SLPT: If you're good at something, people will generally let you know.

$\chi(G,k)$ = # of ways to properly color G given k colors

obviously,
if $\chi(G) > k \Rightarrow \chi(G,k) = 0$

consider clique K_n and some k

Q: How do we determine $\chi(K_n,k)$?

A: We can easily reason our way through it

First, color any $v \in V(K_n)$ with one of k possible colors

Next, color some other $u \in V(K_n)$ $u \neq v$
$\rightarrow$ we have $(k-1)$ possible choices

Next, color $w \in V(K_n)$
$u \neq w \neq v$

$\rightarrow$ we have $(k-2)$ options

... $(k-3)$

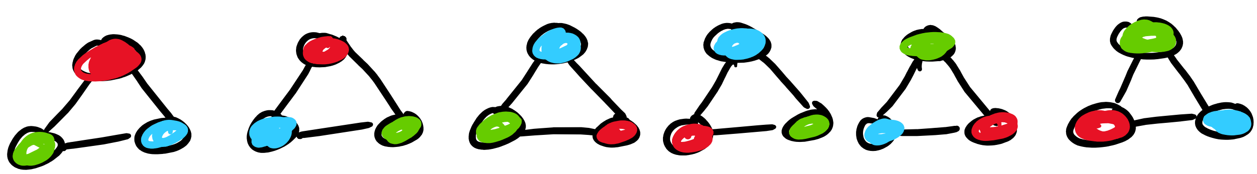
...

color final vertex with $(k-n+1)$ possible colors

$$\chi(K_n, k) = k(k-1)(k-2)\ldots(k-n+1)$$

consider K_3 and $k=3$ $\{$ $\}$

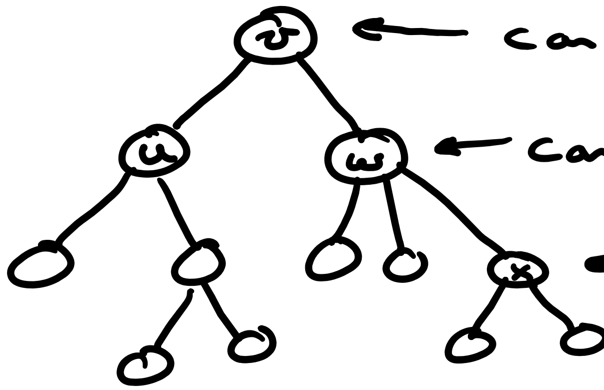
$$\chi(K_n, k) = k(k-1)(k-2)$$
$$= 3(2)(1) = 6$$



Let's talk trees

consider tree T and some BFS (breadth-first search) from some $v \in V(T)$

→ we'll create a BFS tree and consider $\chi(T, k)$



← can use k possible colors

← can use $(k-1)$ colors

← what about x? $(k-1)$ possibilities

As we color T level-by-level, we're only restricted by the single color assigned to a vertex's parent

=> all children have $(k-1)$ choices

$$\chi(T, k) = k(k-1)^{n-1}$$

Generally, we refer to $\chi(G, k)$ as the

Chromatic Polynomial

General form:

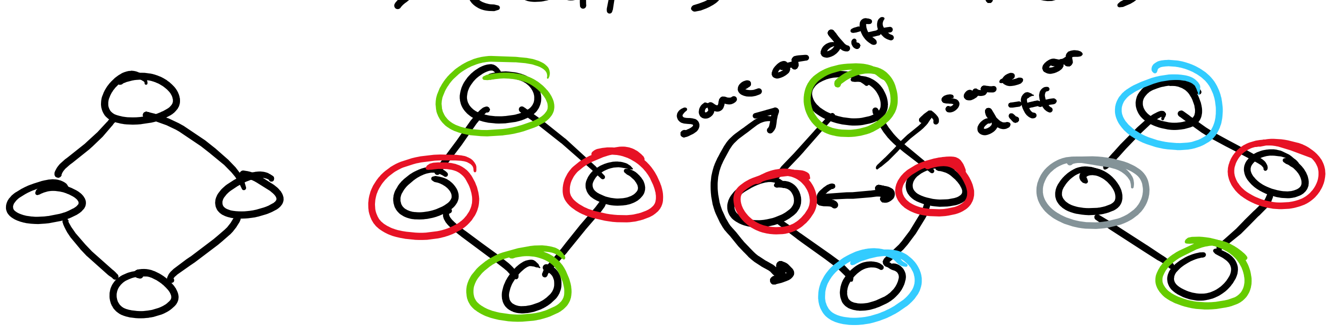
$$\chi(G,k) = \sum_{r=1}^{n} P_r(G)\, k_r$$

$P_r(G)$ = # of ways to partition $V(G)$ into r independent sets

k_r = # of ways to color r independent sets with k colors

$$k_r = k(k-1)(k-2)\ldots(k-r+1)$$

Consider C_4 and its chromatic poly. $\chi(C_4,k)$ and $P_r(G)$



$P_1(C_4)=0$ $\quad$ $P_2(C_4)=1$ $\quad$ $P_3(C_4)=2$ $\quad$ $P_4(C_4)=1$

$$\chi(C_4,k) = \sum_{r=1}^{n=4} P_r(C_4)\, k_r$$

$$= \underbrace{0}_{r=1} + \underbrace{1k(k-1)}_{r=2}$$

$$+ \underbrace{2k(k-1)(k-2)}_{r=3}$$

$$+ \underbrace{1k(k-1)(k-2)(k-3)}_{r=4}$$

$$\chi(C_4, k) = k(k-1) + 2k(k-1)(k-2)$$
$$+ k(k-1)(k-2)(k-3)$$

Note: we can use the chromatic polynomial to get $\chi(G)$

$$\chi(C_4, k=0) = 0$$

$$\chi(C_4, k=1) = 0 + 0 + 0$$

$$\chi(C_4, k=2) = 2(1) + 4(1)(0)$$
$$+ 2(1)(0)(-1)$$
$$= 2 \checkmark$$

$\hookrightarrow$ expected, as $\chi(C_4) = 2$

Q: Can we derive $\chi(G,k)$ in a simpler way?

A: Not really

↳ $P_r(G)$ is "tough" to compute

But → we can also determine $\chi(G,k)$ in a way that doesn't require $P_r(G)$

Fundamental Reduction Theorem

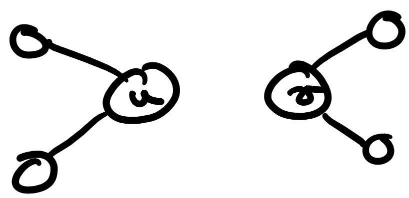
$$\chi(G,k) = \chi(G-e,k) - \chi(G \cdot e, k)$$

for any $e \in E(G)$

$e = (u,v)$

$\chi(G-e,k) = \#$ of ways to color G

$\chi(G-e,k) = \#$ of ways to color G



s.t. $c(u) = c(v)$

or $c(u) \neq c(v)$

(simple graphs)

$\chi(G \cdot e, k) = \#$ of ways to color G

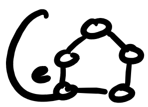
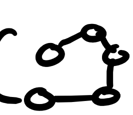
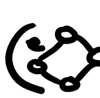
s.t. $c(u) = c(v)$

$$\chi(G,k) = \underbrace{\chi(G-e,k) - \chi(G \cdot e, k)}_{\substack{\#\text{ of ways to color } G \\ \text{s.t. } c(u) \neq c(v)}}$$

Recall: $\chi(K_n, k) = k(k-1)\dots(k-n+1)$

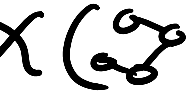
$\chi(T,k) = k(k-1)^{n-1}$

Consider C_5 and the above

$\chi($  $, k) = \chi($  $, k) - \chi($  $, k)$

tree

$= k(k-1)^4 - \Big[\chi($ $, k) - \chi($ $, k)\Big]$

$= k(k-1)^4 - \Big[\chi($  $, k) - \chi($  $, k) \Big]$

tree ✓ clique ✓

$= k(k-1)^4 - k(k-1)^3 + k(k-1)(k-2)$

Let's determine $\chi(C_5)$

$\chi(C_5, k=0) = 0$

$\chi(C_5, k=1) = 0$

$\chi(C_5, k=2) = 2(1)^4 - 2(1)^3 + 0$

$= 2 - 2 = 0$

$\chi(C_5, k=3) = 3(2)^4 - 3(2)^3 + 3(2)(1)$

$= 48 - 24 + 6$

$= 30$

Let's check  it out

→ C_5 has 5 vertices

→ C_5 is color-critical

↳ each [illegible] can be uniquely colored

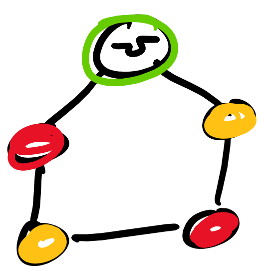
↳ each v can be uniquely colored with one of 3 colors

Algorithmically:

→ each of 5 vertices gets one of 3 possible colors

↳ 15 options

→ For each vertex v, $N(v)$ has the other colors



↳ gives us 2 ways to color $N(v)$

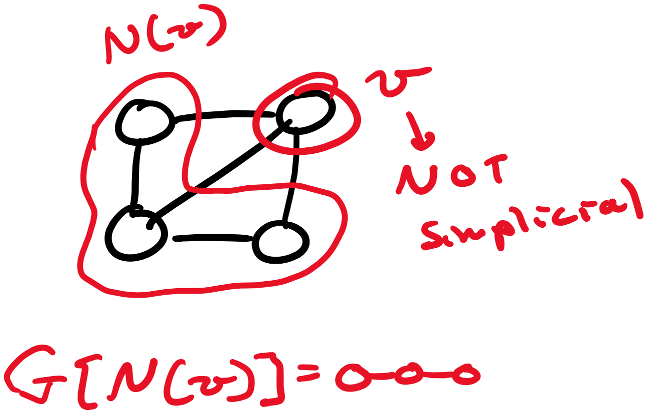
Together $\Rightarrow (15)(2) = 30$ ✓

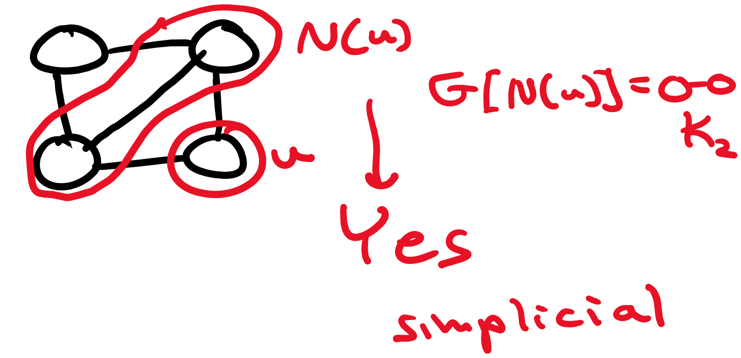
Simplicial vertices

A simplicial vertex is a vertex v where $N(v)$ is a clique where $N(v)$ is a clique

(induced subgraph) ↳ $G[N(v)]$

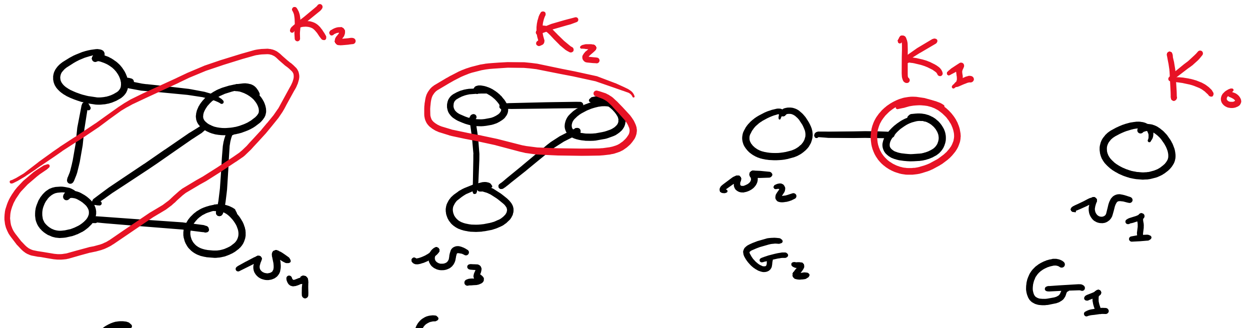
↓ includes $K_0\ K_1\ K_2$





simplicial elimination ordering (SEO)

an ordering $\{v_n v_{n-1} \dots v_1\}$ of all $v \in V(G)$ for deletion, s.t. each v_i is simplicial on induced subgraph $G[\{v_i v_{i-1} \dots v_1\}]$



$SEO = \{v_4 v_3 v_2 v_1\}$

Why do we care?

Why do we care?

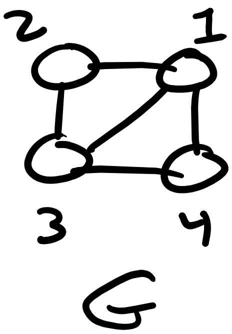
↪ we can use SEO to get $\chi(G,k)$

How?

↪ add $v_1 v_2 v_3 \dots v_i$ to $G_i[\{v_1 \dots v_i\}]$

$$\chi(G,k) = \prod_{i=1}^{n} (k - d'(v_i))$$

↑ degree of v_i in G_i



$d'(v_1) = 0$

$d'(v_2) = 1$

$d'(v_3) = 2$

$d'(v_4) = 2$

↪ $\chi(G,k) = k(k-1)(k-2)^2$

Why does this work?

→ same logic as before, we can guarantee some $k - d'(v_i)$ color choices, as $N(v_i)$ is a clique and therefore will have $|N(v_i)|$ unique colors have $|N(v_i)|$ unique colors

One more time, let's get $\chi($$)$

$\rightarrow \chi(G, k=0) = 0$

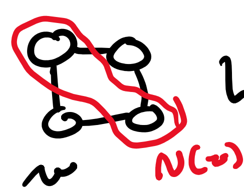
$\chi(G, k=1) = 0$

$\chi(G, k=2) = 0$

$\chi(G, k=3) = 3(2)(1)^2 = 6$

Note: $\omega(G) = 3 = \chi(G)$

recall: clique number

Note 2: 

has no <u>No</u> SEO

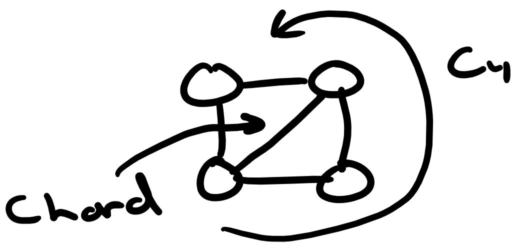
Q: what graphs have an SEO?

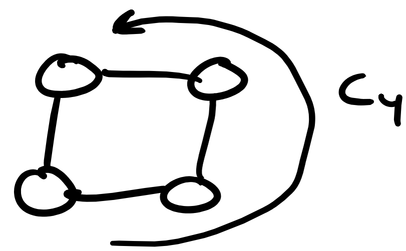
A: chordal graphs

<u>Chordal graph</u>: a simple graph with no chordless cycles

<u>chordless cycles</u>: a cycle of at least size 4 with no chords

<u>chord</u>: an edge with both endpoints on some cycle (but is not a part of that cycle)



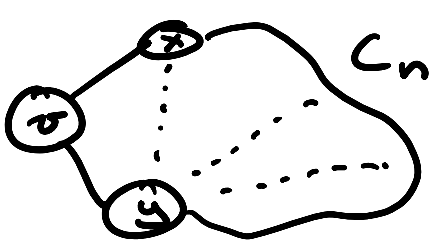


chordless cycle

G has an SEO $\Leftrightarrow$ G is chordal

($\Rightarrow$) Consider some $C_{n \geq 4} \subseteq G$

Consider the first $v \in V(C_n)$ eliminated in the SEO



Note: if $(x, y) \notin E(G)$, then v would not be simplicial

($\Leftarrow$) Show: every chordal graph has

($\Leftarrow$) Show: every chordal graph has a simplicial vertex

Note: deleting this vertex will not introduce a chordless cycle

Strong induction on $|V(G)|$

Basis $P(1) \to$ single vertex is simplicial

$P(n)$: We have G, $|V(G)| = n$

$\hookrightarrow$ consider $x \in V(G)$ and $G-x$

Case 1: $N(x) = \{V(G) - x\}$

$G-x$ is chordal

$\to$ I.H. on $G-x$ $\to$ SEO$(G-x)$

$\to$ any simplicial vertex on $G-x$ is simplicial on G

SEO$(G) = \{x\}$ + SEO$(G-x)$

Case 2: $N(x) \neq \{V(G) - x\}$

define $T = \{$vertices of maximum distance from $x\}$

distance from x}

define H = { subgraph induced on T → G[T] }

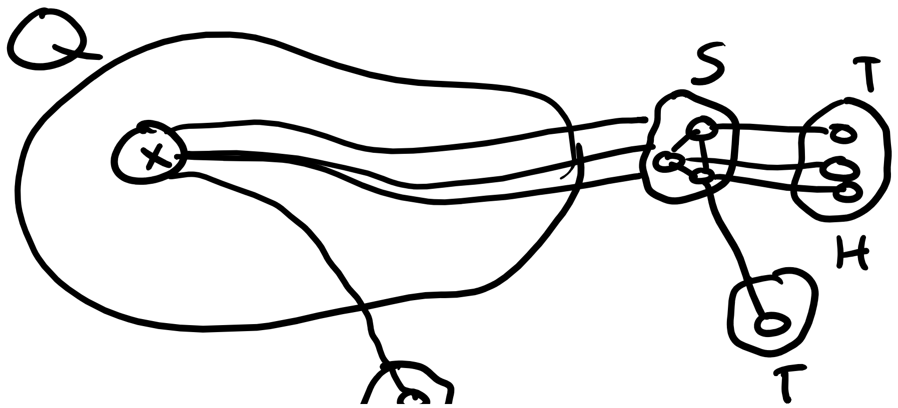
define S = { vertices in G − T with neighbors in V(H) ∪ T }

define Q = component of G − S that contains x

Note: S must be cliques (induced subgraph)

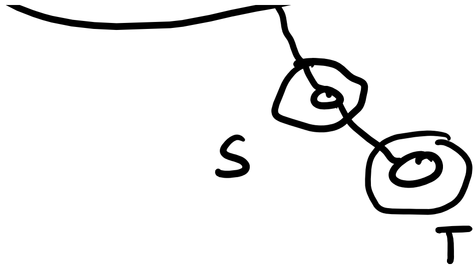
→ has neighbors in Q and H for all $w \in S$

→ any cycle from H → Q → H passing through $u, v \in S$, must have (u, v) as a chord



define $G' = G[S \cup V(H)]$

↳ I.H. on G'



$\to$ I.H. on G'

Consider some $w \in S$

$\to \exists$ simplicial vertex in $H \to z$, $z \in N(w)$

$\to$ also simplicial in G

$\Rightarrow$ so we can construct an SEO on G using this vertex

(apply same logic on $G-z$) $\square$

One more thing

G is perfect if

$$\chi(H) = \omega(H) \quad \forall H \subseteq G$$

Chordal graphs $\Rightarrow$ perfect

Exercise 4 reader