) Email: santi.segui@ub.edu
- [Pere Gilabert](mailto:pere.gilabert@ub.edu) Email: pere.gilabert@ub.edu

Laboratoris

- [Santi Seguí](mailto:santi.segui@ub.edu) Email: santi.segui@ub.edu
- [Pere Gilabert](mailto:pere.gilabert@ub.edu) Email: pere.gilabert@ub.edu
- [Enric Florit](mailto:xxxxxx@ub.edu) Email: xxxxxx@ub.edu

Degut a les restriccions de presencialitat actuals a causa del COVID-19:

- **Com seran les classes Teòriques? (1.5 hora la setmana)**

- S'introdueixen els conceptes teòrics de l'assignatura.

- **Com seran les classes problemes? (1.5 hores a la setmana)**

- Es realitzaràn amb grups reduïts. Es farà un repàs de la teoria i exercicis associats al les sessions corresponent
- S'ha de portar portàtil sempre que es pugui!
- L'alumne ha d'haver treballat el contingut corresponent a la sessió.

- **Com seran les classes de Laboratori? (1.5 hora a la setmana)**

- S'espera que l'alumne treballi amb les diferents pràctiques de forma autònoma i asíncrona.
- Les pràctiques **es realitzen de forma individual.**

ALGORÍSMICA AVANÇADACodi: **364300****Pla docent**

Tipus	Impartició	Crèdits	Curs/Semestre	Unitat Acadèmica
Obligatòria del grau	Quadrienal	6	2 / 1	Departament de Matemàtiques i Informàtica

Programació de l'oferta docent del Primer semestre

Activitat						
Grup	Dies	Horari	Professorat	Aula	Idioma	
Teoria [Presencial]						
M1	dl. dt. dc. dj. dv.	1r sem. 11.00-12.30	Segui Mesquida, Santiago	Aula B5	Català	
Pràctiques de problemes [Presencial]						
ab0	dl. dt. dc. dj. dv.	1r sem. 08.00-09.30	Gilabert Roca, Pere	Aula B1	Català	
cf0	dl. dt. dc. dj. dv.	1r sem. 08.00-09.30	Segui Mesquida, Santiago	Aula B1	Català	
Pràctiques de laboratori d'ordinadors [Presencial]						
a00	dl. dt. dc. dj. dv.	1r sem. 09.30-11.00	Gilabert Roca, Pere	Aula IA	Català	
b00	dl. dt. dc. dj. dv.	1r sem. 09.30-11.00		Aula IB	Català	
c00	dl. dt. dc. dj. dv.	1r sem. 09.30-11.00	Gilabert Roca, Pere	Aula IA	Català	
f00	dl. dt. dc. dj. dv.	1r sem. 09.30-11.00	Segui Mesquida, Santiago	Aula IB	Català	
Exàmens : 1r parcial [Presencial]						
G1	9 de novembre de 2021.	18.00-21.00	Segui Mesquida, Santiago Gilabert Roca, Pere		-	
Exàmens : Final [Presencial]						
G1	13 de gener de 2022.	15.00-18.00	Segui Mesquida, Santiago Gilabert Roca, Pere		-	
Exàmens : Revaluació [Presencial]						
G1	31 de gener de 2022.	15.00-18.00	Segui Mesquida, Santiago Gilabert Roca, Pere		-	

Llenguatge de programació?

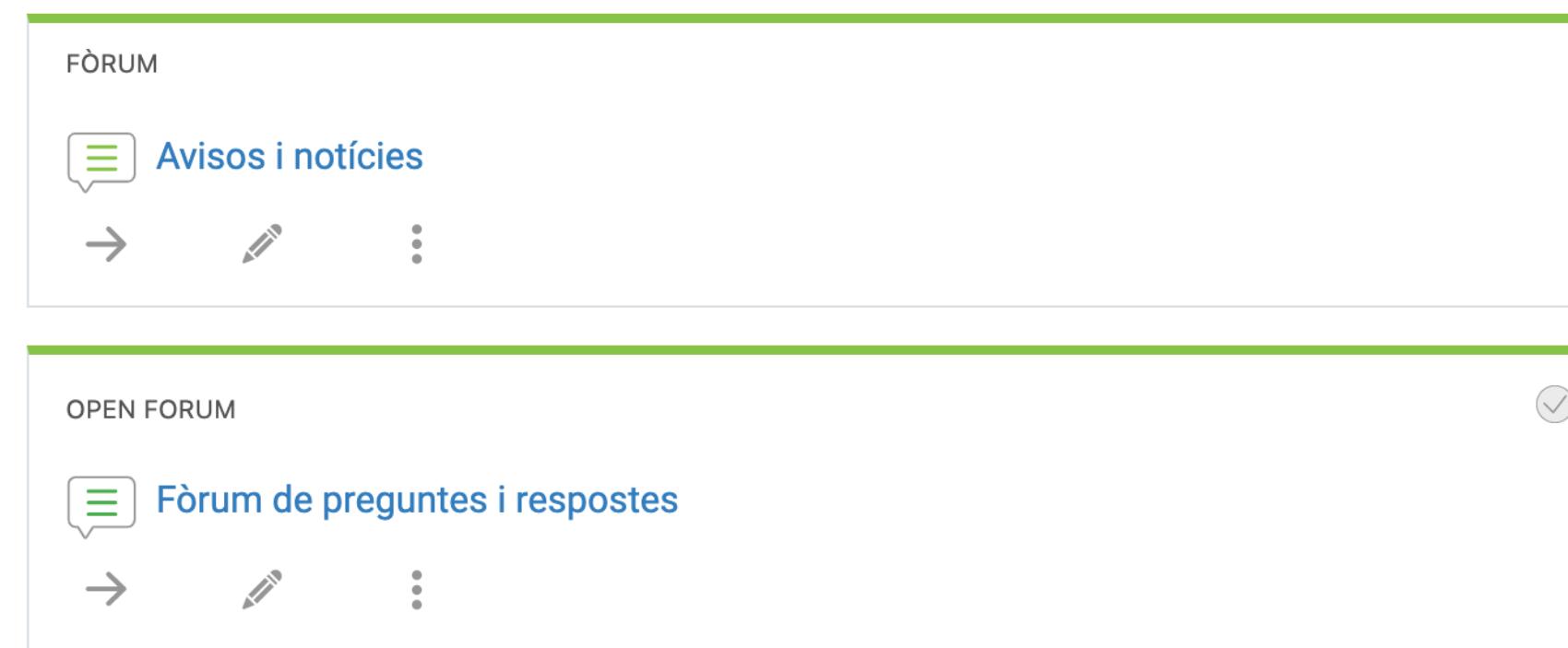
- Python 3.6

Com s'organitza l'assignatura?

- L'assignatura s'imparteix en classes teòriques, problemes i pràctiques. L'assignatura es coordinarà mitjançant una eina electrònica (basada en Moodle 2) que s'anomena Campus Virtual i que és accessible a través de la web. A través d'aquest entorn tindreu: anuncis, apunts, notes, fòrum, calendari, enllaços a la bibliografia, etc.
 - <https://campusvirtual2.ub.edu/>

IMPORTANT: TOTS els dubtes, qüestions, problemes, etc.. de caire no personal s'han de realitzar mitjançant el **Fòrum del Campus Virtual**.
No es contestarà cap correu electrònic que no sigui una qüestió personal.

Fòrum



IMPORTANT: TOTS els dubtes, qüestions, problemes, etc.. de caire no personal s'han de realitzar mitjançant el **Fòrum del Campus Virtual**.

No es contestarà cap correu electrònic que no sigui una qüestió personal.

Què puc preguntar?

- Qualsevol cosa referent a la assignatura! Dubtes, problemes amb còdis, ...

Puc compartir informació?

- Podeu compartir qualsevol cosa que vulgueu. Entre elles documents, vídeos, solicions a problemes, exàmens d'anys passat solucionats, etc... És a dir, qualsevol cosa que creieu que pot ajudar a la resta d'alumnes.

Segur que puc compartir tot el que vulguí?

- **NO!!** Està **totalment prohibit** compartir les vostres pràctiques/problemes avalables, així com qualsevol solució a les pràctiques/problemes avalables de l'assignatura.

Puc contestar?

- Si!!! Una activa participació al fòrum es tindrà en compte de forma positiva (un 8.8 de l'assignatura és pot convertir amb un 9).

Com s'avaluarà l'assignatura?

- L'assignatura seguirà un esquema d'avaluació continuada, amb dos elements principals: exàmens presencials i lliurament de pràctiques

Proves presencials:

- Examen Parcial Teoria (*EPT*)
- Examen Final Teoria (*EFT*)
 - Als exàmens de **teoria no es poden portar apunts.**
- Examen Parcial Pràctiques (*EPP*)
- Examen Final Pràctiques (*EFP*)
 - Als exàmens de **pràctiques es poden portar apunts i material digital offline** però no es podrà fer servir Internet excepte per pujar la prova al final de la sessió.
 - NOTA: Les pràctiques i proves de pràctiques es faran amb NOTEBOOK

Lliurament de pràctiques:

- Lliurament de pràctiques: Cada un dels lliuraments de pràctiques serà avaluat pel professor amb una nota que pot anar de 0 a 10.
- Si l'estudiant no lliura les pràctiques dins del període assenyalat, obtindrà un 0.
- La nota final (*NLP*) de la part de pràctiques és la mitjana de tots els lliuraments.

- Nota Teoria (**NT**): Mitjana dels exàmens de teoria (*NPT* i *NFT*)
 - Nota Proves Pràctiques (**NPP**) : Mitjana dels exàmens de pràctiques (*NPP* i *NFP*)
 - Nota Lliuraments Pràctiques (**NLP**): Mitjana de tots els lliuraments
-
- Per poder estimar la **Nota final** de l'assignatura, l'estudiant ha de complir la condició següent:
 - **Nota final** =
 - $\frac{1}{3}$ Nota Teoria (**NT**) + $\frac{1}{3}$ Nota Proves Pràctiques (**NPP**) + $\frac{1}{3}$ Nota Lliuraments Pràctiques (**NLP**)
-
- Per aprovar l'assignatura s'ha de complir que
 - **Nota final** ≥ 5 ;
 - Tots els exàmens de teoria i pràctiques (*NPT*, *NFT*, *NPP* i *NFP*) han de tenir 4 com a nota mínima.
 - NLP ha ser igual o superior a 4.
 - En cas que **Nota final** ≥ 5 però alguna de les proves tingui una nota inferior a 4, aquesta prova es podrà recuperar individualment el dia de l'examen de Revaluació.
 - En cas que $3.5 \leq \text{Nota final} < 5$ es podrà recuperar l'assignatura amb un **examen únic** el dia de la Revaluació.

Revaluació

- L'alumne té dret a la revaluació sempre que la **Nota Final** obtinguda sigui com a mínim un 3.5.
- La revaluació consta d'un examen de teoria (50%) i un examen pràctic (50%)

Avaluació Única

- L'estudiant que es vulgui acollir a l'avaluació única ho ha de sol·licitar a la Secretaria de la Facultat dins dels terminis establerts cada curs acadèmic.
- L'avaluació única consisteix en un examen final únic que consta d'una part teòrica (ET) i una prova pràctica (PP) davant de l'ordinador.
- En aquest cas, la nota s'avalua de la manera següent:
 - **Nota_Final** = $0,5 * ET + 0,5 * PP$

Tutories

- Els professors de pràctiques us informaran dels seus horaris de tutories.
- Es important enviar un correu al professor abans per concertar l'horari de la consulta.
- La consulta seran **principalment de forma online**

	Dimarts		GRUP DIMARTS		GRUP DIJOUS		
Setmana	Teòria	Problemes	Pràctiques	Problemes	Pràctiques		
13-Set	Introducció (Presentació, Notació, Representació)				Introducció	Prac. Introductoria	
20-Set	Grafs I (DFS, BFS, Components Connexes, Dijkstra)		Introducció	Prac. Introductoria	Grafs I	Prac Grafs	
27-Set	Grafs II (Bellman Ford, Ordre topològic)		Grafs I	Prac Grafs			
4-Oct	Grafs III (Ford Fulkensen, Min-Cut Max-Flow)		Grafs II	Prac Grafs	Grafs II	Prac Grafs	
11-Oct	FESTIU		FESTIU	FESTIU	Grafs III		
18-Oct	Greedy I Minimum Spanning Tree, Kruskal, Unin-Find		Grafs III			Greedy I	Prac Greedy
25-Oct	Greedy II Prim, Graph Coloring, Hoffman Code		Greedy I	Prac Greedy	Greedy II	Prac Greedy	
1-Nov	Programació Dinàmica I Fibonacci, Floyd Warshall, Motchila, Viatjer riu		Greedy II	Prac Greedy			
8-Nov	Setmana Examens				Programació Dinàmica I	Prac PD	
15-Nov	Programació Dinàmica II LIS, levenshtein, Viatjant Comerç		Programació Dinàmica I	Prac PD	Programació Dinàmica II	Prac PD	
22-Nov	Enumeratius I: Backtracking Motxilla, Sudoku, Pintar Mapa		Programació Dinàmica II	Prac PD	Enumeratius I	Prac Enumeratius	
29-Nov	Enumeratius II: Branch & Bound Motxilla, Tasques, Puzzle		Enumeratius I	Prac Enumeratius			
6-Dec	FESTIU		FESTIU	FESTIU	Enumeratius II	Prac Enumeratius	
13-Dec	Enumeratius III: Branch & Bound Puzzle (Matriu Reduïda)		Enumeratius II	Prac Enumeratius			
20-Dec	Sessió Dubtes/Repàs				Prac Enumeratius		

Bibliografia

- G. Heineman, G. Pollice, S. Selkow, Algorithms in a Nutshell, O'Reilly Media, Inc. ©2008 ISBN:059651624X 9780596516246
- Mari Wahl, Python and Algorithms, University of New York at Stony Brook, http://www.astro.sunysb.edu/steinkirch/reviews/algorithms_in_python.pdf, 2013
- Robert Sedgewick, Algorithms in C, Third edition, Addison Wesley, 1998.
- Cormen, T.T. [et al]. Introduction to algorithms. Cambridge, Mass: MIT Press; New York; McGraw-Hill, 2007.
- S. Dasgupta, C.H. Papadimitriou and U.V. Vazirani. Algorithms; McGraw-Hill, 2006.
- Ian Parberry. Problems on algorithms. Englewood Cliffs; Prentice Hall, 1995. ISBN: 0-13-433558-9.
- N. Wirth. Algoritmos+Estructuras de Datos=Programas; Ediciones del Castillo, 1980. ISBN: 84-219- 0172-9

queixes, problemes, suggeriments,

Utilitzeu el següent formulari **anònim** per qualsevol d'aquestes coses

Bustia de Suggeriments/Queixes/Problemes

* Required

Exposa la teva queixa/suggeriment/problema *

Your answer

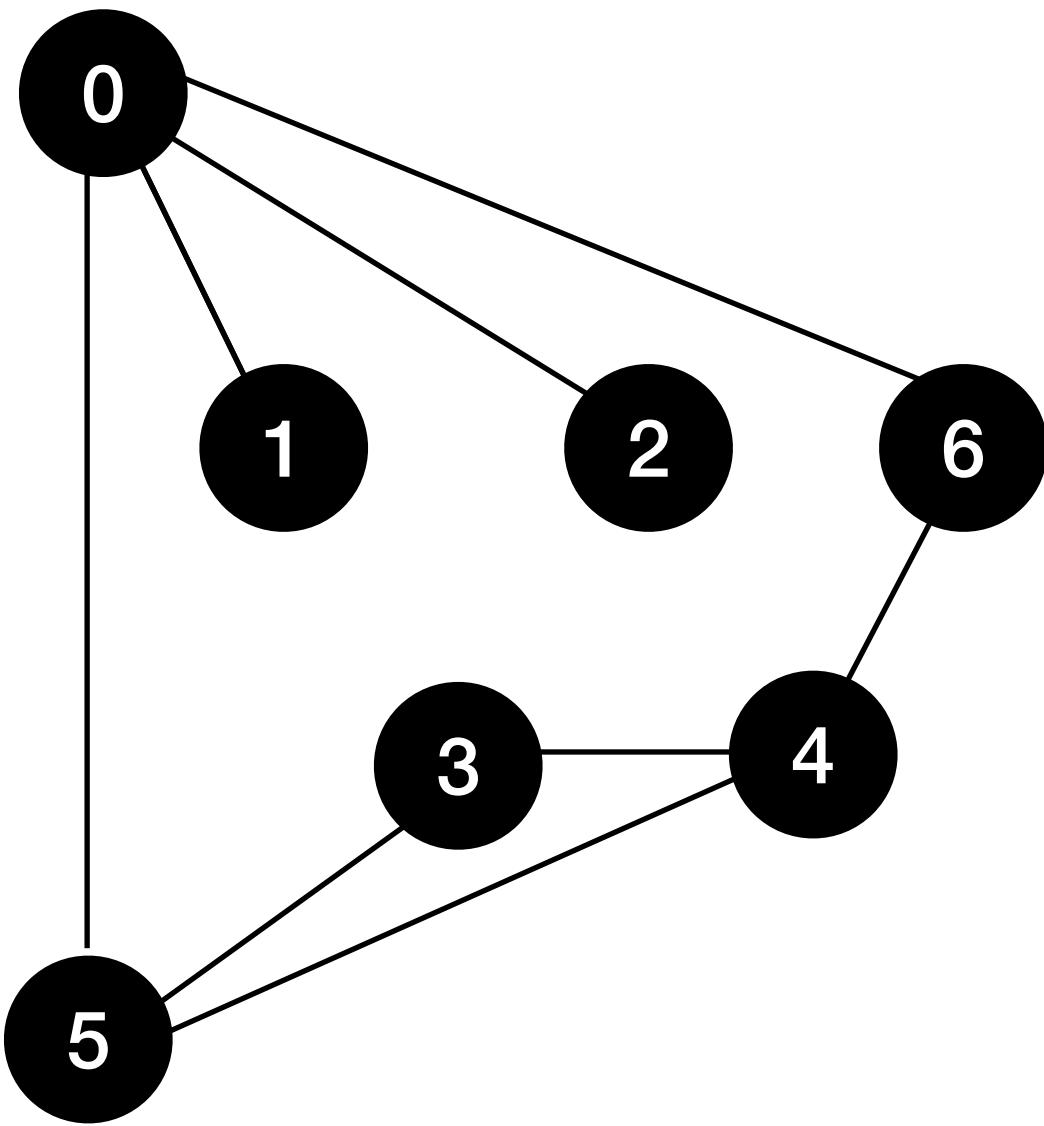
Nivell de satisfació del curs

	1	2	3	4	5	
Suspès	<input type="radio"/>	Excel·lent				

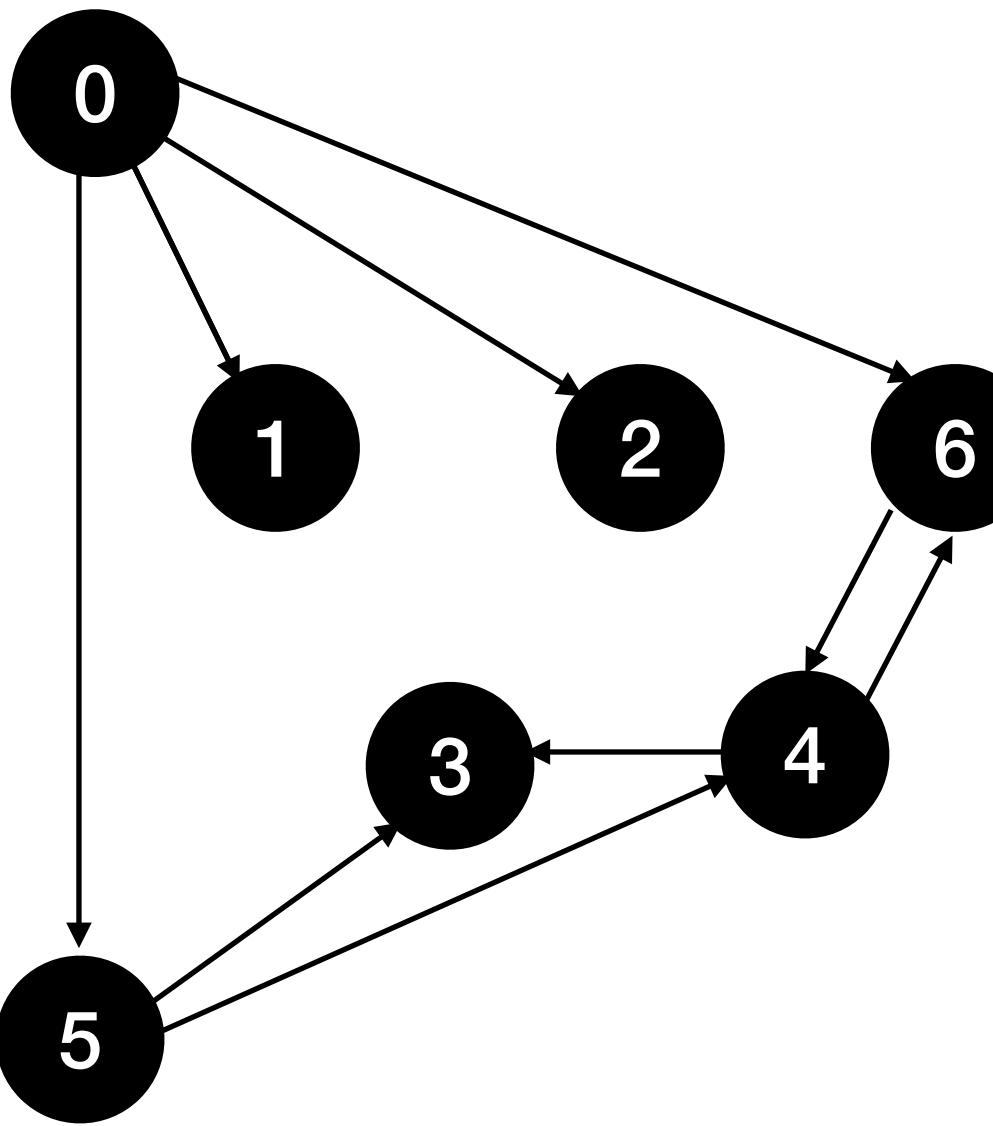
Submit

Representació dels grafs

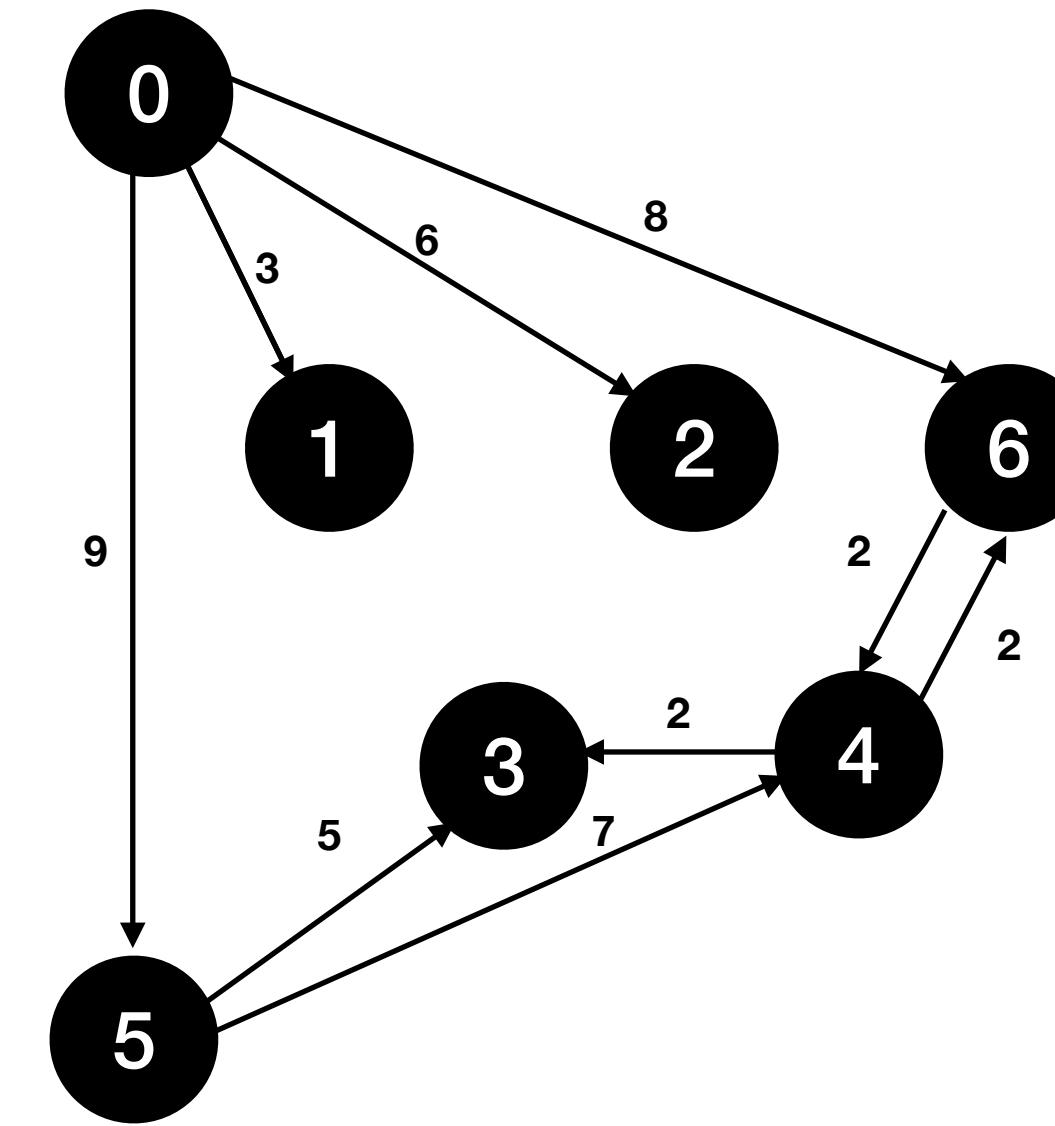
Representació dels grafs



Graf no dirigit



Graf dirigit



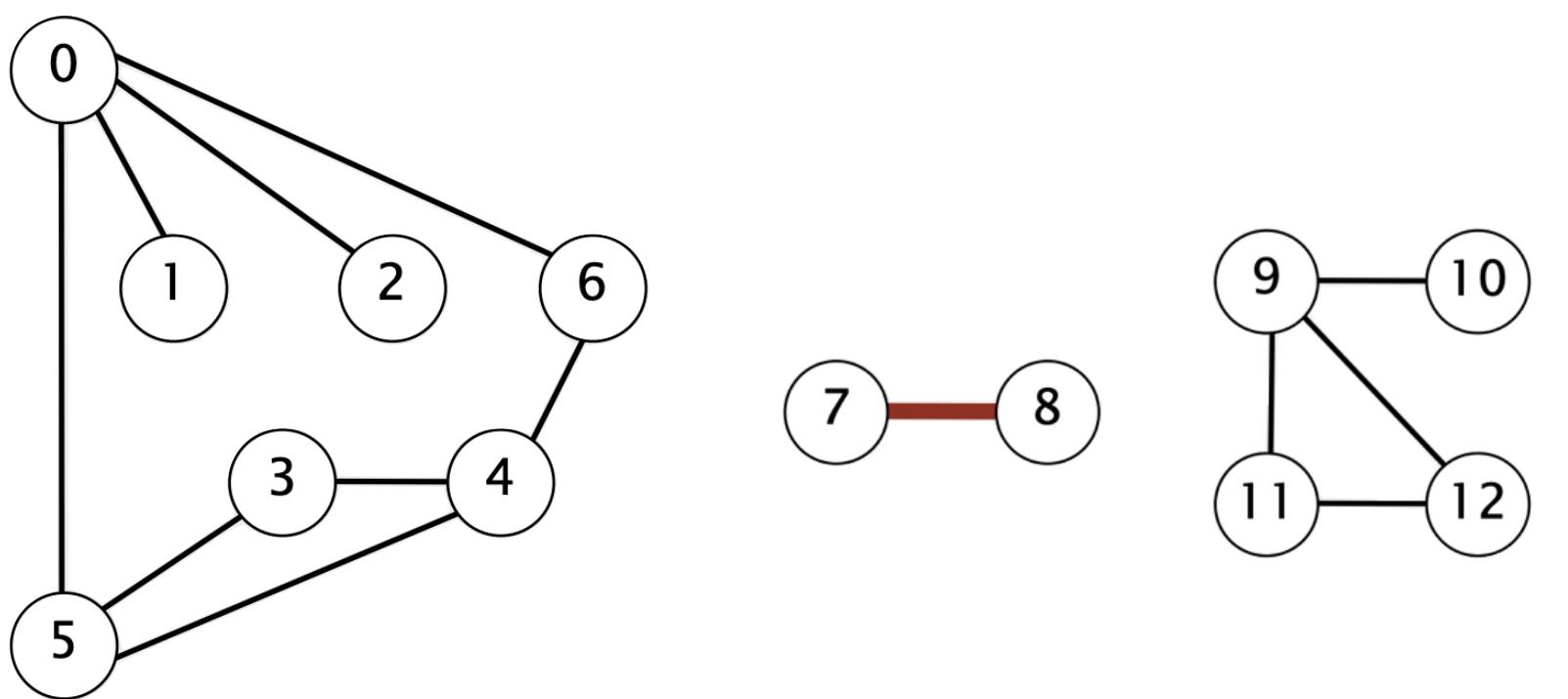
Graf dirigit amb pesos

Com representem aquests grafs $G(V, E)$?

Representació dels grafs

- Com representem un graf?

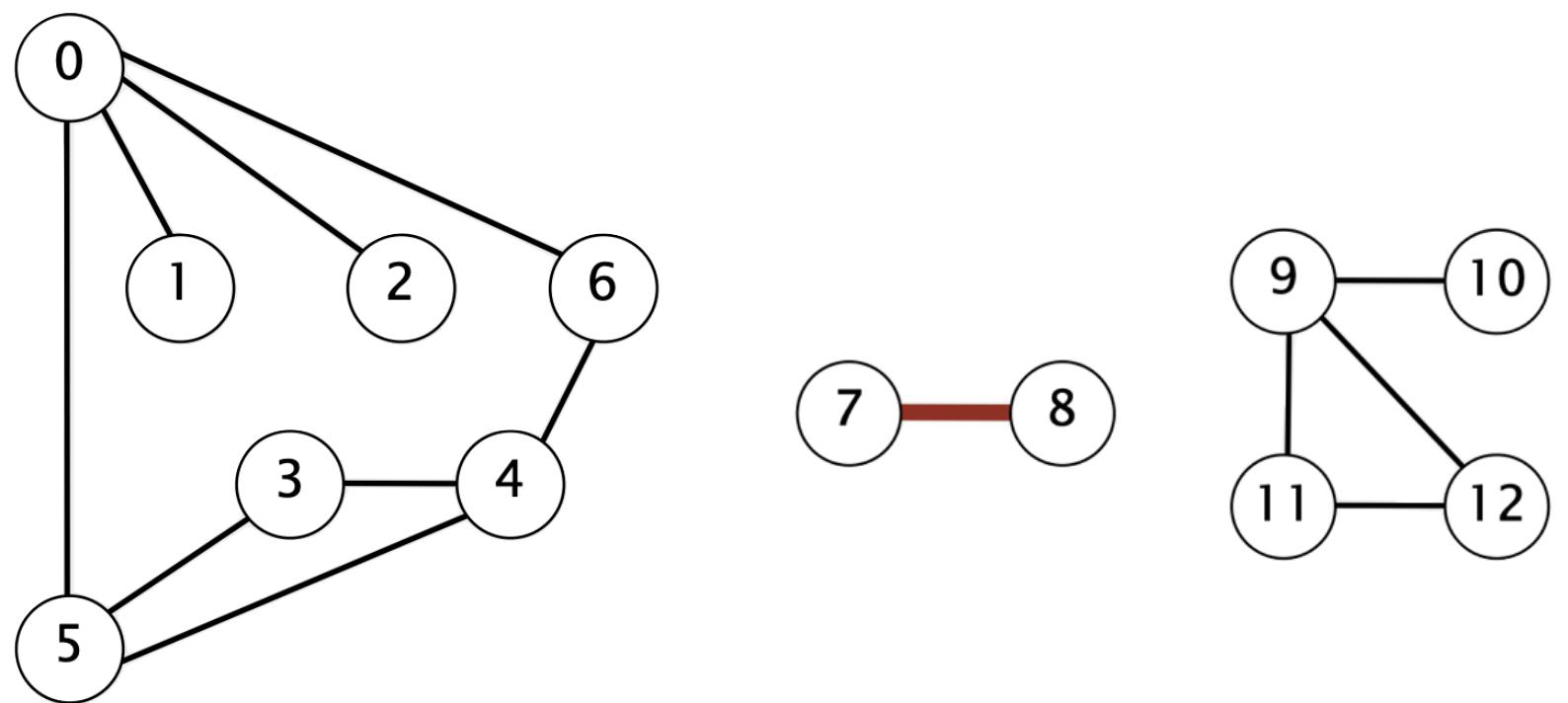
- **Linked list o array**
- Matriu d'adjacència
- Llista d'adjacència



0	1
0	2
0	5
0	6
3	4
3	5
4	5
4	6
7	8
9	10
9	11
9	12
11	12

Representació dels grafs

- Com representem un graf?
 - Linked list o array
 - **Matriu d'adjacència**
 - Llista d'adjacència



La mateixa
dada

	0	1	2	3	4	5	6	7	8	9	10	11	12
0	0	1	1	0	0	1	1	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0	0	0	0	0
2	1	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	1	1	0	0	0	0	0	0	0
4	0	0	0	1	0	1	1	0	0	0	0	0	0
5	1	0	0	1	1	0	0	0	0	0	0	0	0
6	1	0	0	0	1	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	1	0	0	0	0
8	0	0	0	0	0	0	0	1	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	1	1	1
10	0	0	0	0	0	0	0	0	0	1	0	0	0
11	0	0	0	0	0	0	0	0	0	1	0	0	1
12	0	0	0	0	0	0	0	0	0	1	0	1	0

Representació dels grafs

- Com representem un graf?
 - Linked list o array
 - **Matriu d'adjacència**
 - Llista d'adjacència

Que guardem per a les arestes?

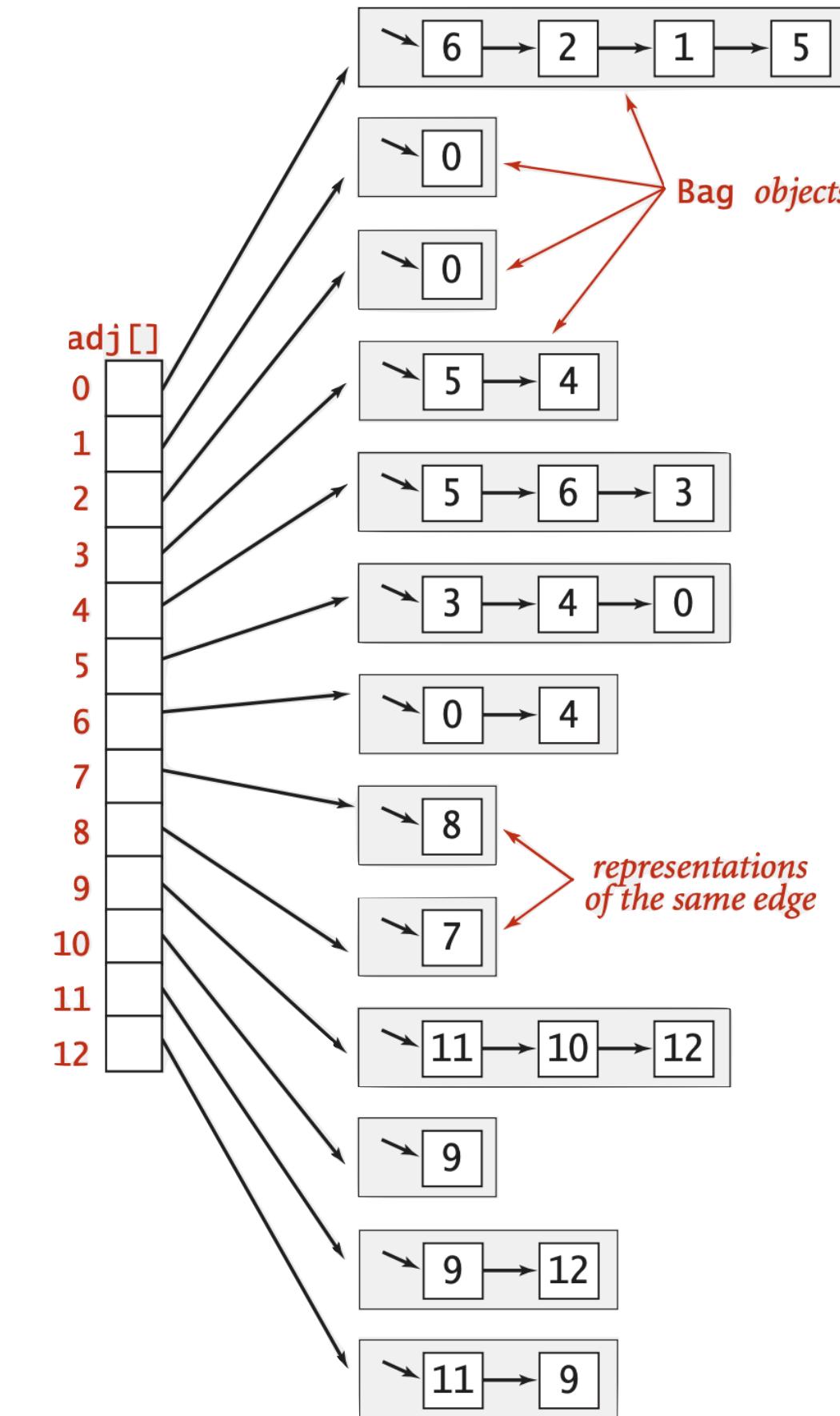
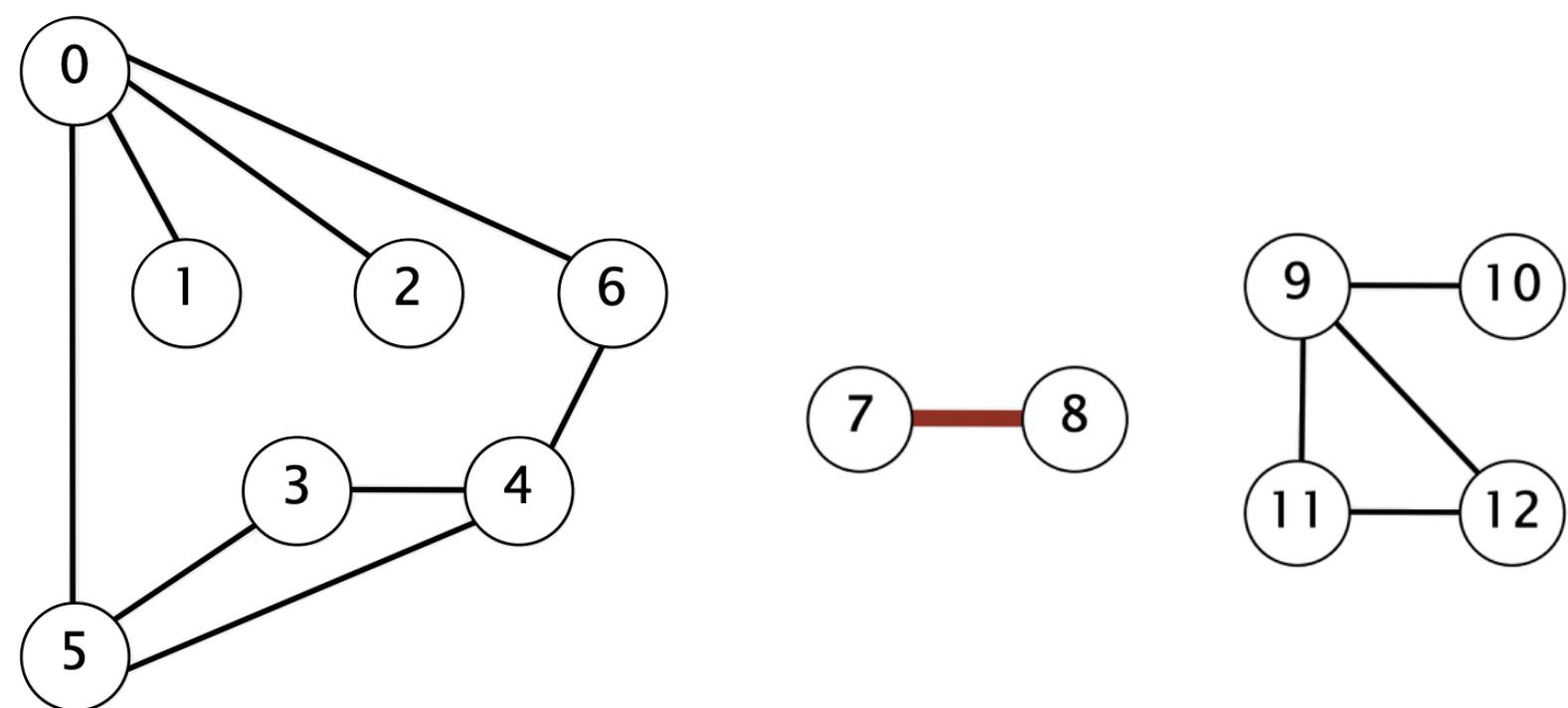
- 1/0 o True/false
- Pes/cost de l'aresta
 - utilitzem valors ficticis a les posicions on no tenim aresta {-∞,0,∞}
- Objecte de l'aresta

	1	2	3	4	5	6	7
1	∞	1	∞	∞	∞	∞	∞
2	∞	∞	∞	3	∞	15	∞
3	23	3	∞	8	∞	∞	∞
4	∞	∞	3	∞	∞	5	12
5	∞	∞	∞	∞	∞	∞	55
6	∞	1	4	4	∞	∞	∞
7	∞	∞	∞	∞	∞	6	∞

Representació dels grafs

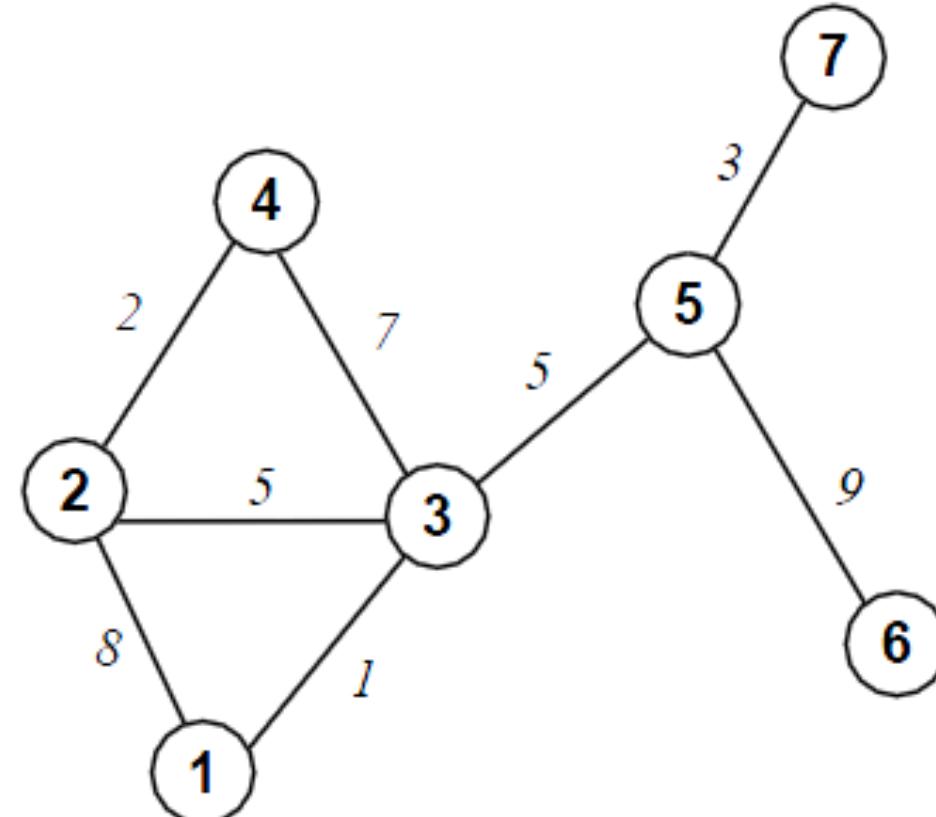
- Com representem un graf?

- Linked list o array
- Matriu d'adjacència
- **Llista d'adjacència**



Representació dels grafs

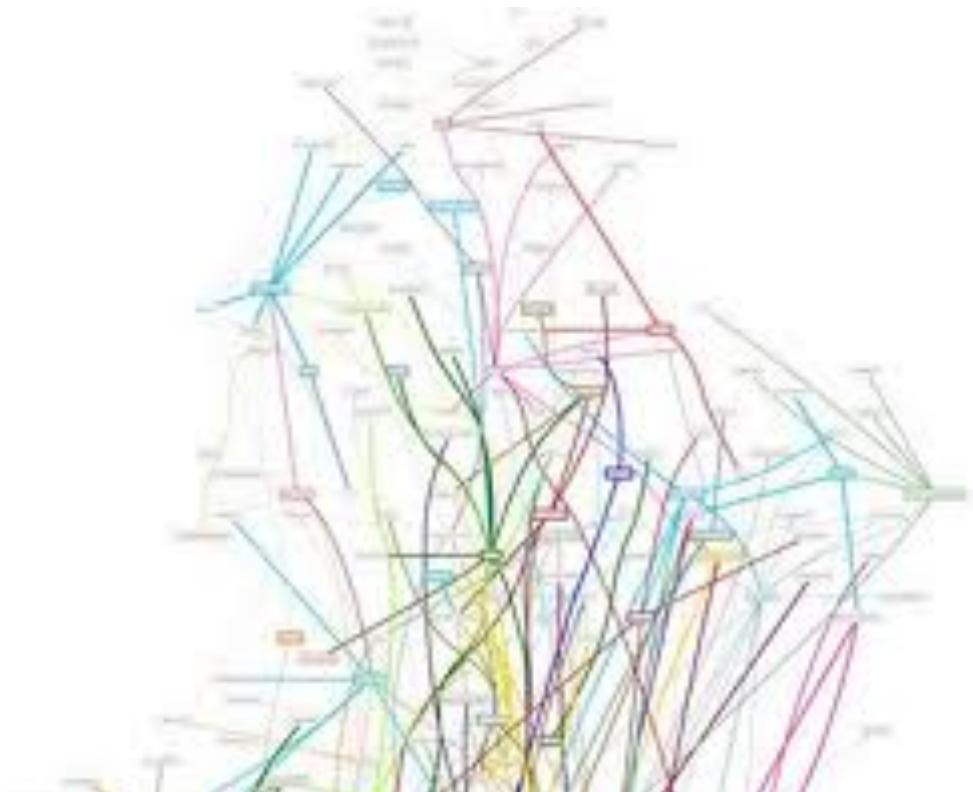
- Com representem un graf?
 - Linked list o array
 - Matriu d'adjacència
 - **Llista d'adjacència**



1	•	→	(2,8)	(3,1)		
2	•	→	(1,8)	(3,5)	(4,2)	
3	•	→	(1,1)	(4,7)	(2,5)	(5,5)
4	•	→	(2,2)	(3,7)		
5	•	→	(3,5)	(7,3)	(6,9)	
6	•	→	(5,9)			
7	•	→	(5,3)			

Algorismes sobre grafs

- Matriu ó llista d'adjacència?
 - Matriu $\rightarrow |V|^2$ posicions \rightarrow un accés
 - Llista $\rightarrow |E|$ posicions \rightarrow mínim un accés memòria versus localització
 - Graf **dens** versus graf **sparse**

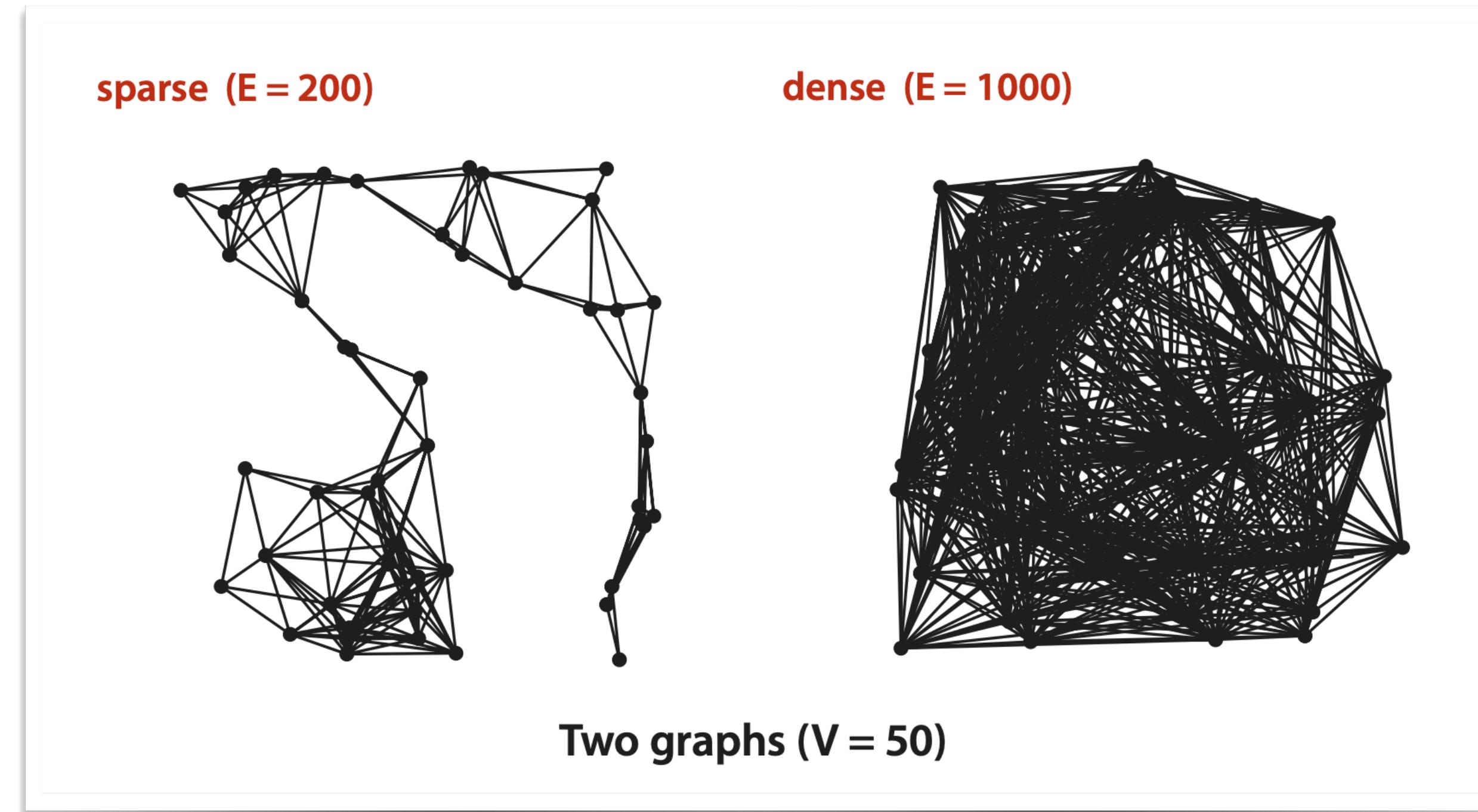


Què faríeu servir per codificar tots els enllaços de les pàgines del www?

Representació dels grafs

- A la pràctica: Quin utilitzar?
 - Els grafs reals acostumen a ser **sparse**

Alt nombre de nodes, grau
mitja del nodes petit



Representació dels grafs

- A la pràctica: Quin utilitzar?
 - Els algoritmes es basen en iterar sobre els nodes adjacents del *vertex*
 - Els grafs reals acostumen a ser **sparse**  Alt nombre de nodes, grau mitja del nodes petit

representation	space	add edge	edge between v and w?	iterate over vertices adjacent to v?
list of edges				
adjacency matrix				
adjacency lists				

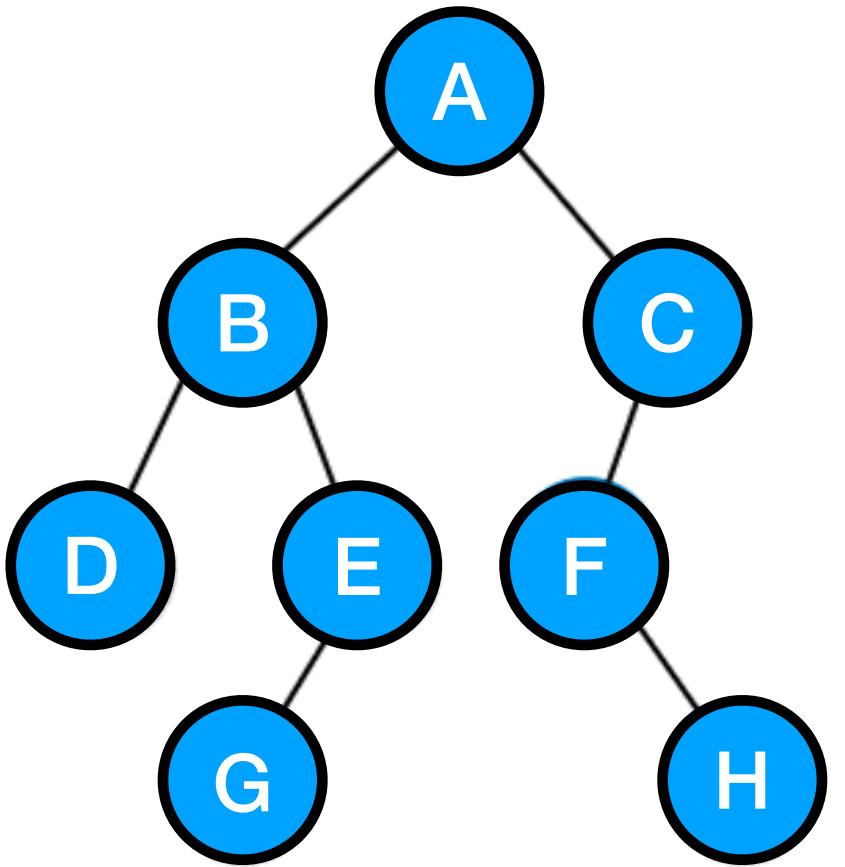
Representació dels grafs

- A la pràctica: Quin utilitzar?
 - Els algoritmes es basen en iterar sobre els nodes adjacents del vèrtex
 - Els grafs reals acostumen a ser **sparse** 
Alt nombre de nodes, grau mitja del nodes petit

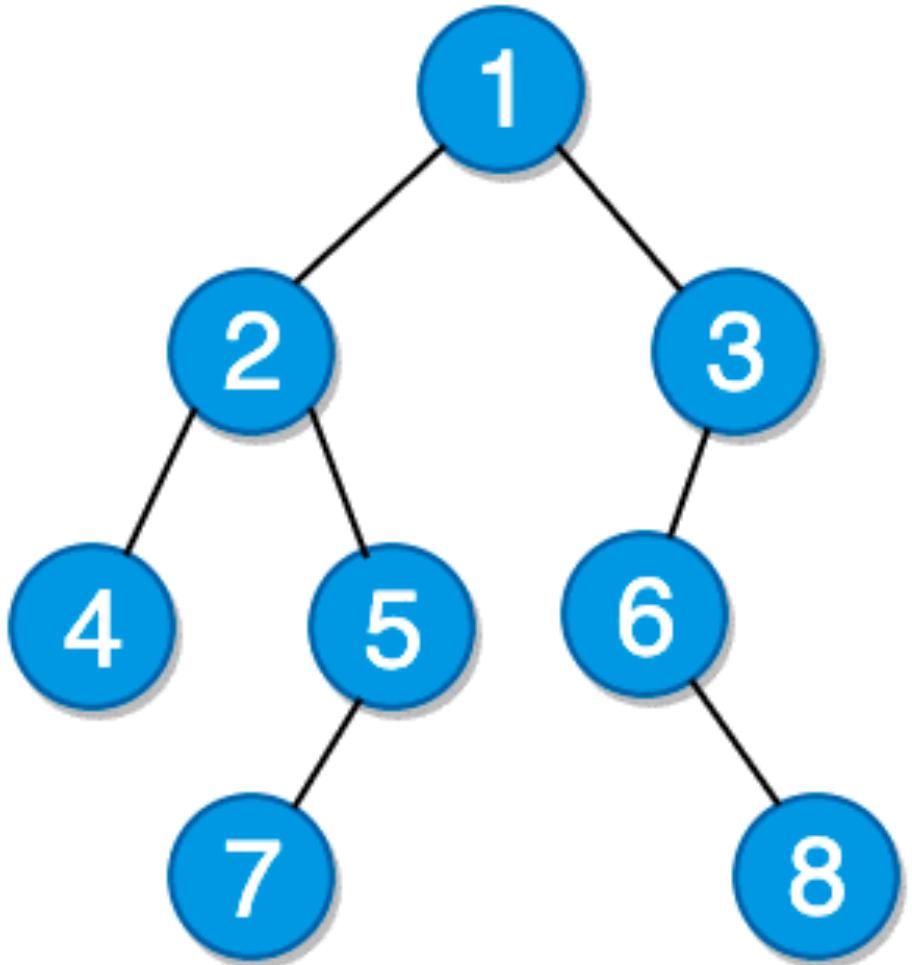
representation	space	add edge	edge between v and w?	iterate over vertices adjacent to v?
list of edges	E	1	E	E
adjacency matrix	V^2	1 *	1	V
adjacency lists	$E + V$	1	$\text{degree}(v)$	$\text{degree}(v)$

Recap: Recorregut Arbres

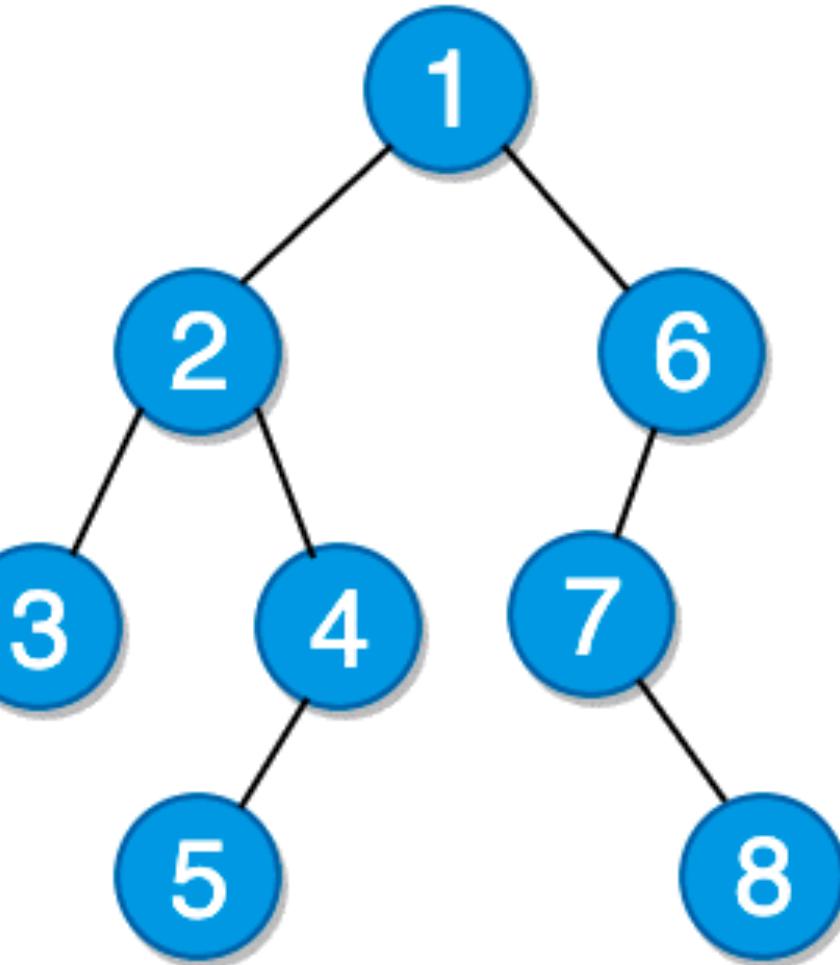
Recap: Recorregut



BFS



DFS



Donat el següent arbre:

```
# Using a Python dictionary to act as an adjacency list
graph = {
    'A' : ['B','C'],
    'B' : ['D', 'E'],
    'C' : ['F'],
    'D' : [],
    'E' : ['G'],
    'F' : ['H'],
    'G' : [],
    'H' : []
}
```

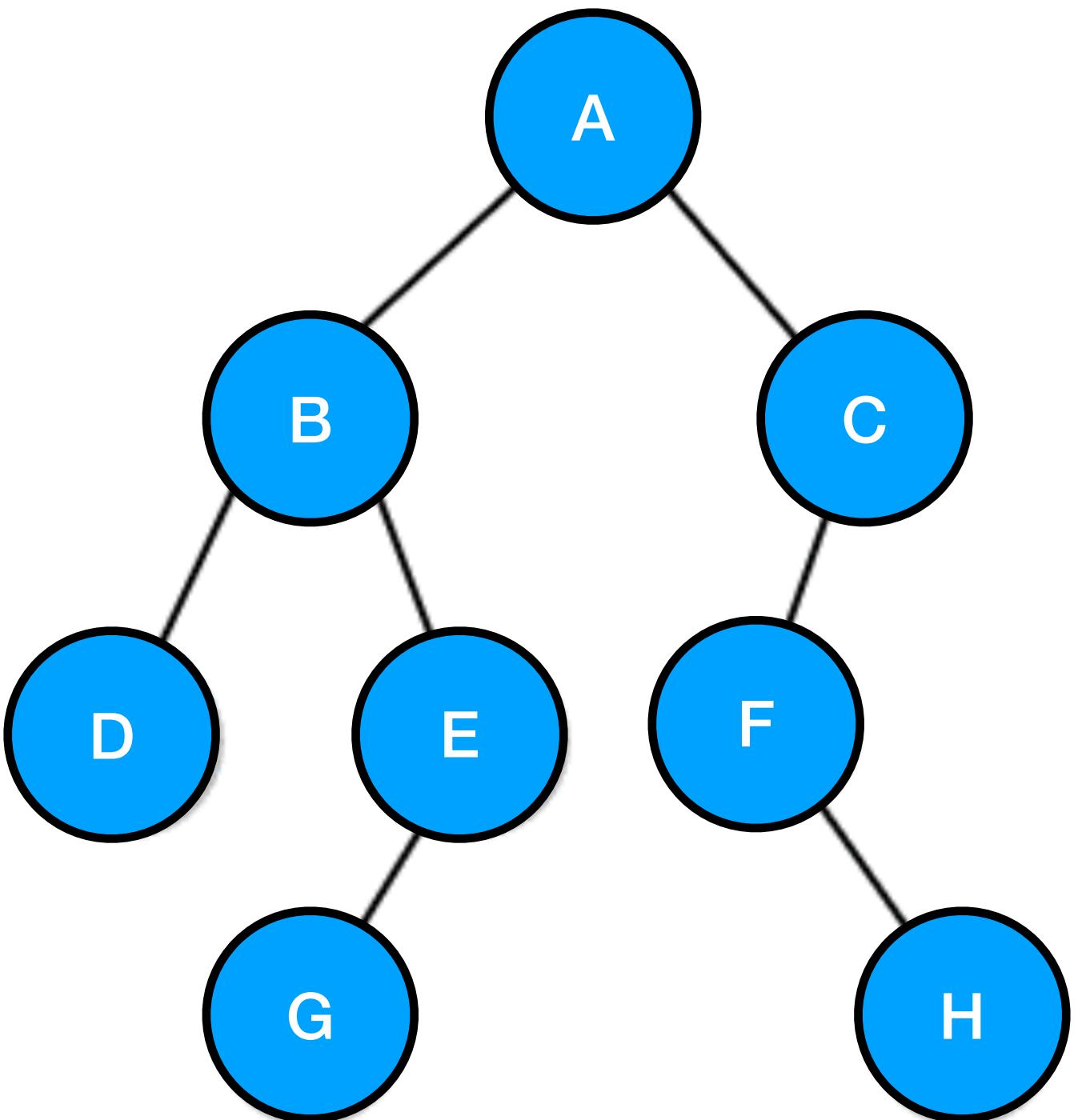
```
visited = set() # Set to keep track of visited nodes.
```

```
def explore(visited, graph, node):
    if node not in visited:
        print (node)
        visited.add(node)
        for neighbour in graph[node]:
            explore(visited, graph, neighbour)

# Driver Code
explore(visited, graph, 'A')
```

Quina és la sortida del codi anterior? Què fa?

Recap: Recorregut



Donat el següent arbre:

```
# Using a Python dictionary to act as an adjacency list
graph = {
    'A' : ['B','C'],
    'B' : ['D', 'E'],
    'C' : ['F'],
    'D' : [],
    'E' : ['G'],
    'F' : ['H'],
    'G' : [],
    'H' : []
}

visited = set() # Set to keep track of visited nodes.

def explore(visited, graph, node):
    if node not in visited:
        print (node)
        visited.add(node)
        for neighbour in graph[node]:
            explore(visited, graph, neighbour)

# Driver Code
explore(visited, graph, 'A')
```

Quina és la sortida del codi anterior? Què fa?