CS70 – Spring 2026

Lecture 5 : Feb. 3

Today: Graphs

Goals

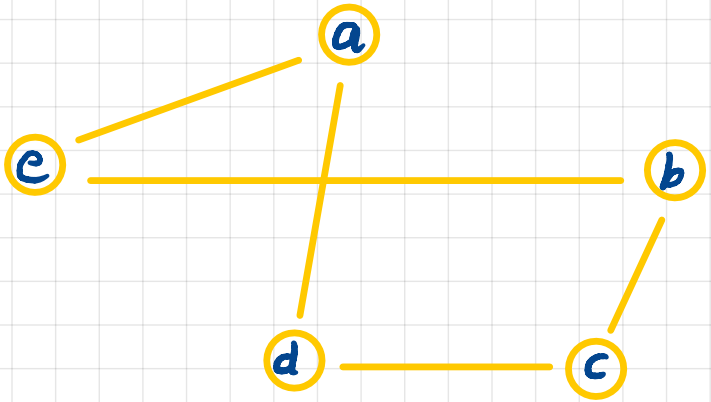
- Understand graphs as an important model for many phenomena in CS & real world
- Explore some basic properties of graphs:
  - paths in graphs
  - trees & complete graphs
  - planar graphs
  - connectivity & hypercubes

} next lecture

Graphs

$G = (V, E)$

vertices — edges

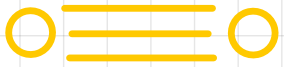


$V = \{a, b, c, d, e\}$

$E = \{\{a,e\}, \{a,d\}, \{e,b\}, \{b,c\}, \{c,d\}\}$

Undirected graph

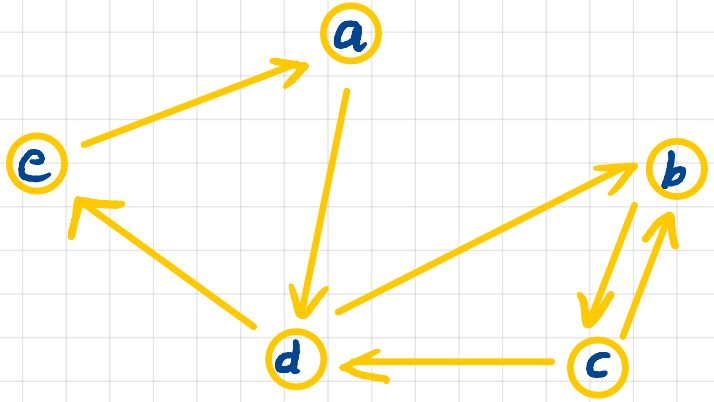
An edge is an unordered pair of vertices

Unless otherwise stated, we don't allow loops  or multiple edges 

I.e., our graphs are simple graphs, not multigraphs

Directed Graphs

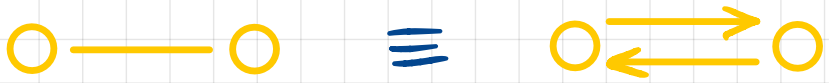
$G = (V, E)$



$V = \{a, b, c, d, e\}$

$E = \{(e,a), (d,e), (a,d), (d,b), (b,c), (c,b), (c,d)\}$

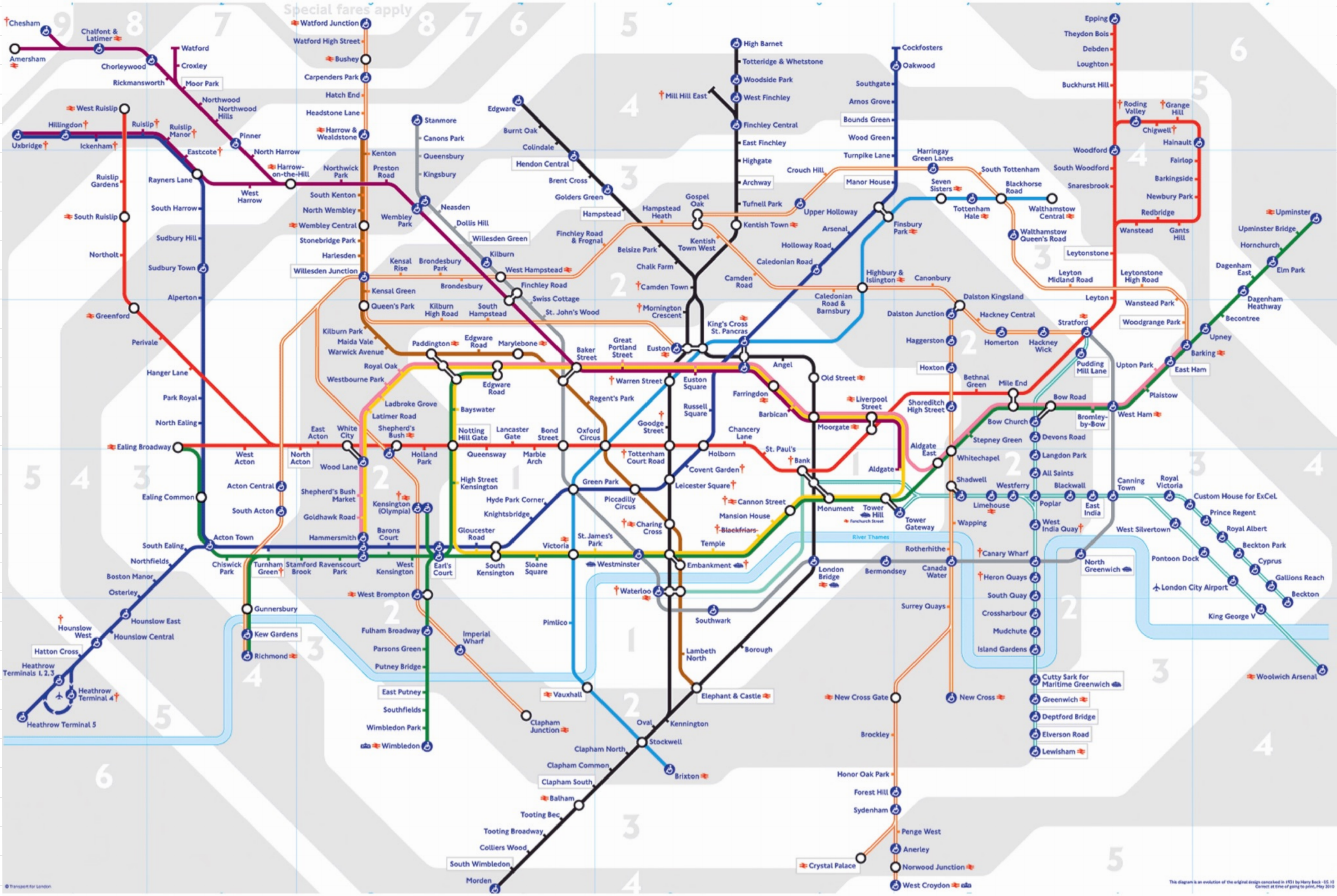
Same, except edges are now ordered pairs of vertices

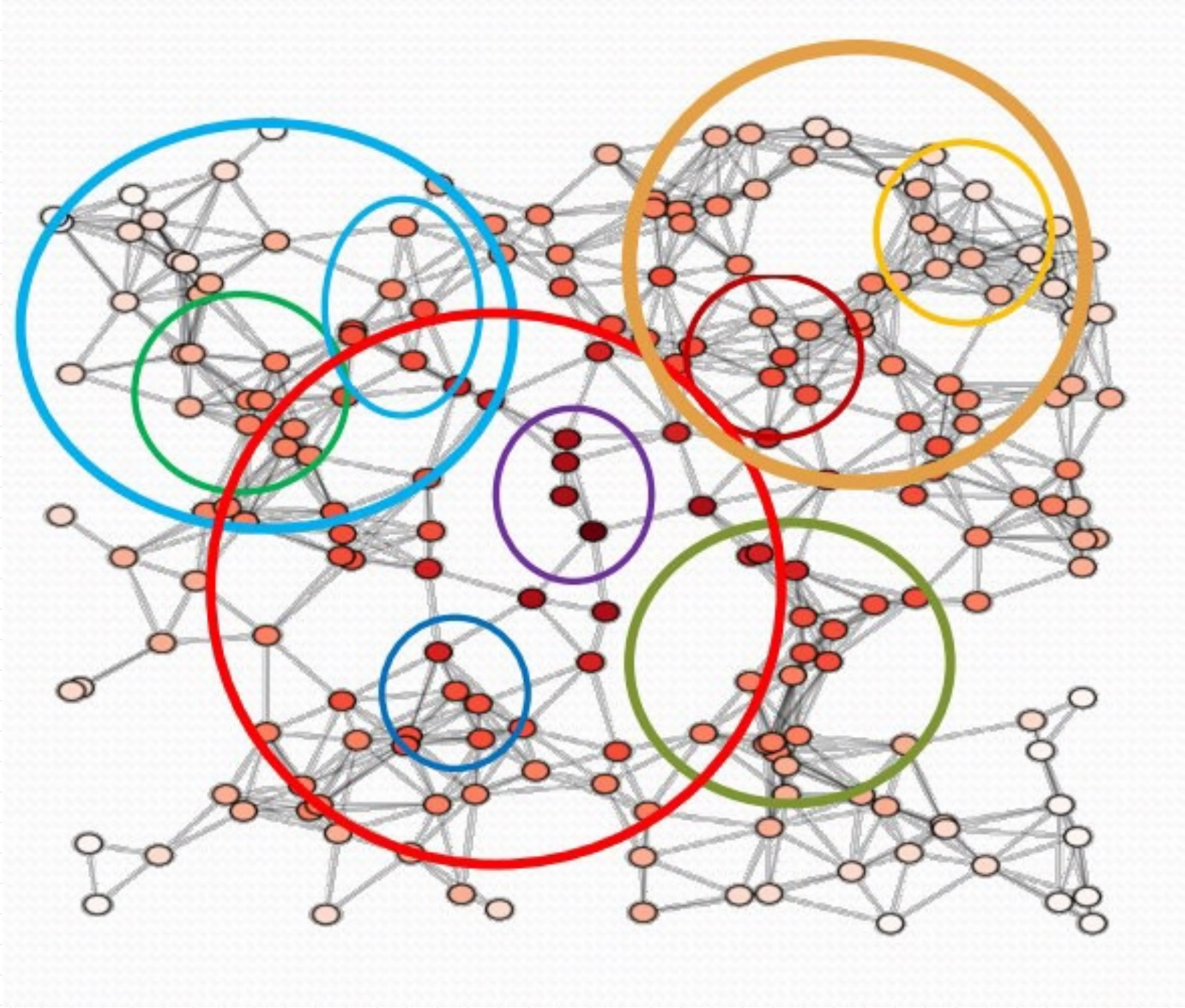
Note: Undirected graphs are a special case of directed graphs: 

Examples of Undirected Graphs

- road networks (no one-way streets)
- power grids
- social networks ("knows")
- graphical models (ML)

⋮


Special fares apply
Chesham
Amersham
Chalfont & Latimer
Chorleywood
Rickmansworth
Watford
Croxley
Moor Park
Northwood
Northwood Hills
Pinner
North Harrow
Harrow-on-the-Hill
West Harrow
West Ruislip
Ruislip
Ruislip Manor
Hillingdon
Uxbridge
Ickenham
Eastcote
Ruislip Gardens
Rayners Lane
South Ruislip
South Harrow
Sudbury Hill
Northolt
Sudbury Town
Alperton
Greenford
Perivale
Hanger Lane
Park Royal
North Ealing
Ealing Broadway
West Acton
North Acton
East Acton
White City
Ealing Common
Acton Central
South Acton
Acton Town
South Ealing
Northfields
Boston Manor
Osterley
Hounslow East
Hounslow West
Hounslow Central
Hatton Cross
Heathrow Terminals 1, 2, 3
Heathrow Terminal 4
Heathrow Terminal 5
Chiswick Park
Turnham Green
Stamford Brook
Ravenscourt Park
Gunnersbury
Kew Gardens
Richmond
Watford Junction
Watford High Street
Bushey
Carpenders Park
Hatch End
Headstone Lane
Harrow & Wealdstone
Kenton
South Kenton
North Wembley
Wembley Central
Stonebridge Park
Harlesden
Willesden Junction
Kensal Green
Queen's Park
Kilburn Park
Maida Vale
Warwick Avenue
Paddington
Royal Oak
Westbourne Park
Ladbroke Grove
Latimer Road
Shepherd's Bush
Wood Lane
Shepherd's Bush Market
Goldhawk Road
Hammersmith
Barons Court
Holland Park
Kensington (Olympia)
West Kensington
Earl's Court
West Brompton
Fulham Broadway
Parsons Green
Putney Bridge
East Putney
Southfields
Wimbledon Park
Wimbledon
Imperial Wharf
Clapham Junction
Northwick Park
Preston Road
Wembley Park
Stanmore
Canons Park
Queensbury
Kingsbury
Neasden
Dollis Hill
Willesden Green
Kilburn
West Hampstead
Finchley Road
Swiss Cottage
St. John's Wood
Kensal Rise
Brondesbury Park
Brondesbury
Kilburn High Road
South Hampstead
Edgware
Burnt Oak
Colindale
Hendon Central
Brent Cross
Golders Green
Hampstead
Finchley Road & Frognal
Belsize Park
Chalk Farm
Camden Town
Mornington Crescent
Hampstead Heath
Gospel Oak
Kentish Town West
Kentish Town
Camden Road
High Barnet
Totteridge & Whetstone
Woodside Park
West Finchley
Mill Hill East
Finchley Central
East Finchley
Highgate
Archway
Tufnell Park
Edgware Road
Marylebone
Baker Street
Great Portland Street
Euston
Warren Street
Euston Square
Regent's Park
Bayswater
Notting Hill Gate
Queensway
Lancaster Gate
Marble Arch
Bond Street
Oxford Circus
Goodge Street
Tottenham Court Road
Russell Square
Holborn
Chancery Lane
Farringdon
Barbican
King's Cross St. Pancras
Angel
Old Street
Moorgate
Liverpool Street
St. Paul's
Bank
Aldgate
Aldgate East
High Street Kensington
Hyde Park Corner
Knightsbridge
Gloucester Road
South Kensington
Sloane Square
Victoria
St. James's Park
Westminster
Green Park
Piccadilly Circus
Leicester Square
Covent Garden
Charing Cross
Embankment
Temple
Blackfriars
Mansion House
Cannon Street
Monument
Tower Hill
Fenchurch Street
Tower Gateway
Waterloo
Southwark
London Bridge
Borough
Lambeth North
Elephant & Castle
Pimlico
Vauxhall
Oval
Kennington
Stockwell
Brixton
Clapham North
Clapham Common
Clapham South
Balham
Tooting Bec
Tooting Broadway
Colliers Wood
South Wimbledon
Morden
River Thames
Cockfosters
Oakwood
Southgate
Arnos Grove
Bounds Green
Wood Green
Turnpike Lane
Manor House
Harringay Green Lanes
Crouch Hill
Upper Holloway
Finsbury Park
Arsenal
Holloway Road
Caledonian Road
Caledonian Road & Barnsbury
Highbury & Islington
Canonbury
Dalston Kingsland
Dalston Junction
Haggerston
Hoxton
Shoreditch High Street
Whitechapel
Shadwell
Wapping
Rotherhithe
Canada Water
Surrey Quays
New Cross Gate
New Cross
Brockley
Honor Oak Park
Forest Hill
Sydenham
Penge West
Anerley
Norwood Junction
West Croydon
Crystal Palace
Bermondsey
Seven Sisters
Tottenham Hale
South Tottenham
Blackhorse Road
Walthamstow Central
Walthamstow Queen's Road
Leytonstone
Leyton Midland Road
Leytonstone High Road
Leyton
Wanstead Park
Woodgrange Park
Hackney Central
Homerton
Hackney Wick
Stratford
Pudding Mill Lane
Bethnal Green
Mile End
Bow Road
Bow Church
Stepney Green
Bromley-by-Bow
West Ham
Plaistow
Upton Park
East Ham
Barking
Upney
Becontree
Dagenham Heathway
Dagenham East
Elm Park
Hornchurch
Upminster Bridge
Upminster
Devons Road
Langdon Park
All Saints
Limehouse
Westferry
Poplar
Blackwall
East India
West India Quay
Canary Wharf
Heron Quays
South Quay
Crossharbour
Mudchute
Island Gardens
Cutty Sark for Maritime Greenwich
Greenwich
Deptford Bridge
Elverson Road
Lewisham
Canning Town
North Greenwich
Royal Victoria
Custom House for ExCeL
Prince Regent
Royal Albert
Beckton Park
Cyprus
Gallions Reach
Beckton
West Silvertown
Pontoon Dock
London City Airport
King George V
Woolwich Arsenal
Epping
Theydon Bois
Debden
Loughton
Buckhurst Hill
Roding Valley
Grange Hill
Chigwell
Hainault
Woodford
South Woodford
Fairlop
Barkingside
Snaresbrook
Newbury Park
Redbridge
Wanstead
Gants Hill
© Transport for London

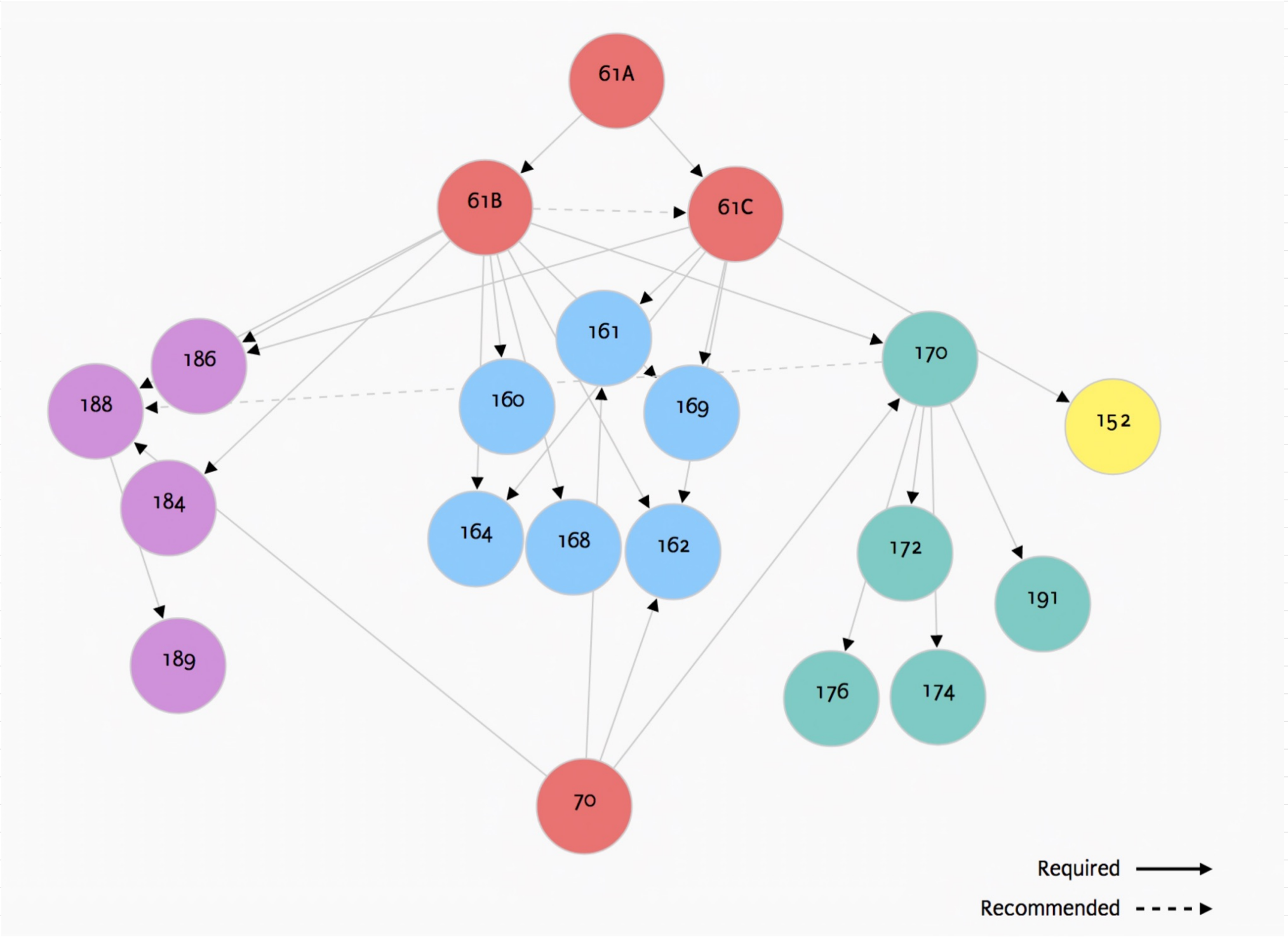


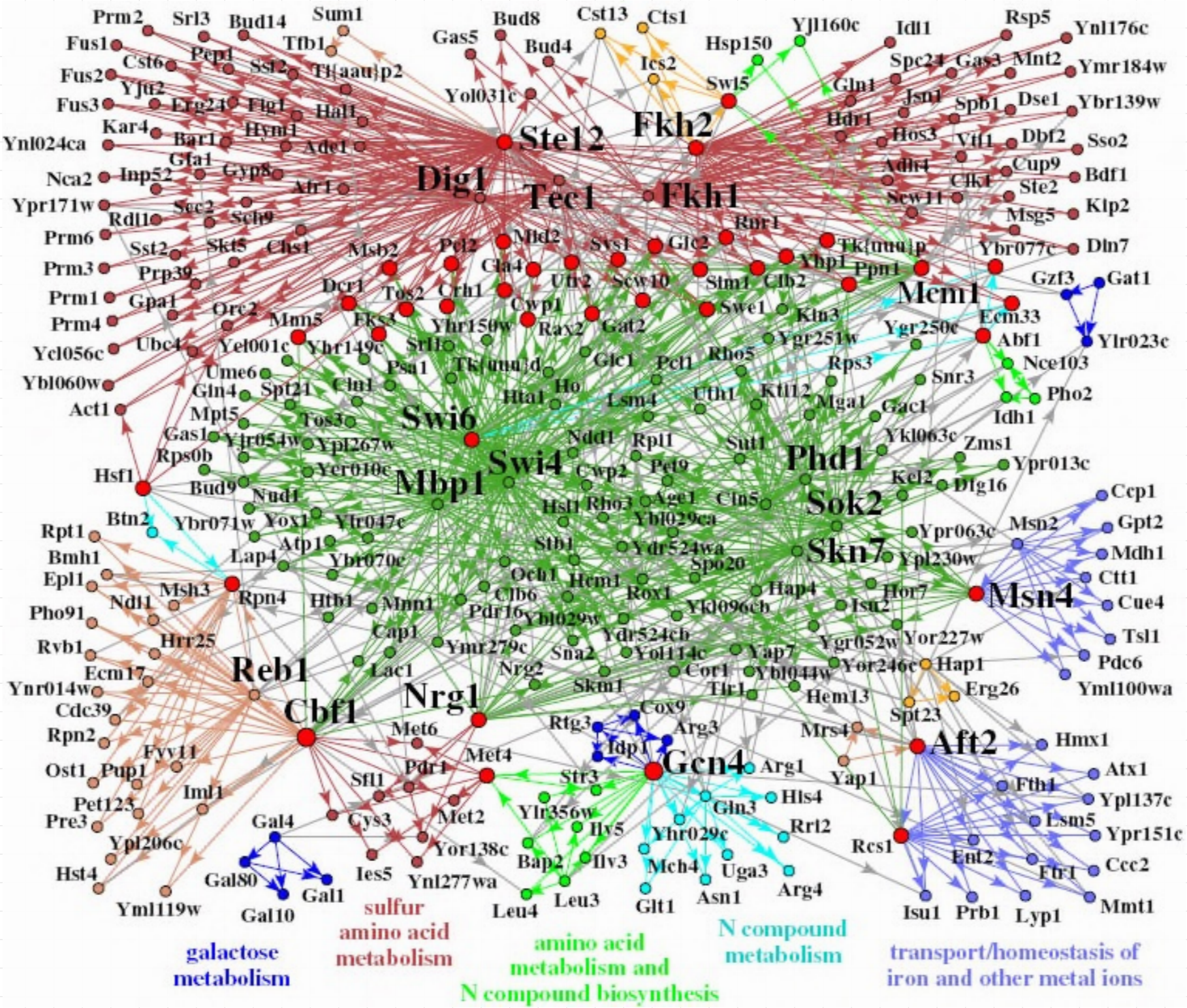
Renren Social Network: Chen Wen Xin

Examples of Directed Graphs

- internet (web links)
- course prerequisites
- biological networks
- social networks ("recognizes")

⋮


61A
61B
61C
186
188
161
160
169
170
152
184
164
168
162
172
191
189
176
174
70
Required
Recommended


Ste12
Fkh2
Dig1
Tec1
Fkh1
Swi6
Swi4
Mbp1
Phd1
Sok2
Skn7
Mcm1
Msn4
Reb1
Cbf1
Nrg1
Gcn4
Aft2
galactose metabolism
sulfur amino acid metabolism
amino acid metabolism and N compound biosynthesis
N compound metabolism
transport/homeostasis of iron and other metal ions

SocNetV 2.5 - cp.net

Network Edit Analyze Layout Options Help

cp.net

Control Panel

Network

Auto Create: Select

Subgraph: Select

Edge Mode: Directed

Transform: Select

Analyze

Matrix: Select

Cohesion: Select

Prominence: Select

Communities: Select

Equivalence: Select

Layout

By Prominence Index

Index: Degree Centrality

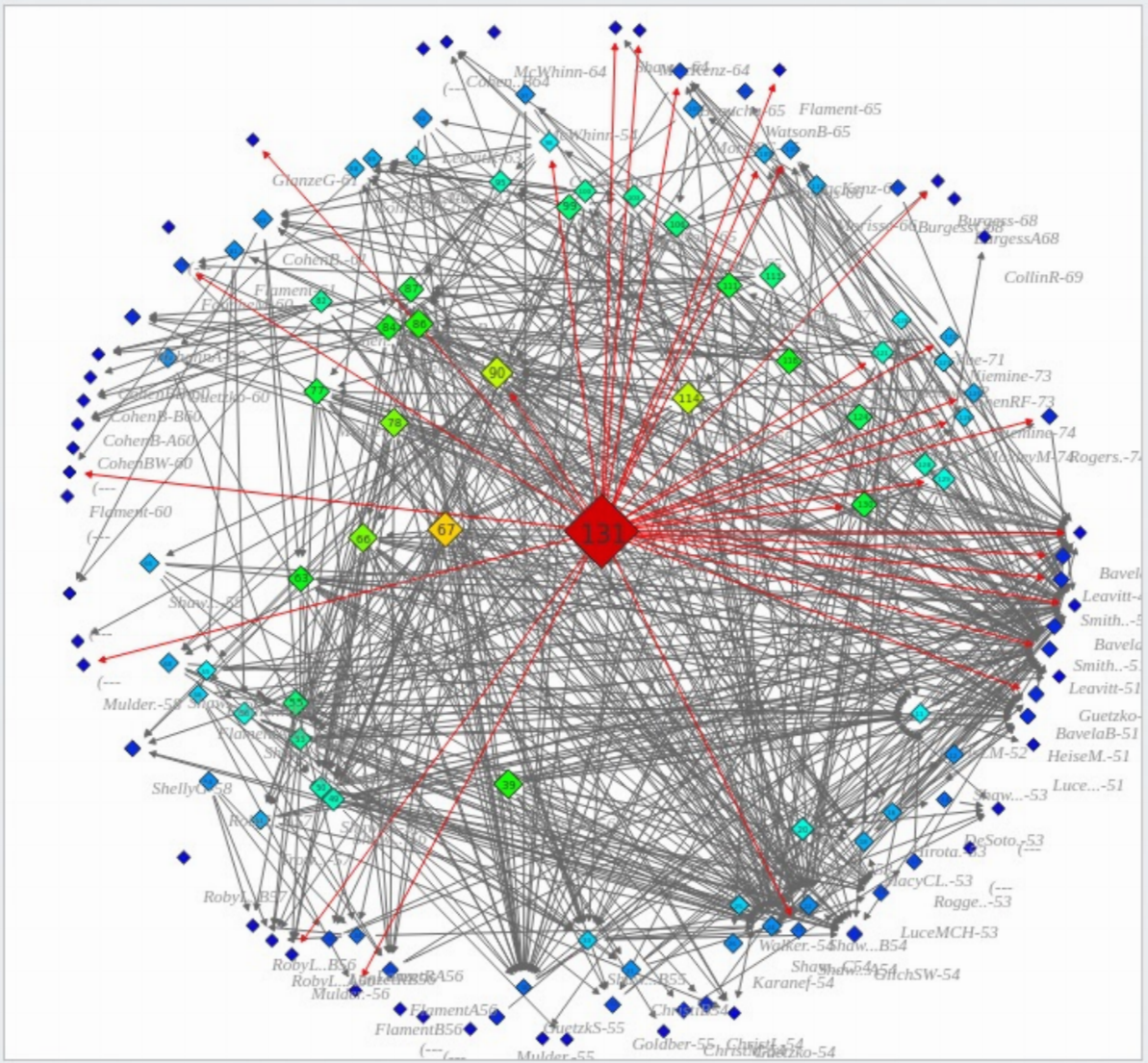
Type: Node Color

Apply

By Force-Directed Model

Model: Kamada-Kawai

Apply



Statistics Panel

Network

Type:	Directed
Nodes:	131
Arcs:	631
Density:	0.0370523

Selection

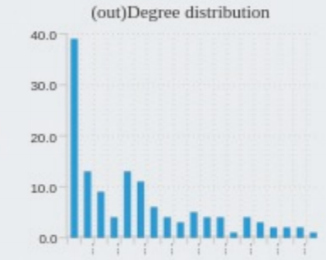
Nodes:	1
Arcs:	0

Clicked Node

Number:	131
In-Degree:	0
Out-Degree:	29
Clu.Coef.	0

Clicked Edge

Name:	-
Weight:	-



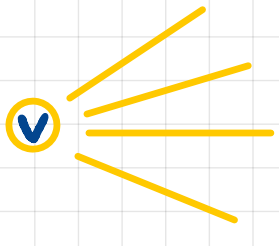
https://socnetv.org

Some Terminology

$e = \{u, v\}$ edge

u, v are adjacent or neighbors

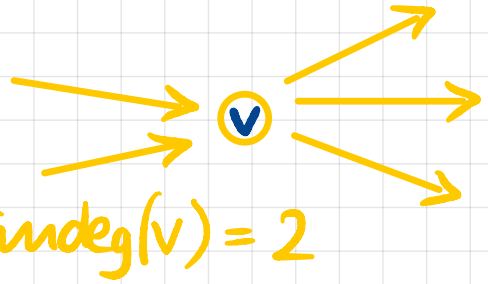
e is incident on u & v



degree of v = number of neighbors

If v has degree 0 it is isolated

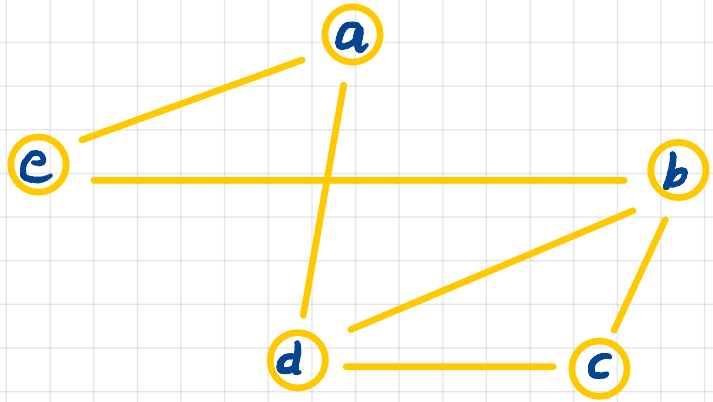
For directed graphs:



Q: What is the sum of all vertex degrees in G?

A: $\sum_{v \in V} \deg(v) = 2|E|$

More Terminology



E.g.

path: $e-a-d-c$

cycle: $d-c-b-d$

A (simple) <u>path</u> is a sequence of edges

$\{v_1,v_2\}, \{v_2,v_3\}, \{v_3,v_4\}, \ldots, \{v_{k-1},v_k\}$

s.t. all the v_i are distinct (except possibly $v_k = v_1$)

[Equivalently: $v_1 - v_2 - v_3 - \cdots - v_k$]

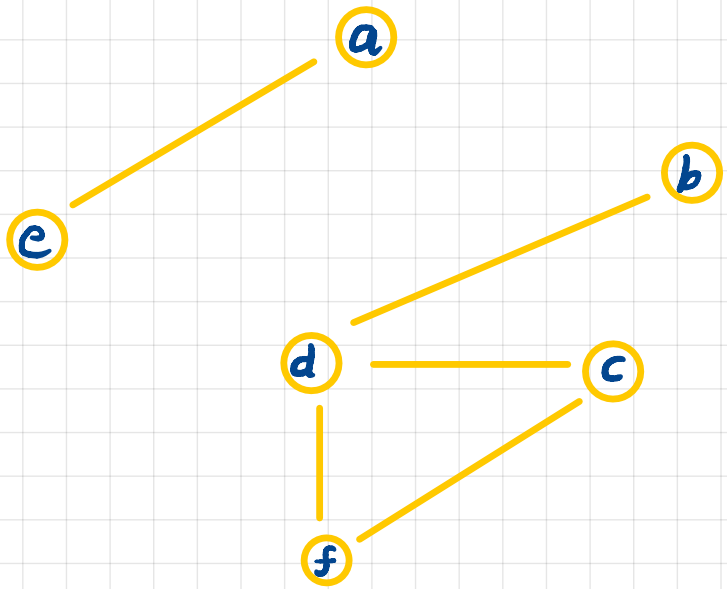
A <u>cycle</u> is a path where $v_1 = v_k$

A <u>walk</u> is any sequence of edges as above (repeated vertices/edges allowed)

A <u>tour</u> is a walk where $v_1 = v_k$

A undirected graph is connected if there is a path from any vertex to any other vertex

Any graph can be decomposed into connected components



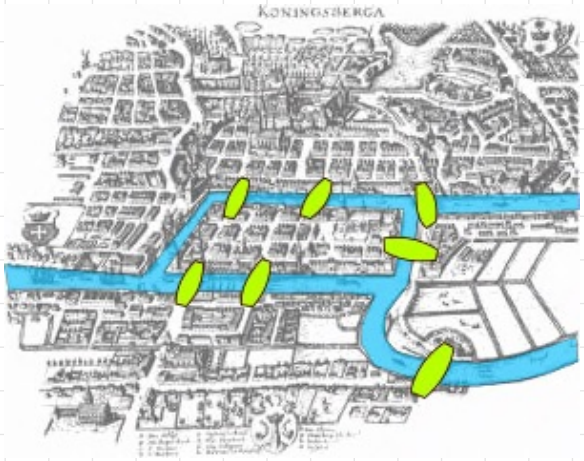
A directed graph is strongly connected if ∃ (directed) path from every vertex to every other vertex

Graph Theory : Topics

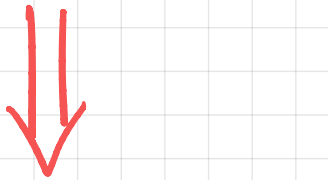
- Eulerian tours
- Trees & complete graphs
- Planar graphs
- Connectivity & hypercubes

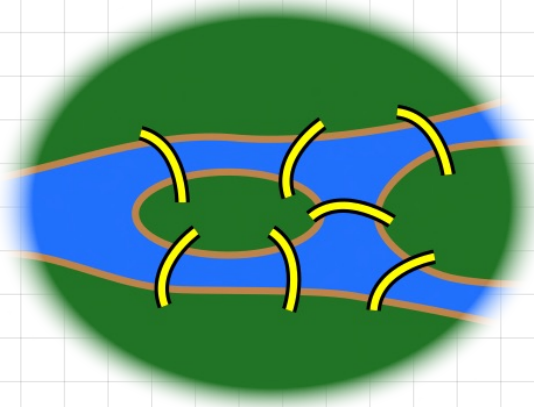
} next lecture

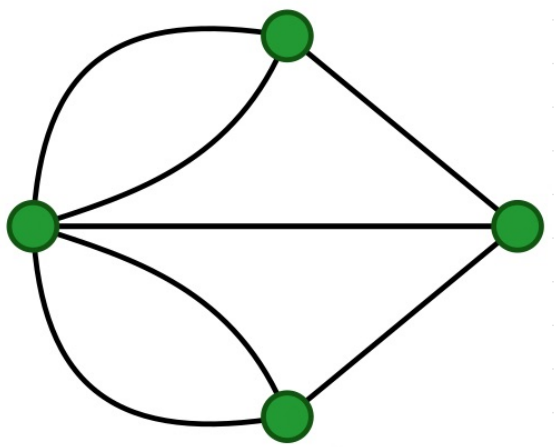
Euler: Bridges of Königsberg (1736)



Can we cross all 7 bridges once & end up where we started?





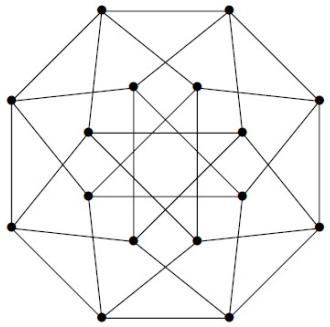


(multi) graph

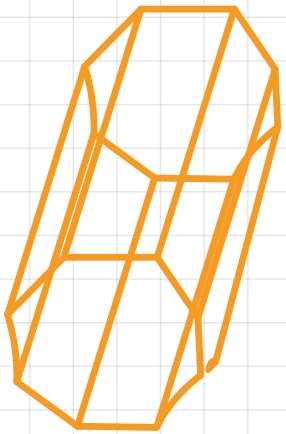
Defn: An **Eulerian tour** of a graph is a {tour / walk} that traverses each edge exactly once and ends up at the starting vertex

Theorem [Euler]: A connected graph has an Eulerian tour $\iff$ the degree of every vertex is even

Examples

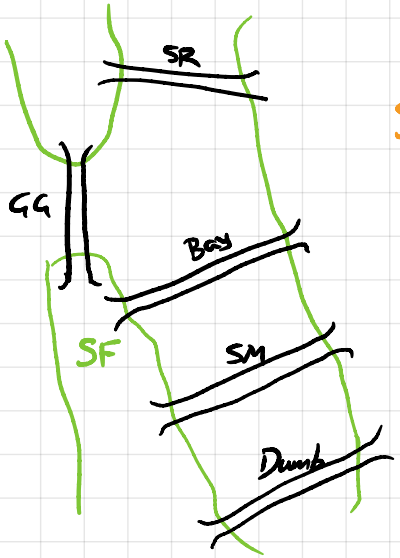


✓



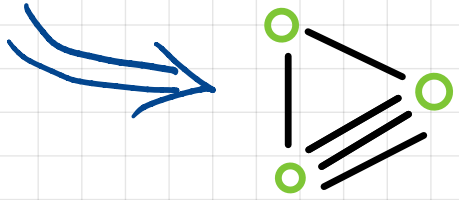
✗

octagonal "lampshade"



SF Bay bridges

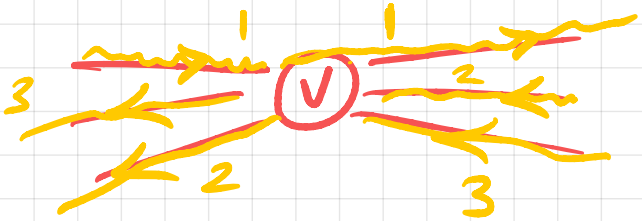
✓



Theorem [Euler]: A connected graph has an Eulerian tour $\iff$ the degree of every vertex is even

Proof: $\Rightarrow$ In any Eulerian tour, entries to & exits from a vertex occur in pairs, using different edges:

E.g.

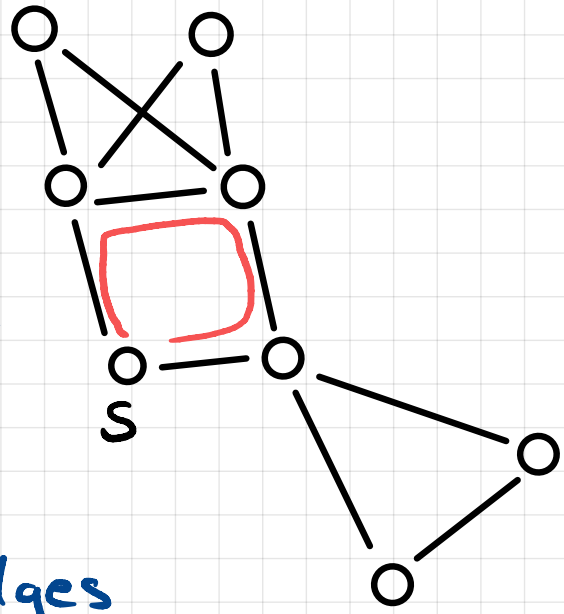


$\Rightarrow$ can pair up incoming & outgoing edges at v on the tour

$\Rightarrow$ # of edges inc. at v is even

$\Leftarrow$ Assuming G is connected & all vertex degrees are even — give an algorithm for constructing an E. tour

□



subroutine FindTour(G, s)

start at s

repeat

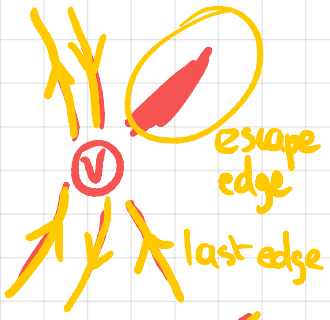
choose any untraversed edge incident on current vertex & traverse it

until get stuck

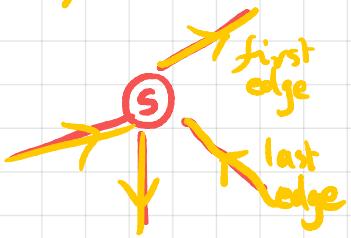
return the tour formed by traversed edges

Claim: FindTour(G, s) always gets stuck at s

Proof: For any vertex $v \neq s$, whenever our tour arrives at v it has used up an **odd** # of v's edges.



Hence there is always an "escape" edge available at v



Whenever tour arrives **at s**, it has used up an **even** no. of s's edges — so it **can** get stuck at s.

Since must get stuck eventually, this must be at s □

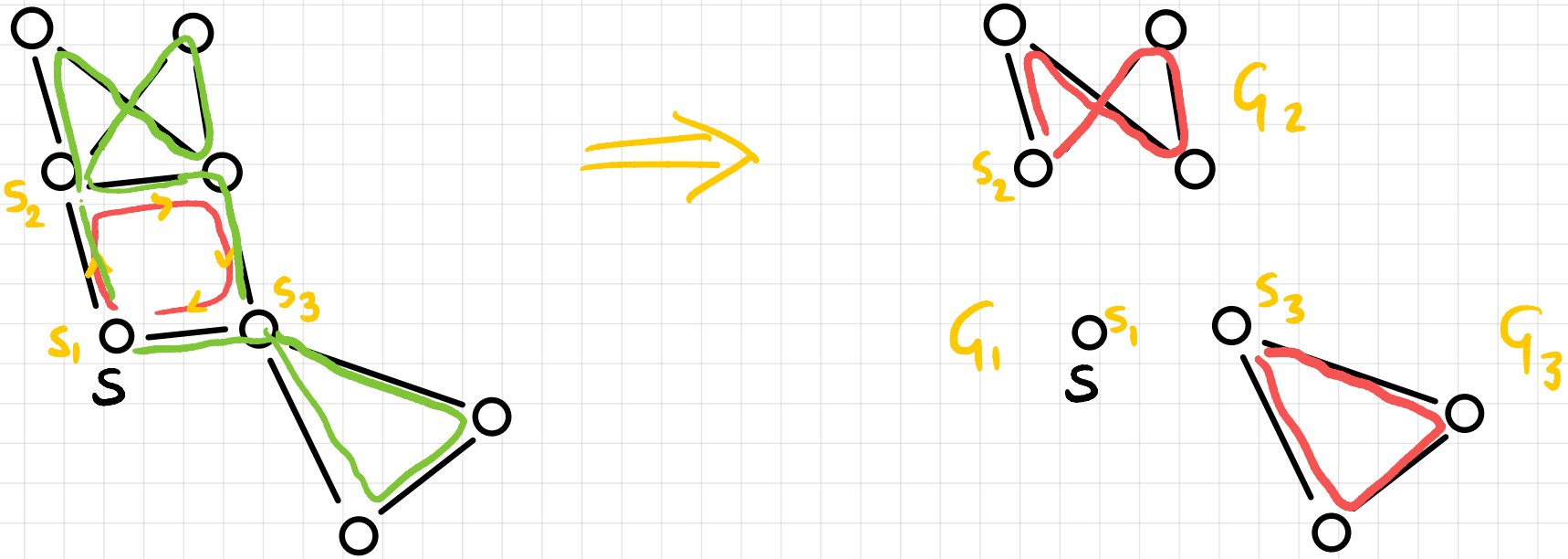
Euler (G, s)

$T =$ FindTour (G, s)

remove edges of T from G

let $\{G_1, \ldots, G_k\}$ be the connected components of the remaining graph, and s_i the first vertex of G_i visited by T

output (Splice $(T,$ Euler $(G_1, s_1), \ldots,$ Euler $(G_k, s_k))$)



Theorem: For any connected G with even degrees, Euler(G, s) outputs an Euler tour starting & ending at s

Proof: (Strong) induction on # edges m in G

Base: $m = 0$ ✓ (nothing to prove)

Ind. step: Assume true for any graph with $\leq m$ edges.

Sp. G has $m+1$ edges.

Removing T = FindTour(G, s) from G leaves G' with connected components $G_1, \ldots, G_k$, all having <u>even</u> degrees (because T has even degrees!)

By strong induction hypothesis, Euler(G_i, s_i) finds an E. tour in each G_i

Hence splice(...) returns an E. tour in G □

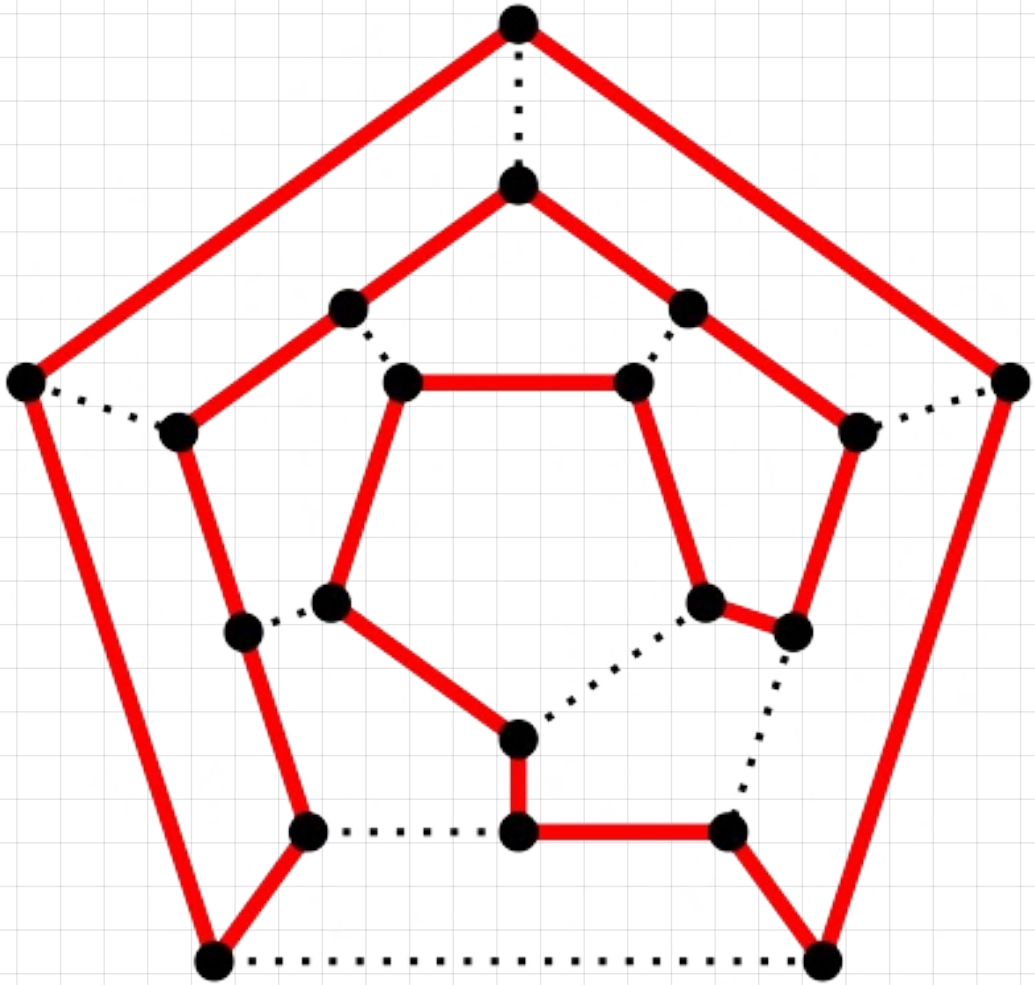
Eulerian Tours – Wrap-Up

1. **Exercise**: modify proof of Euler's Thm. to prove:

   Connected G has an Eulerian walk starting at s & ending at t $\iff$ all vertices of G except s & t have even degree

2. A Hamilton cycle in G is a cycle in G that visits every vertex exactly once

   - no simple characterization
   - NP-complete (related to Traveling Salesman)



Example of a graph with a Hamilton cycle
("dodecahedron" graph)

Graph Theory : Topics

- Eulerian tours
- Trees & complete graphs
- Planar graphs
- Connectivity & hypercubes

} next lecture

Trees

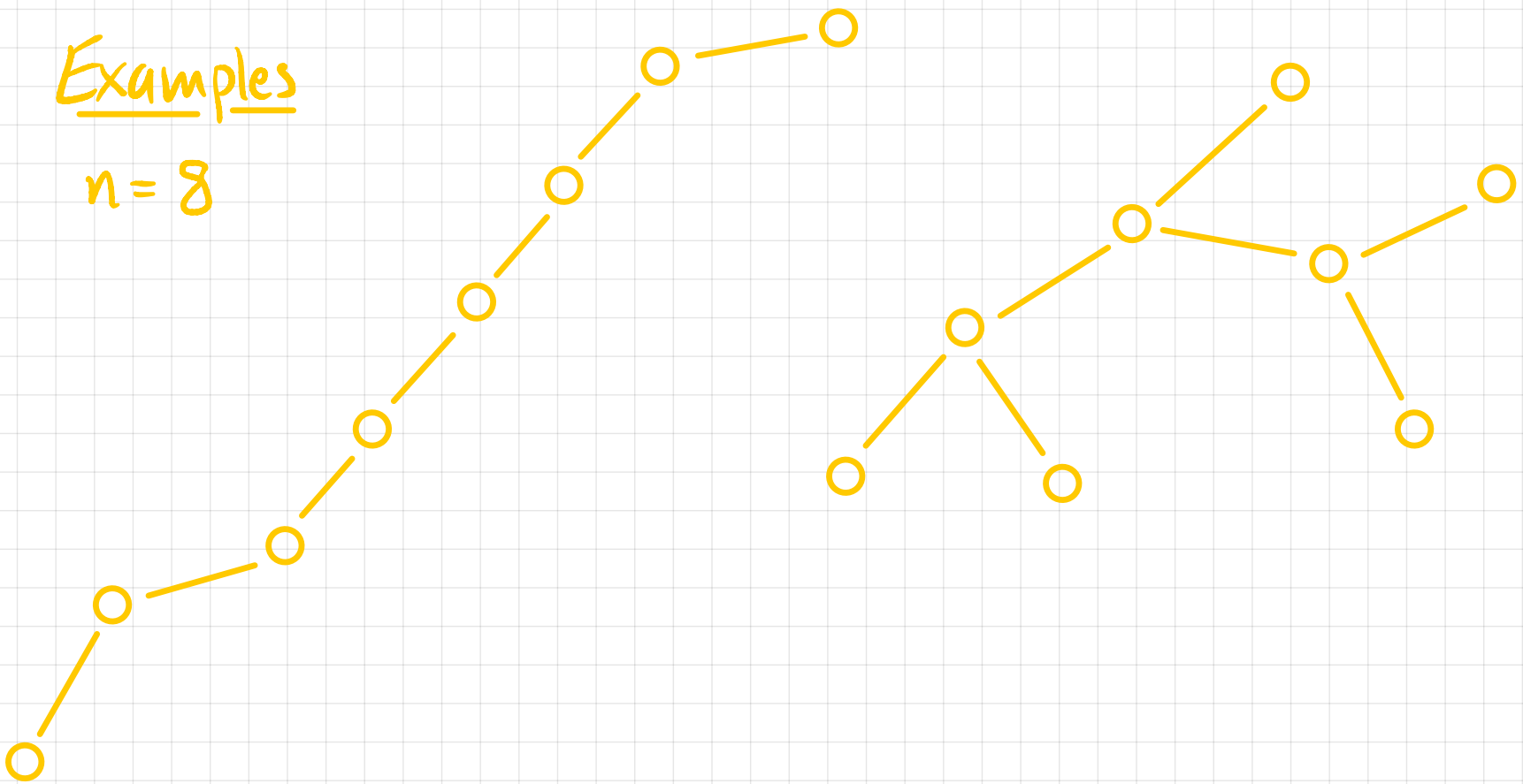
An undirected graph G with n vertices is a tree if either of the following equiv. conditions holds:

(i) G is connected & has $n-1$ edges

(ii) G is connected & has no cycles

Examples

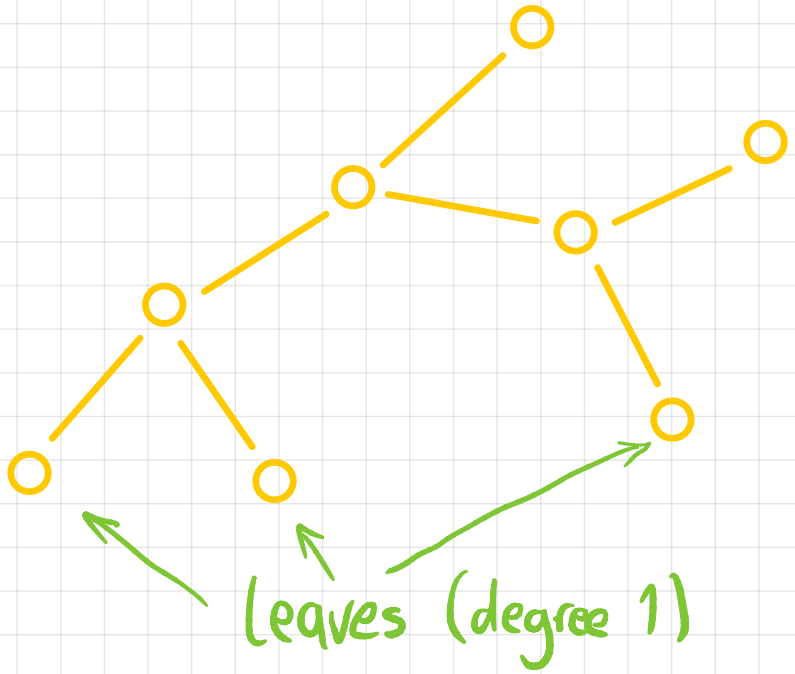
$n = 8$

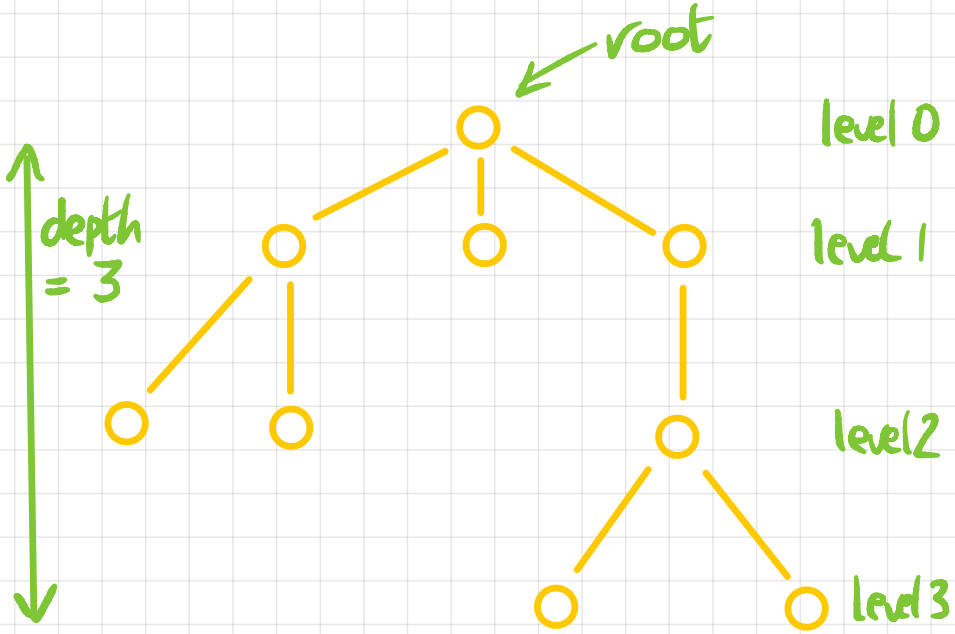


Trees show up in:

- data structures (BST, Red-Black trees, ...)
- phylogenetic trees (evolutionary biology, genealogy)
- epidemic models
- decision trees

⋮





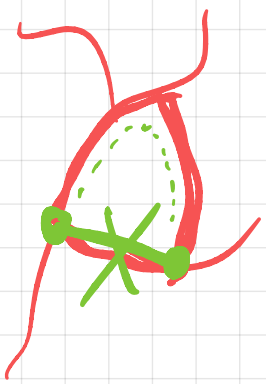
rooted tree

Theorem: G is connected & has $n-1$ edges $\iff$ G is connected & has no cycles

Proof: $\Rightarrow$ Proof by contradiction.

Let G be connected & have $n-1$ edges.

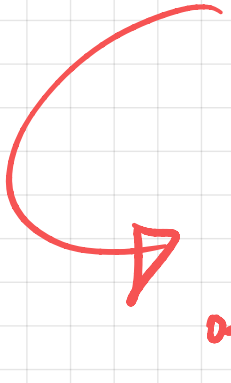
Sp. for ※ that G contains a cycle



Then we can delete any edge of the cycle and G remains connected

So we get a graph with only $n-2$ edges that is connected.

But this is a ※ since we need at least $n-1$ edges to connect n vertices □



Each edge reduces # of components by at most 1
Start with n components & end with 1 component

Theorem: G is connected & has $n-1$ edges $\iff$ G is connected & has no cycles

Proof: $\Leftarrow$ Proof by induction on n.

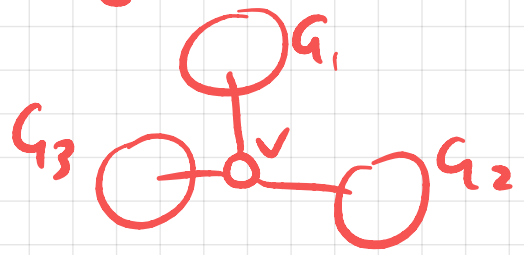
Base case: $n=1$ ○ isolated vertex $n-1=0$ edges ✓

Inductive Step: Assume true for $1 \leq n \leq k$

Prove for $n = k+1$

Sp. G has $k+1$ vertices, connected, no cycles.

Remove one vertex v from G → new graph G' that still has no cycles.



G' consists of connected components

$G_1, G_2, \ldots, G_\ell$ — say G_i has k_i vertices

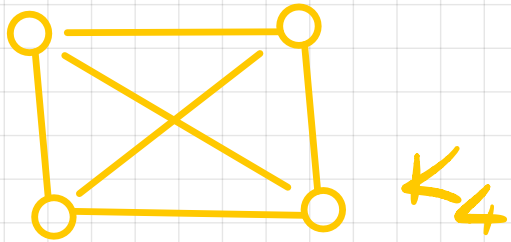
By strong induction hypothesis, each G_i has $(k_i - 1)$ edges

So G has $\sum_{i=1}^{\ell}(k_i - 1) + \ell = \sum_{i=1}^{\ell} k_i = k$ □

The Complete Graph

The complete graph on n vertices, K_n, is the graph that contains all possible edges

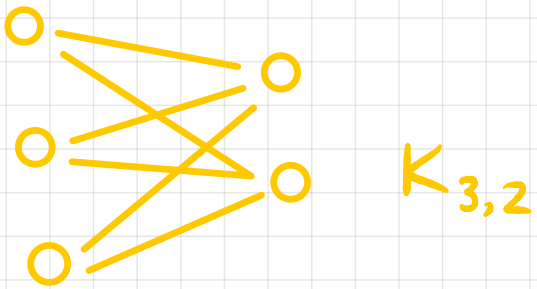
(so # of edges is)



Notes:

1. K_n is unique but $\exists$ many (n^{n-2}) trees on n vertices
2. K_n is maximally connected (need to remove at least $n-1$ edges to disconnect); trees are minimally connected (removing any edge disconnects)
3. Complete bipartite graph $K_{n,m}$:

$$\#\text{ of edges} =$$



Summary

- Graphs: directed & undirected
- Paths, cycles, walks, tours
- Eulerian tours
- Trees, complete graphs

Next lecture

- Planar graphs
- Hypercubes & connectivity