Announcements

- HW 5 Due Sunday
- Please let me know if you need to take exam 2 not-during class hours

Last Time: Colorings

<u>**Definition:**</u> A (vertex) *coloring* of a graph G is an assignment of a color to each vertex of G so that no two adjacent vertices have the same color. This is an *n-coloring* if only n different colors are used.

<u>**Definition:**</u> The *Chromatic Number*, $\chi(G)$, of a graph G is the smallest number n so that G has an n-coloring.

Today

- Basic facts about chromatic numbers
- Greedy colorings
- Brook’s Theorem

Question: Paths

What is $\chi(P_n)$?

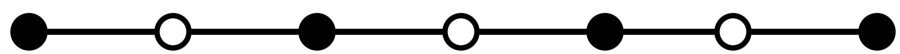
A) 1

B) 2

C) 3

D) 4

E) n



Actually, any tree is bipartite.

Question: Cycles

What is $\chi(C_n)$?

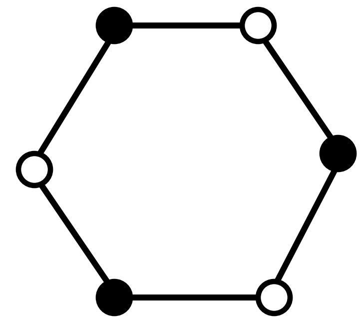
A) 1

B) 2

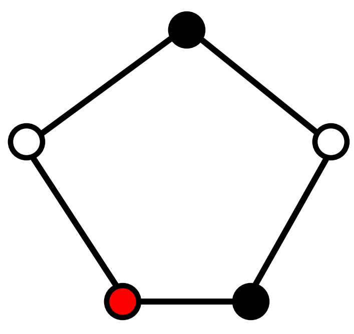
C) 3

D) 2 or 3 depending on n

E) n



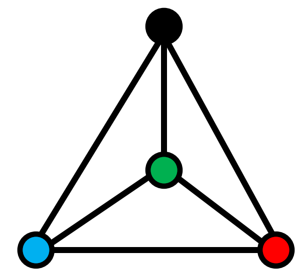
$\chi(C_n)$ is 2 for n even, 3 for n odd.



Question: Complete Graphs

What is $\chi(K_n)$?

A) 1
B) 2
C) 3
D) n-1
E) n



Each vertex must be a different color.

Cliques

<u>Definition:</u> The *clique number*, $\omega(G)$, of a graph G is the largest n so that K_n is a subgraph of G.

<u>Corollary:</u> For any graph G, $\chi(G) \geq \omega(G)$.

<u>Note:</u> This bound is far from tight.

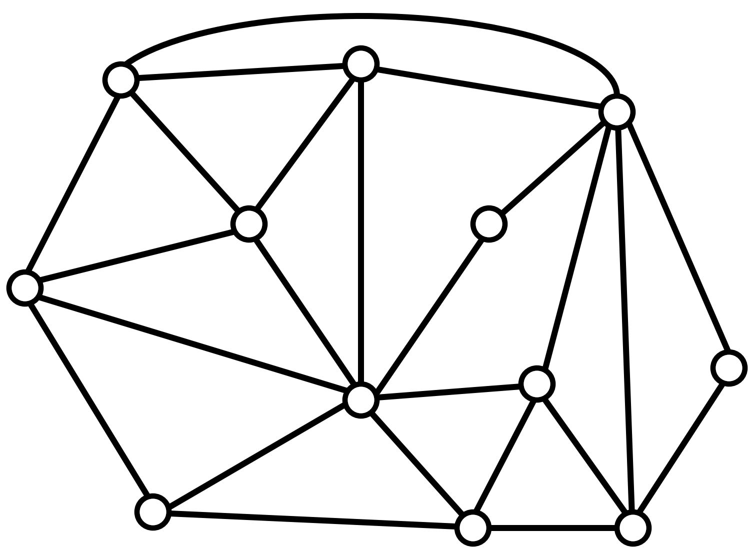
Upper Bound

Lemma: For G a graph on n vertices, then $\chi(G) \leq n$.

Proof: Give each vertex a different color.

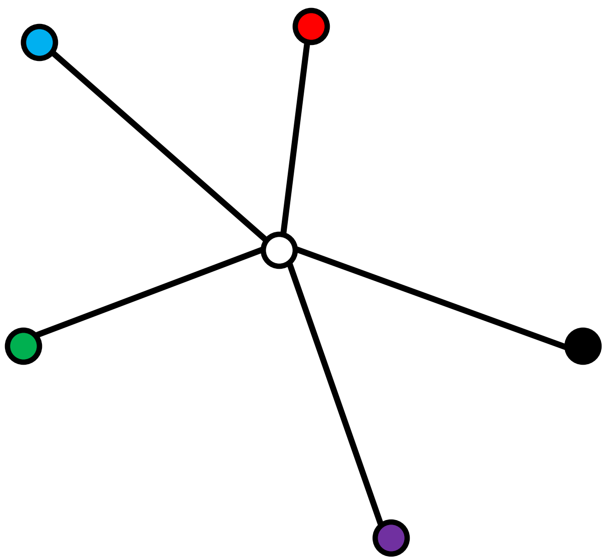
Greedy Coloring

Coloring Strategy: Color vertices one at a time, giving each a color that doesn't conflict.



Greedy Coloring II

If a vertex v has d(v) neighbors, it is enough to have d(v)+1 colors to choose from.



Max Degree

Definition: For a graph G, let $\Delta(G)$ denote the maximum degree of any vertex of G.

Lemma: For any graph G, $\chi(G) \leq \Delta(G)+1$.

Proof: Use the greedy coloring.

Note: This bound is again often far from tight.

e.g: $\chi(K_{n,n}) = 2$, but $\Delta(K_{n,n}) = n$.

Two Cases Where $\chi(G) = \Delta(G)+1$

C_n for n odd

- $\chi(G) = 3$
- $\Delta(G) = 2$

K_n for $n > 1$

- $\chi(G) = n$
- $\Delta(G) = n-1$

That’s it!

Theorem 1.43 (Brook’s Theorem): If G is a finite connected graph that is neither an odd cycle nor a complete graph,
$\chi(G) \leq \Delta(G)$.

$\Delta(G) \leq 2$

$\Delta(G) = 0$ or 1

- Only graph is complete graph.

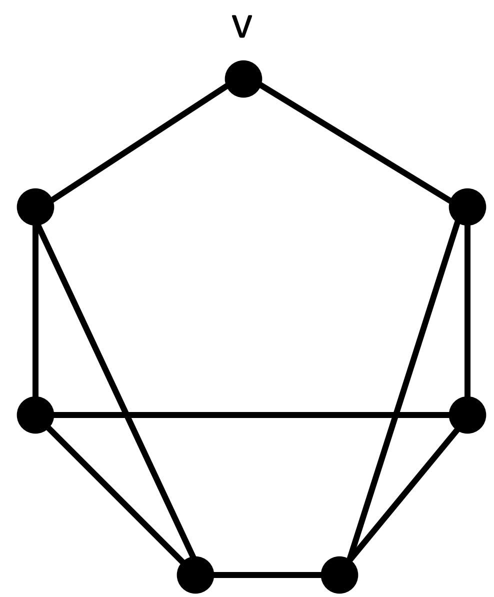
$\Delta(G) = 2$. Possibilities are:

- P_n: $\chi(G) = 2$
- C_n, n even: $\chi(G) = 2$
- C_n, n odd: Odd cycle

Non-Regular Graphs

If G not regular

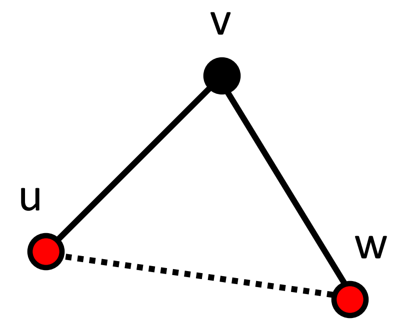
- Some v, $d(v) < \Delta(G)$
- If you can color G-v, greedily color v
- But v's neighbors in G-v have smaller degree
- Recurse



Regular Graphs

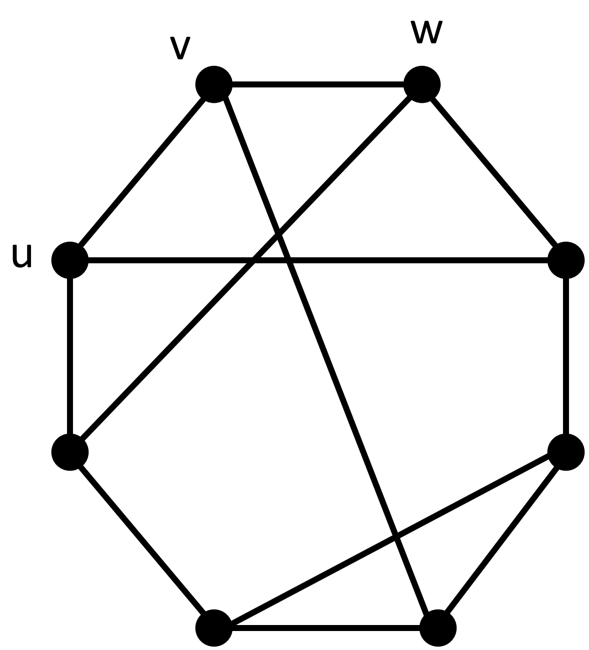
Idea: Want vertex v where two neighbors u, w have the same color.

- Find v with two non-adjacent neighbors (can unless G is complete graph)
- Try to color G-v with u, w the same color.



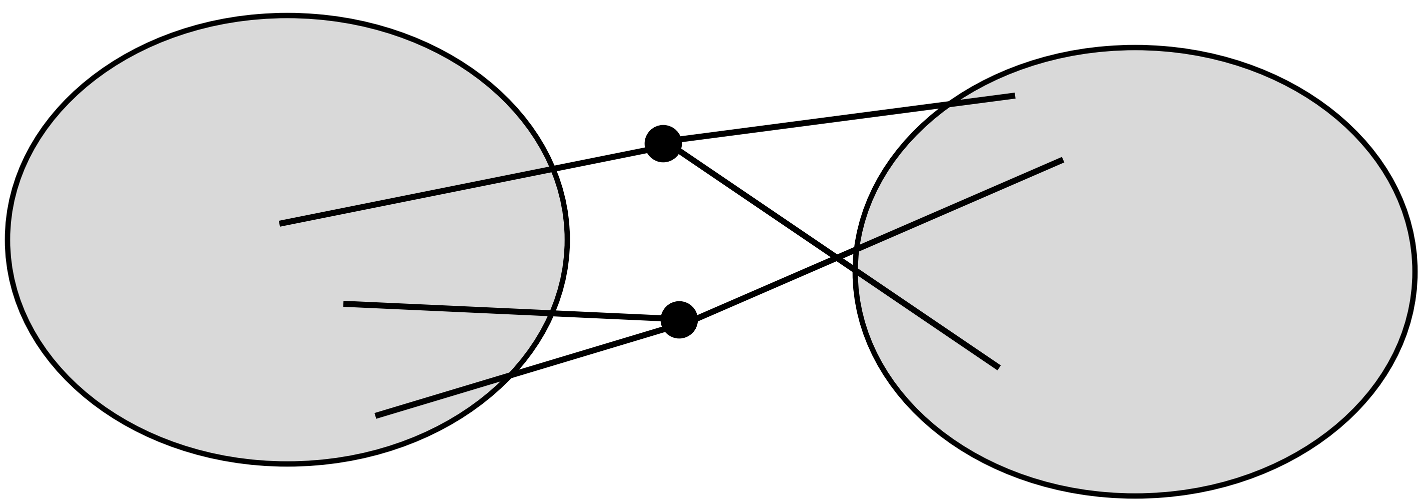
Not a Cut Set

If {u,w} is not a cut set, we can do our recursive coloring, assigning colors to u and w first.



What if {u,w} is a Cutset?

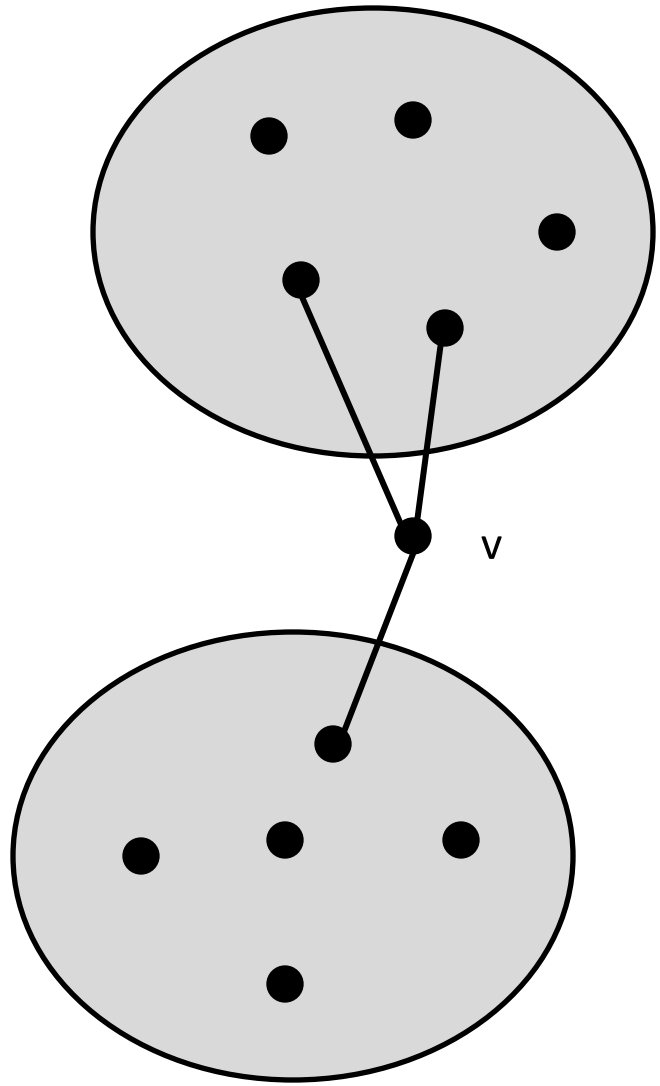
- Split graph into two parts
- Color each half inductively
- Change colors so that they match.



Case 1: Single Cut Vertex

Suppose v is a cut vertex.

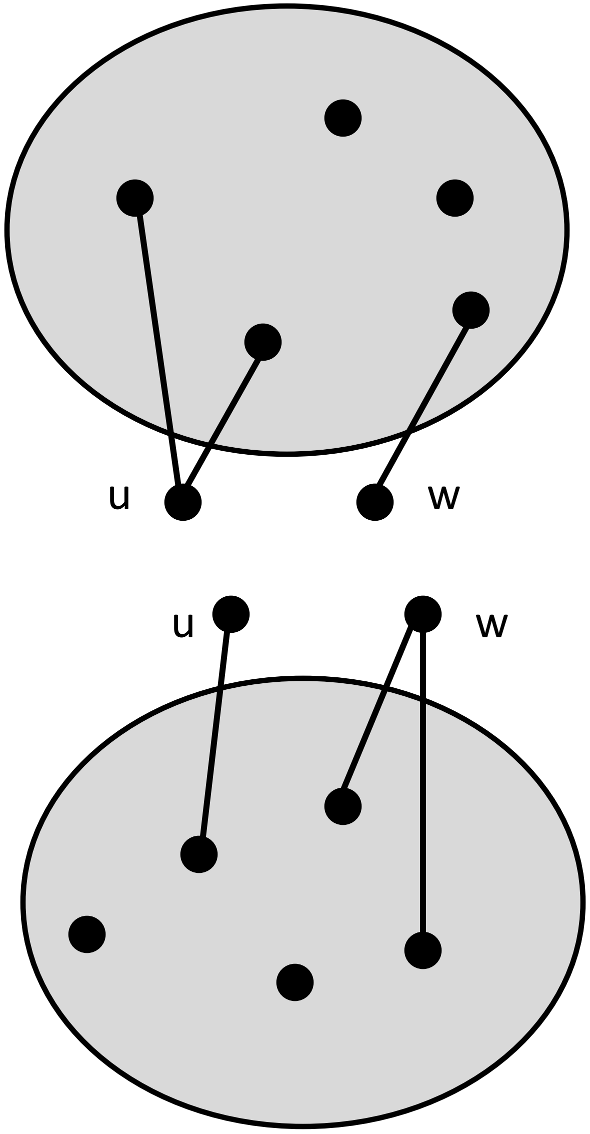
- Inductively color each component of G-v.
- For each component can change colors in that component.
- Arrange so two neighbors of v are same color
- Color v



Case 2: Two Cut Vertices

Idea:

- Color comp 1 + {u,w}
- Color comp 2 + {u,w}
- Swap colors so they match.
- Combine



Case 2a

Try to make u, w different colors on both sides.

- Use greedy coloring, choosing colors for u and w last.
- Unless they each connect to $\Delta(G)$-1 vertices on same side, can pick different colors.
- Make colors on top same as on bottom.

Case 2b

Suppose that u, w can only be the same in any coloring of top component.

- Must each connect to $\Delta(G)$-1 on top
- Only connect to 1 each on bottom.
- Color bottom component, can pick u, w same color (only 2 disallowed options).