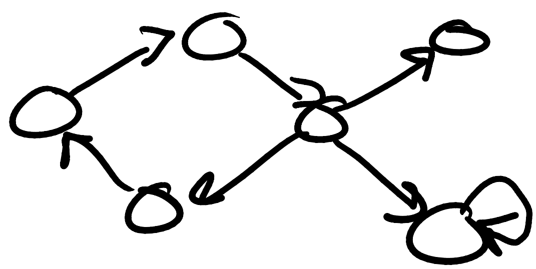
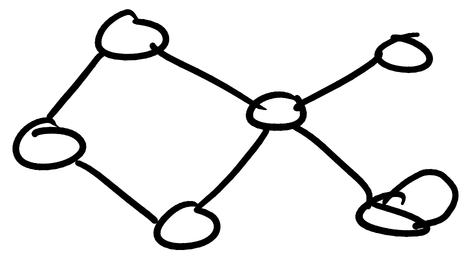
Recall the <u>underlying</u> graph of digraph D



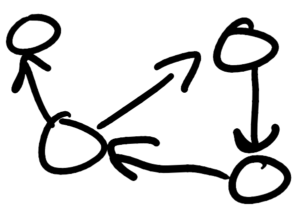
$\Rightarrow$



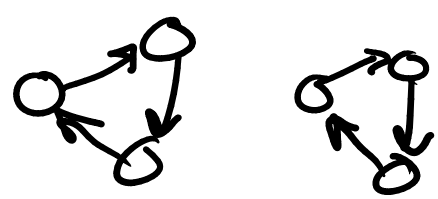
D

underlying graph of D

D is <u>weakly connected</u> if the underlying graph of D is connected



weakly connected

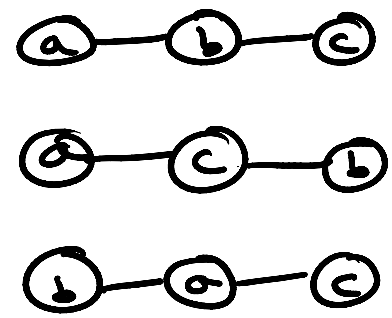


not weakly connected

Cayley's formula: there exists $n^{(n-2)}$ possible trees for $n = |V(T)|$

$n = 2 \Rightarrow 2^{2-2} = 1$

$n = 3 \Rightarrow 3^{3-2} = 3$



Prüfer code: a sequence of labels for tree T s.t. the length of the sequence is $n-2$ and the sequence is comprised of T's vertex labels

$A = \{a_1, a_2 \ldots a_{n-2}\}$

$S = \{\text{vertex labels of } T\}$

$a_i \in S$ sortable

Q: How do we construct T given prüfer code A?

CreatePrüfer(T):

$A = \emptyset$

$A = \emptyset$

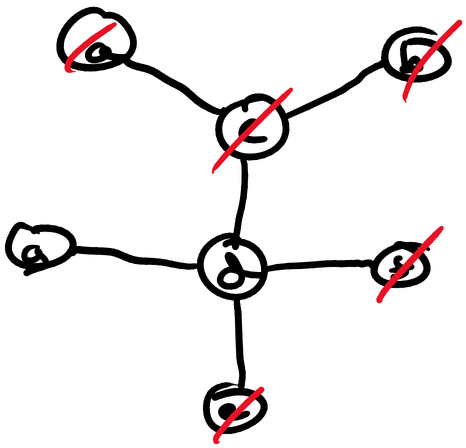
for $i = 1 \dots (n-2)$

  ℓ = label of the least remaining leaf

  $T = T - \ell$

  a_i = remaining neighbor of ℓ

  $A \leftarrow a_i$



$S = \{\cancel{a}\ \cancel{b}\ \cancel{c}\ d\ \cancel{e}\ \cancel{f}\ g\}$

$A = \{c\ c\ d\ d\ d\}$

Q2: Given a prüfer code, how do we construct a tree?

CreateTree(A, S):

  $V(T) = S$

  $E(T) = \emptyset$ ← empty set

  consider all s as "unmarked"

  for $i = 1 \dots (n-2)$

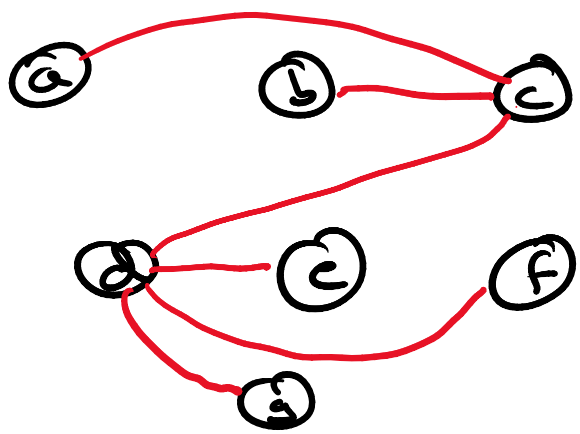
    x = least unmarked vertex in S

x = least unmarked vertex in S
that is not in $a_i \dots a_{n-2}$
mark x in S
$E(T) \leftarrow (x, a_i)$

(x, y) = remaining unmarked in S
$E(T) \leftarrow (x, y)$

$A = \{c\ c\ d\ d\ d\}$

$S = \{\cancel{a}\ \cancel{b}\ c\ \underline{d}\ \cancel{e}\ \cancel{f}\ \underline{g}\}$



Take away → for a given tree T and
vertex set S, we have a unique
code A
and given code A and set S,
we can construct a unique T

$f(T) = A$

$$f(T) = A$$

bijection

Q3: How's this relate to the enumerative properties of trees / Cayley's formula?

A3: There are $n^{(n-2)}$ ways to write a unique prüfer code.

$$A = \{a_i \ldots a_{n-2}\}$$

$a \in 1 \ldots n$ values

Proofer code prüf.

aka prooving Cayley's

What we're proving: existence and uniqueness of the mapping between trees and prüfer codes

We'll do strong induction on $n = |S|$

$S = \{a, b\}$

Base $P(2)$: (a) — (b)

Basis $P(2)$: $a - b$

$S = \{a, b\}$

$f(T) = A = \{\}$

$2^{2-2} = 2^0 = 1$ ✓

Consider $P(n) = T$

- Tree T with $V(T) = S$, $|S| = n$
- consider x as least value in S where x is a leaf in T
- consider a as neighbor of x

Construct $P(k) = T' = T - x$

$S' = S - x$

via I.H.: $A' = \{a_2 \ldots a_{n-2}\}$

$\rightarrow A'$ exists and is unique for a given (S', T')

Going from $P(k) \rightarrow P(n)$

$T' \rightarrow T$, we add back leaf x

$S' \rightarrow S$, we add x back to S

$A' \rightarrow A$

$\Rightarrow$ From our Prüfer code

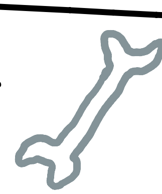
⇒ From our Prüfer code algorithm, the vertex x and edge (x, a) would be first selected for removal, so first value in A is guaranteed to be a (neighbor of ~~X~~)

$A = \{a, \{A'\}\}$ $f(T) = A$

↪ A exists for our T, S and is unique

⇒ it follows that Cayley's holds $\square$

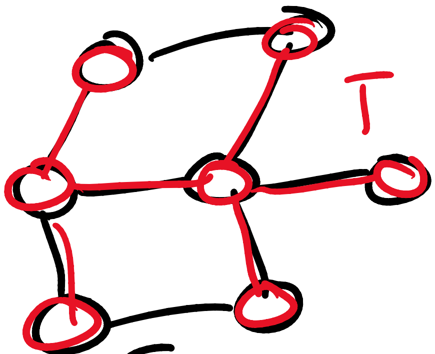
Spanning Trees



(spanner)

Spanning subgraph S of G is a subgraph s.t. $V(S) = V(G)$

Spanning tree is a spanning subgraph that is a tree subgraph that is a tree



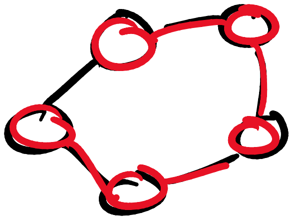
$\tau(G)$: # spanning trees on G

$\tau(T) = 1$

$\tau(P_n) = 1$

$\tau(C_n) = n$

$\tau(K_n) = n^{n-2}$

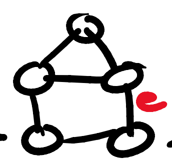
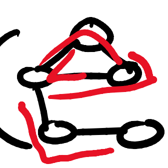
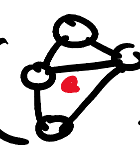


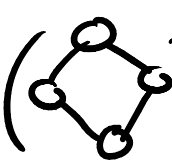
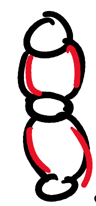
Q4: How can we count spanning trees on general graphs?

A: we can define a recurrence:

$$\underbrace{\tau(G)}_{\#\,\text{STs}} = \underbrace{\tau(G-e)}_{\#\,\text{ST w/o } e} + \underbrace{\tau(G\cdot e)}_{\#\,\text{ST with } e}$$

edge contraction

$\tau($  $) = \tau($  $) + \tau($  $)$

$3 + \tau($  $) + \tau($  $)$

$3 + 4 + 4 = 11$

$3 + 4 + 4 = 11$

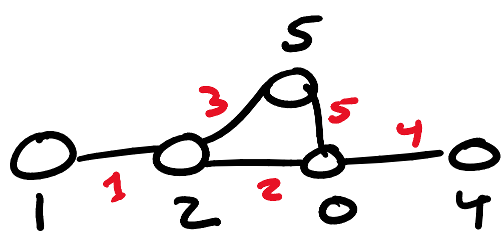
Graceful graphs

↳ graphs with a graceful labeling

Graceful labeling: a labels of vertices and edges of some G s.t.

$\forall v \in V(G): L(v) = 0 \dots m = |E(G)|$ and is unique

$\forall e \in E(G): e = (u,v): L(e) = |L(u) - L(v)|$ and is unique



graceful ✓

Graceful Tree Conjecture: (Ringel-Kotzig)

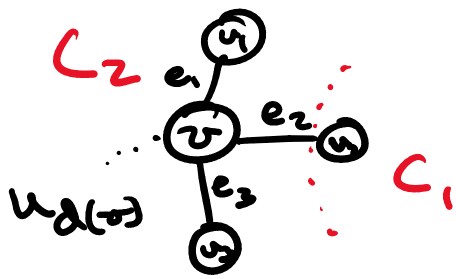
All trees are graceful (unproven)

Quiz4

$\forall v \in V(T): d(v) = \text{odd}$

$\iff T - e$ results in two odd components $\forall e \in E(T)$

$(\Rightarrow)$



Consider any cut e

Note: $d(u)$ in the new comp. is now even

Hint: degree sum formula and parity

Same for C_2

$(\Leftarrow)$ Consider the same above configuration around v

Note: $|V(C_1)|$ is odd

Also Note: any $|V(C_i)|$ is odd for any cut e_i around vertex v