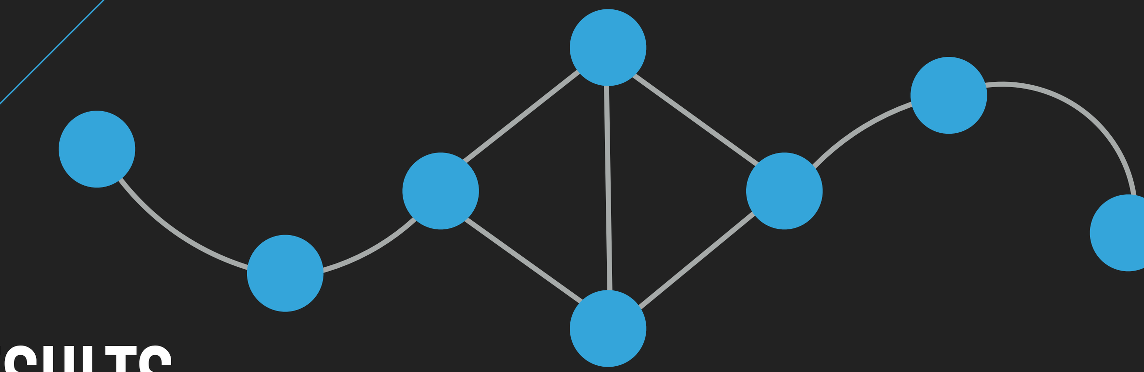
CONNECTIVITY OF COLORING GRAPHS

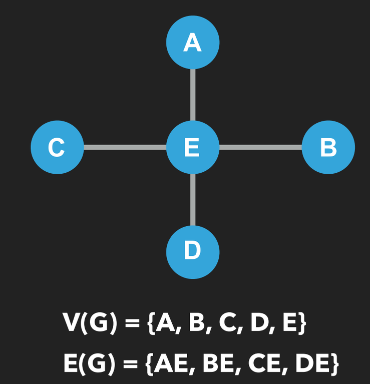
MAXINE XIN, INSTRUCTED BY DR. PRATEEK BHAKTA
DEPART. OF COMPUTER SCIENCE, UNIV. OF RICHMOND



TERMINOLOGIES

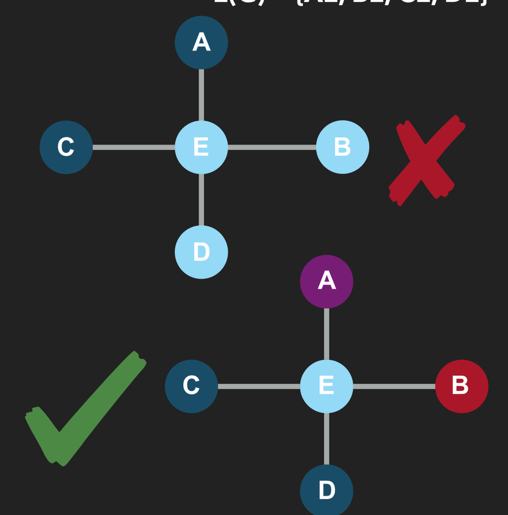
Graph

A graph G is a diagram composed of a set of **vertices**, V(G), and a set of **edges**, E(G), each one of which connects two neighboring vertices.



Coloring of a Graph

A coloring is a way to color the vertices of a graph such that every edge of the graph is incident to vertices of **different** colors

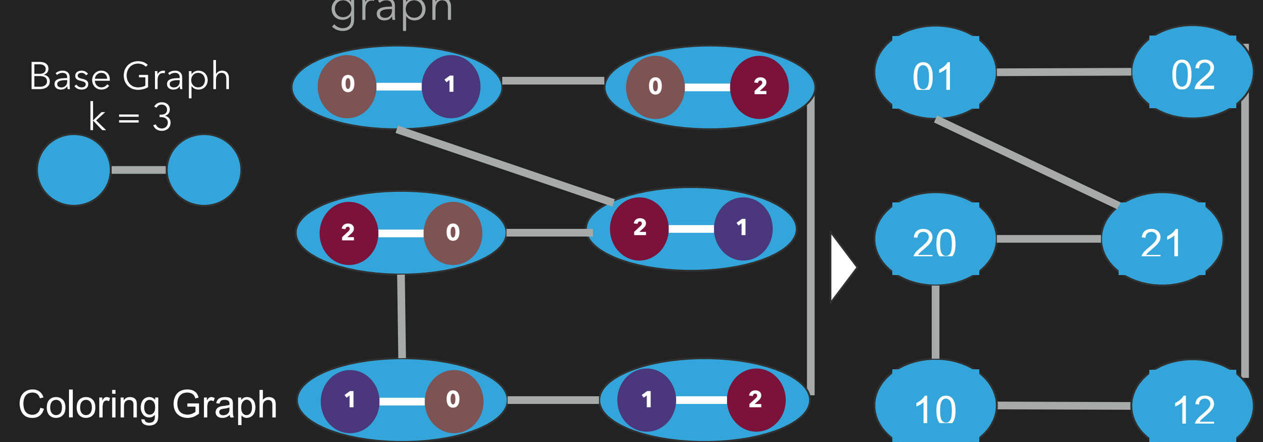


Coloring Graph

The k-coloring graph of G is composed of all of the k-colorings of G:

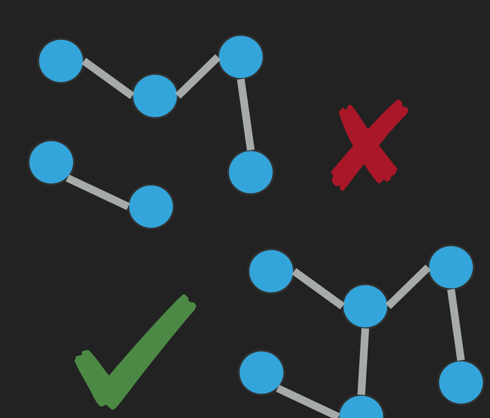
Given a base graph G, the coloring Graph $C_k(G)$ is composed of:

- $V(C_k(G))$: **All k-colorings** of G
- $E(C_k(G))$: **Connect** two k-colorings if they are **different at only one vertex** of the base graph



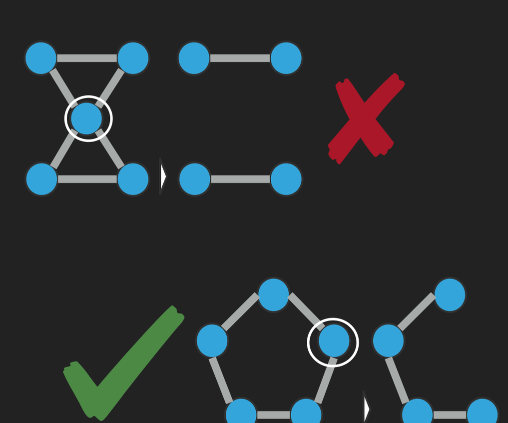
Connectivity: connected

A graph G is connected if there is **a path** between **every two vertices** of G.



Connectivity: 2-connected

A Graph G is 2 - connected if after **removing any one vertex v and its incident edges** from G, G - v is still connected.



RESEARCH PROCESS

Step 1

Generate the coloring graph of a base graph:

- Generate all the k - colorings of the base graph
- Store each colored base graph as a vertex of the coloring graph

ROADBLOCK1

Too many colorings

Too many graph objects

OUT OF MEMORY

SOLUTION 1

Substituted **colored base graph objects** with **long** type encodings of the coloring

Step 2

Program a method that outputs the connectivity of an input coloring graph

- Implement the **Tarjan algorithm**(an enhanced depth first search)

SOLUTION 2

Changed the **recursion** to **iteration**, while kept the function of Tarjan unchanged

ROADBLOCK2

Recursive algorithm

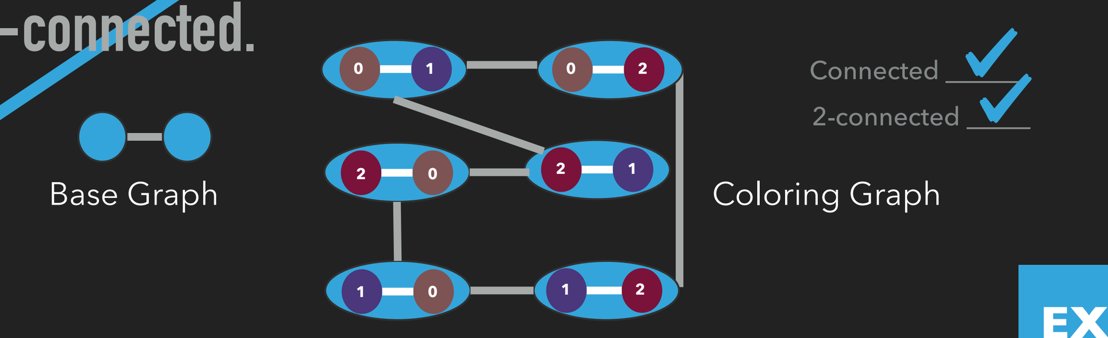
Too many CGraph vertices

RECURSIVE DEPTH LIMIT

GOAL:

Prove or disprove:

If $C_k(G)$, $C_{k+1}(G)$, $C_{k+2}(G)$, $C_{k+3}(G)$... are connected, then $C_{k+1}(G)$, $C_{k+2}(G)$, $C_{k+3}(G)$... are all 2-connected.



DISPROVED

ROADBLOCK4

No small counterexample exists

Hard to automatically generate counterexamples

Combined **mathematical intuition** with programming tools to manually generate a counterexample

SOLUTION 4

1. Include a **HashMap**
2. Combine HashMap with Tarjan

Able to explore the neighbors as the connectivity is being checked, so neighbor **(arraylist of CVertex objects) was removed**

SOLUTION 3

ROADBLOCK3

Each CVertex has a neighbor field

Neighbor: an ArrayList of CVertex objects

OUT OF MEMORY

Try searching for a counterexample:

- Find a base graph G such that $C_k(G)$ is connected and $C_{k+1}(G)$ is not 2-connected

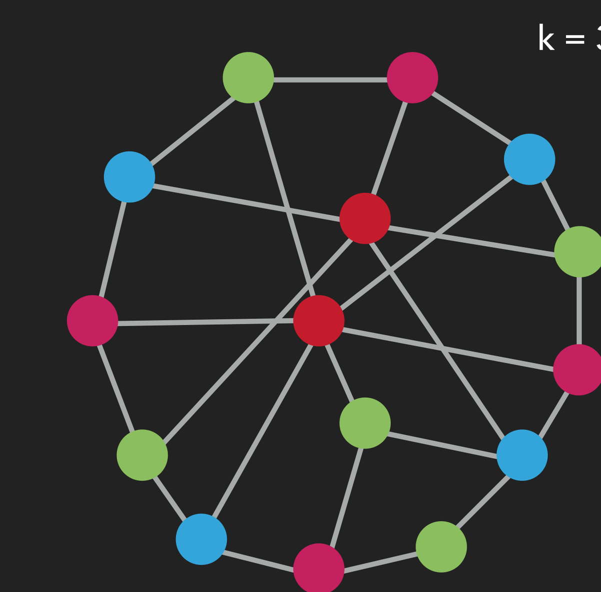
Step 3

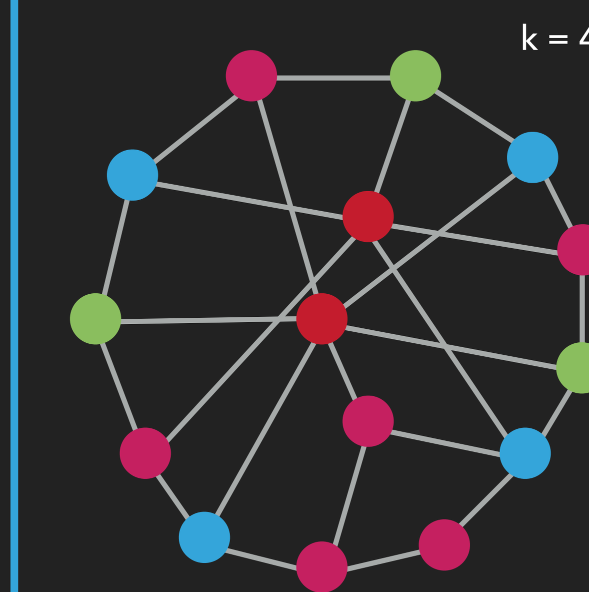
Step 2

RESEARCH PROCESS

RESULTS

Counterexample found!






```
tarjan starting
tarjan started, running…
size of cut vertices: 0
tarjan finished
This coloring graph is 2-
connected
```

```
tarjan starting
tarjan started, running…
size of cut vertices: 48
tarjan finished
```

$C_3(G)$ is 2-connected

$C_3(G)$ is connected

$C_4(G)$, $C_5(G)$, $C_6(G)$, ... are connected

BUT... >>

$C_{3+1}(G) = C_4(G)$ is not 2-connected

FUTURE DIRECTIONS

- If $C_k(G)$ is connected, does there exist any i, **i > 1**, such that $C_{k+i}(G)$ must be 2-connected?
- Does there exist **certain graphs** on which our conjecture is true? If so, what features do these graphs have?
- What kind of structures can exist in a **non 2-connected** coloring graph? How to manipulate the base graph to create the coloring graphs of certain structures?
- Considering the **limited color choices** in the offshoots of a non 2-connected coloring graph, is it possible that all such coloring graphs are **bipartite**?