Map Coloring

- Can we color each state with one of three colors, so that no two adjacent states have the same color?
Map Coloring and Graphs

- Place a vertex in each state/face.
Map Coloring and Graphs

Place a vertex in each state/face.
Place an edge between every pair of vertices that represent adjacent faces.

Map Coloring and Graphs

- The problem. Can we color the vertices using $k$ colors, such that every edge is adjacent to two different colors?
- Such a coloring is called a $k$-coloring.
Scheduling Exams

• **Problem.** We wish to set dates for the exams of every course in *Caltech.*
  ◦ Two exams cannot be on the same day if the classes have at least one student in common.
  ◦ How many exam days are necessary?

Solution

• **Solution.** Build a graph:
  ◦ A vertex for every class.
  ◦ An edge between every pair of classes with at least one common student.
  ◦ Find the minimum $k$ such that the graph is $k$-colorable. Every color corresponds to a different date.
The Case of Two Colors

- **Problem.** Given a graph $G = (V, E)$, check whether it has a 2-coloring.

Bipartite Graphs

- An undirected graph is **bipartite** if it admits a 2-coloring.
- We can partition the vertices of a bipartite graph into two sets, with every edge having one vertex in each set.
Problem: Prime Sums

- **Question.**
  - Let $G = (V, E)$ be an undirected graph with a vertex set $V = \{1, 2, \ldots, n\}$.
  - There is an edge between vertices $i$ and $j$ if and only if $i + j$ is prime.
  - Is $G$ bipartite?

- **Yes!** We can put every odd number on one side and every even number on the other.
Bipartite Graphs Characterization

• **Claim.** A graph \( G = (V, E) \) is bipartite if and only if it does not contain **cycles of odd length.**

Proof: One Direction

• **Assume** that \( G \) is bipartite and prove that \( G \) contains no odd-length cycles:
  ◦ Every edge connects the two sides of \( G \).
  ◦ A path that starts and ends in the same side must have an **even number of edges.**
  ◦ Any cycle must have an even number of edges.
Proof: The Other Direction

- Assume that $G$ contains no odd-length cycles and prove that $G$ is bipartite:
  - If $G$ is not connected, we prove the claim for each connected component separately. Thus, assume that $G$ is connected.
  - We prove the claim by describing an algorithm that finds a 2-coloring of $G$.

2-Coloring Algorithm

- Run the **BFS algorithm** from an arbitrary vertex $v$.
- Color the vertices of odd levels **red**, and vertices of even levels **blue**.
Correctness of the Algorithm

- **Prove.** No edge is *monochromatic*:
  - An edge of $G$ either connects vertices in consecutive levels of the BFS tree, or vertices in the same level.
  - An edge between consecutive levels connects a **blue vertex** and a **red vertex**.
  - It remains to prove that no edge connects two vertices from the same level.

Correctness of the Algorithm (2)

- **For contradiction**, assume that the edge $(u, v) \in E$ where $u, v \in V$ are in the same level of the BFS tree.
- Let $s$ be their **lowest common ancestor**.
- Let $P$ denote the path between $s$ and $u$. Let $Q$ denote the path between $s$ and $v$.
- If $P$ is of length $n$, so is $Q$.
- Connecting $P$, $Q$, and the edge $(u, v)$ yields a cycle of length $2n + 1$. **Contradiction!**
A More General Algorithm

- **Problem.** Change the previous algorithm so that it receives any graph \( G \).
  - If \( G \) is bipartite, output a 2-coloring.
  - Otherwise, output an error message.

Solution

- Change the BFS so that when it examines an edge, it checks whether both of its endpoints are on the same level:
  - **If we find such an edge, we stop the algorithm and output an error message.**
Example

Who’s suing who in the mobile business

Example (cont.)
The Four Color Theorem

- **Theorem.** Every map has a 4-coloring.
  - Asked over 150 years ago.
  - Over the decades several false proofs were published.
  - Proved in 1976 by Appel and Haken. Extremely complicated proof that relies on a computer program.

The Four Color Theorem

- **Question.** Does the four color theorem imply that every graph has a 4-coloring?
  - No! While every map corresponds to a graph, most graphs do not correspond to a map.
  - (Graphs that correspond to a map are called planar and can be drawn without edge crossings.)
Coloring Graphs with Bounded Degrees

- **Problem.** Show that any graph $G = (V, E)$ with every vertex of $V$ of degree at most $k$ admits a $(k + 1)$-coloring.

- **Proof.**
  - At each step choose an arbitrary uncolored vertex $v$.
  - Since $v$ has at most $k$ neighbors, one of the $k + 1$ colors must be OK for $v$.

**Example: $k + 1$-coloring**

```
1 2 3 4 5 6
```
Sometimes We Cannot Do Better

- $K_n$ - complete graph of $n$ vertices.
- Max degree: $n - 1$.
- Colors needed: $n$.

- $C_n$ - cycle of odd length $n$.
- Max degree: 2.
- Colors needed: 3.

Better Graph Coloring

- **Problem.** Show that if a graph $G = (V, E)$ satisfies:
  - Every vertex of $V$ has degree at most $k$.
  - $G$ is connected.
  - At least one vertex has degree $< k$.

  Then $G$ has a $k$-coloring.

For a proof check your solution of the third assignment.
3-Colorable Graphs

- **Problem.** Let $G = (V, E)$ be a graph that is 3-colorable, and let $n = |V|$.
  - Describe an efficient algorithm for coloring $G$ with $4\sqrt{n}$ colors.

- **Observation.** For any $v \in V$, the set of neighbors of $v$ can be colored using two colors.
  - Otherwise, we would need four colors to color $v$ and its neighbors.

Solution

- **Algorithm:**
  - As long as there is a vertex $v$ of degree at least $\sqrt{n}$, color $v$ with one new color and then color $v$’s neighbors with two other new colors. Then remove $v$ and its neighbors.
    - At each step we remove at least $\sqrt{n} + 1$ vertices, so there are less than $\sqrt{n}$ steps.
    - This step requires *less than* $3\sqrt{n}$ colors.
  - When all the remaining vertices have degree at most $\sqrt{n} - 1$, we know how to color the graph using *at most* $\sqrt{n}$ colors.
3-Colorable Graphs are Frustrating!

- It is **probably impossible** to color every 3-colorable graph in a reasonable time, using a constant number of colors.
- In 2007, Chlamtac presented an efficient algorithm for coloring using $cn^{0.2072}$ colors.
  - This algorithm is WAY TOO COMPLICATED for us to discuss.

The End: Three utilities problem