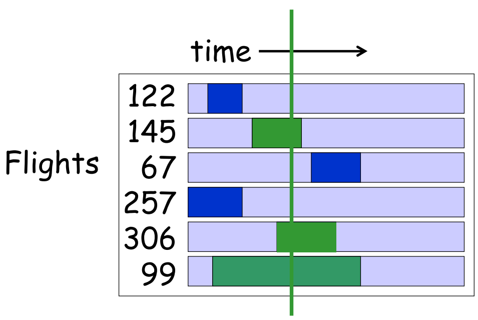
Graph Coloring

Flight Gates

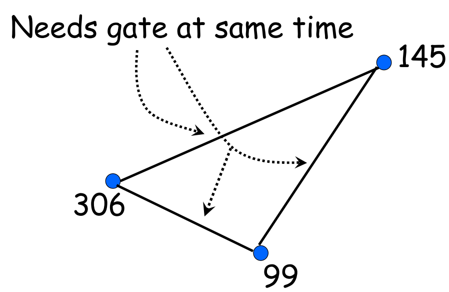
flights need gates, but times overlap.

how many gates needed?

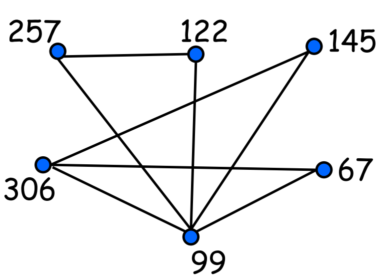
Airline Schedule



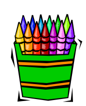
Conflicts Among Three



Model all Conflicts with a Graph



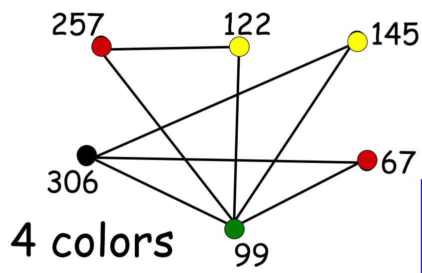
Color vertices



so adjacent vertices have different colors.

\# colors = # gates needed

Coloring the Vertices

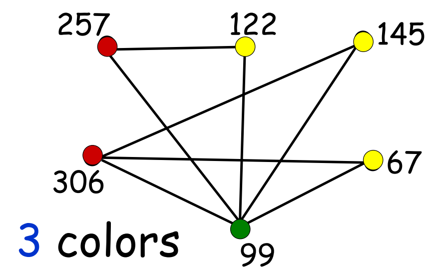


assign gates:

- 257, 67
- 122,145
- 99
- 306

4 colors
4 gates

Better coloring

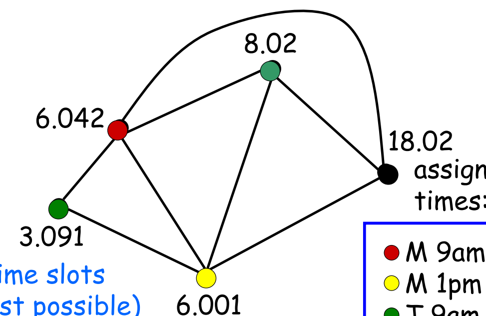


3 colors
3 gates

Final Exams

subjects conflict if student takes both, so
need different time slots.
how short an exam period?

Model as a Graph

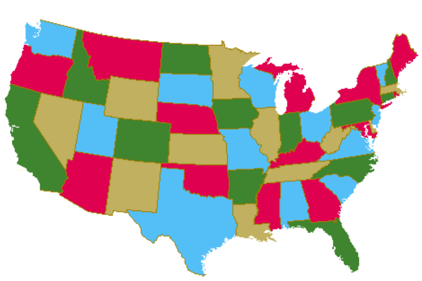


assign times:

- M 9am
- M 1pm
- T 9am
- T 1pm

4 time slots
(best possible)

Map Coloring



Four Color Theorem

any planar map is 4-colorable.
1850's: false proof published
(was correct for 5 colors).
1970's: prf with much computing
1990's: much improved

Chromatic Number

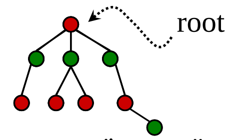
min #colors for G is

chromatic number, $\chi(G)$

lemma:

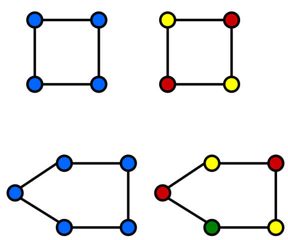
$$\boxed{\chi(tree) = 2}$$

Trees are 2-colorable



Pick any vertex as "root."
if (unique) path from root is
even length:
odd length:

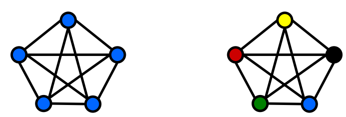
Simple Cycles



$$\chi(C_{even}) = 2$$

$$\chi(C_{odd}) = 3$$

Complete Graph K_5



$$\chi(K_n) = n$$

Bounded Degree

if all vertex degrees $\leq k$, then

$$\chi(G) \leq k+1$$

… by simple recursive coloring procedure

Coloring with d_{max} colors

Induction Hypothesis *P(n)* ::=
if G has n vertices, all degrees $\leq$ *dmax*,
then $\chi(G) \leq$ *dmax + 1* colors

Base Case: works for n =1 vertex

Inductive Step: given n +1 vertex graph

* remove one vertex
* color remaining graph in $\leq$ *dmax +1* colors
* put vertex back. since degree $\leq$ *dmax,* must be one color left over for it.

Arbitrary Graphs

2-colorable? --easy to check
3-colorable? --hard to check
(even if planar)
find $\chi(G)$? --theoretically no harder than 3-color, but harder in practice

Team Problems

Problems

1–3