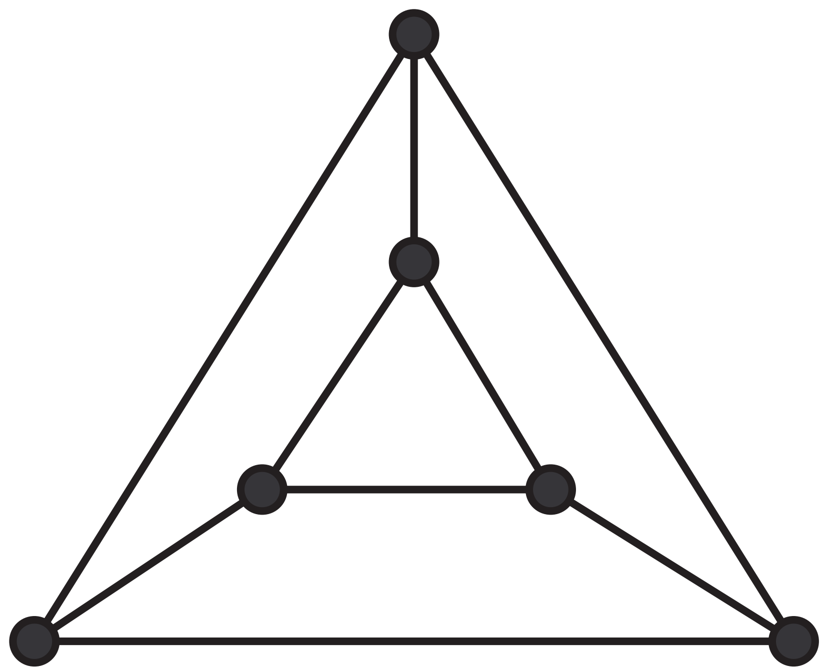
Graph Algorithms

Edge Coloring

The Input Graph

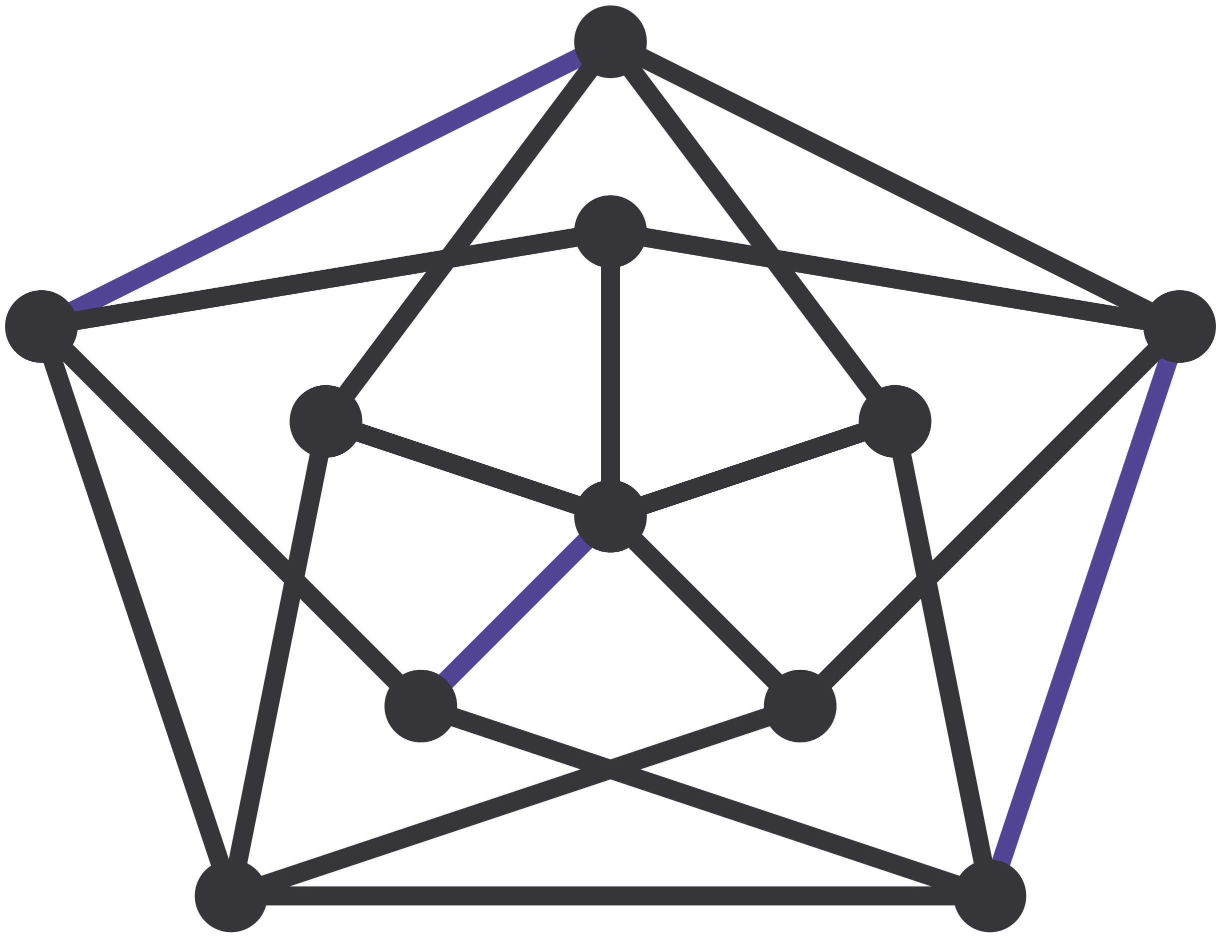
* A simple and undirected graph $G = (V, E)$ with n vertices in V, m edges in E, and maximum degree Δ.



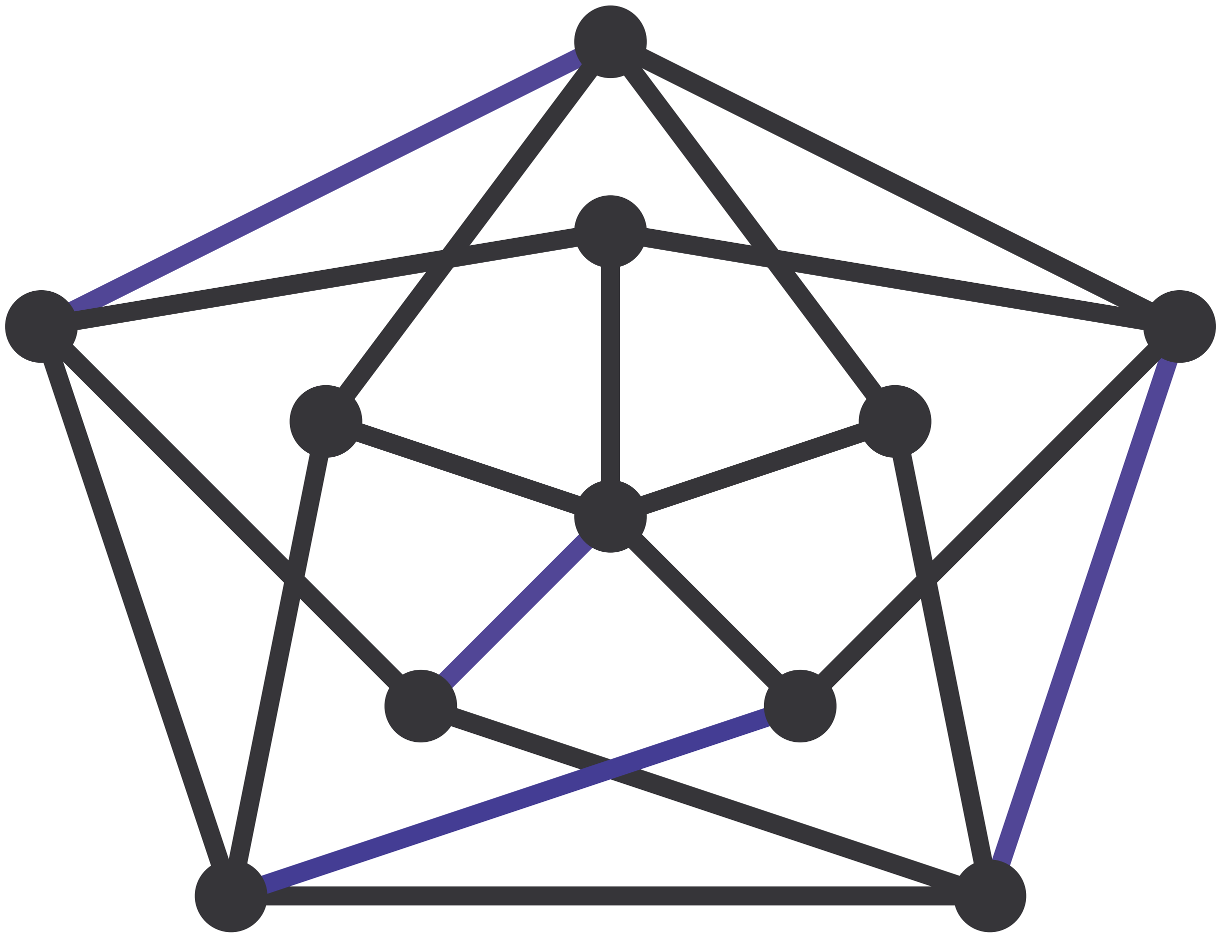
Matchings

- ⋆ A matching, $M \subseteq E$, is a set of edges such that any 2 edges from the set do not intersect.
  - $\forall_{(u,v)\neq(u',v')\in M}\,(u \neq u', u \neq v', v \neq u', v \neq v')$.
- ⋆ A perfect matching, $M \subseteq E$, is a matching that covers all the vertices.
  - $\forall_{u\in V}\exists\{(u,v) \in M\}$.

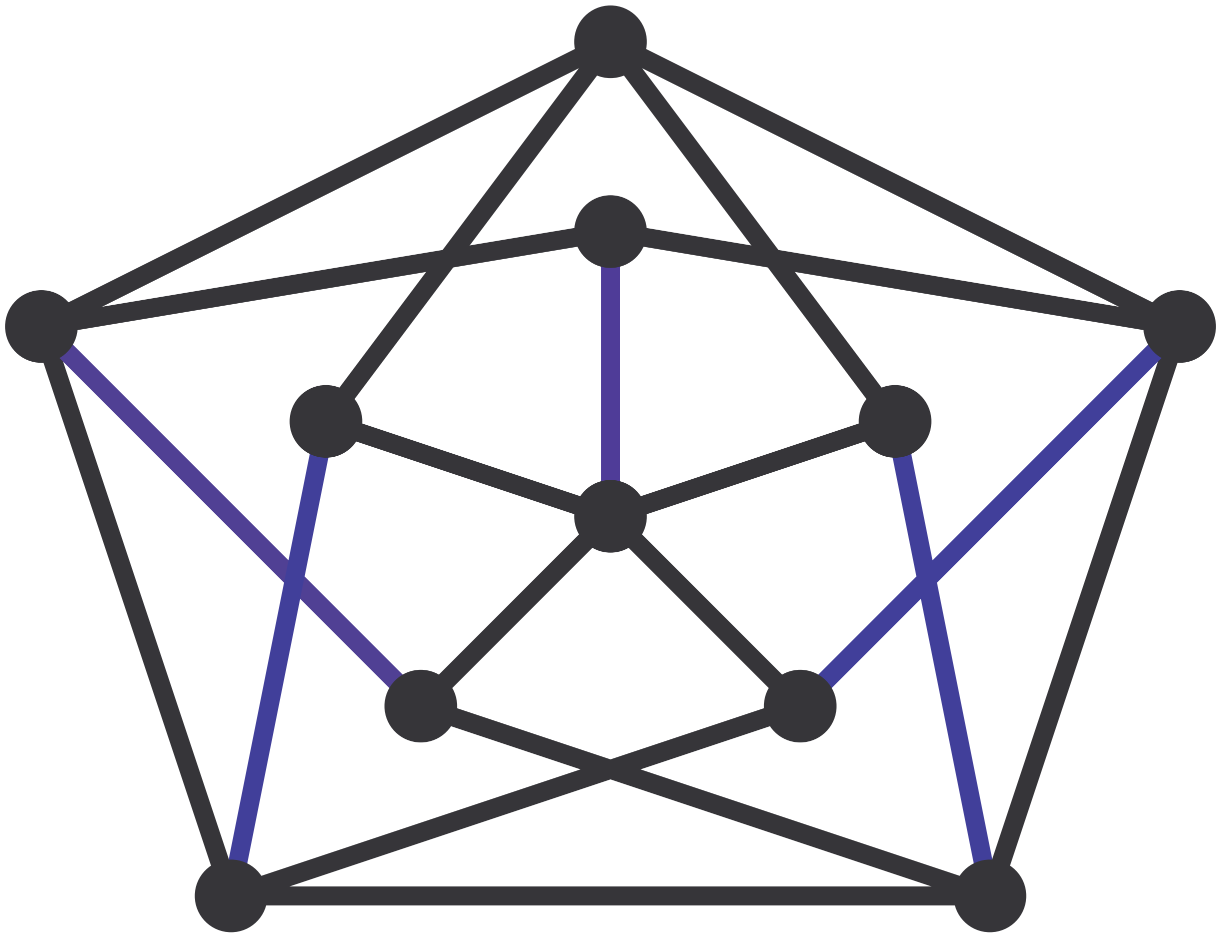
Example: Matching



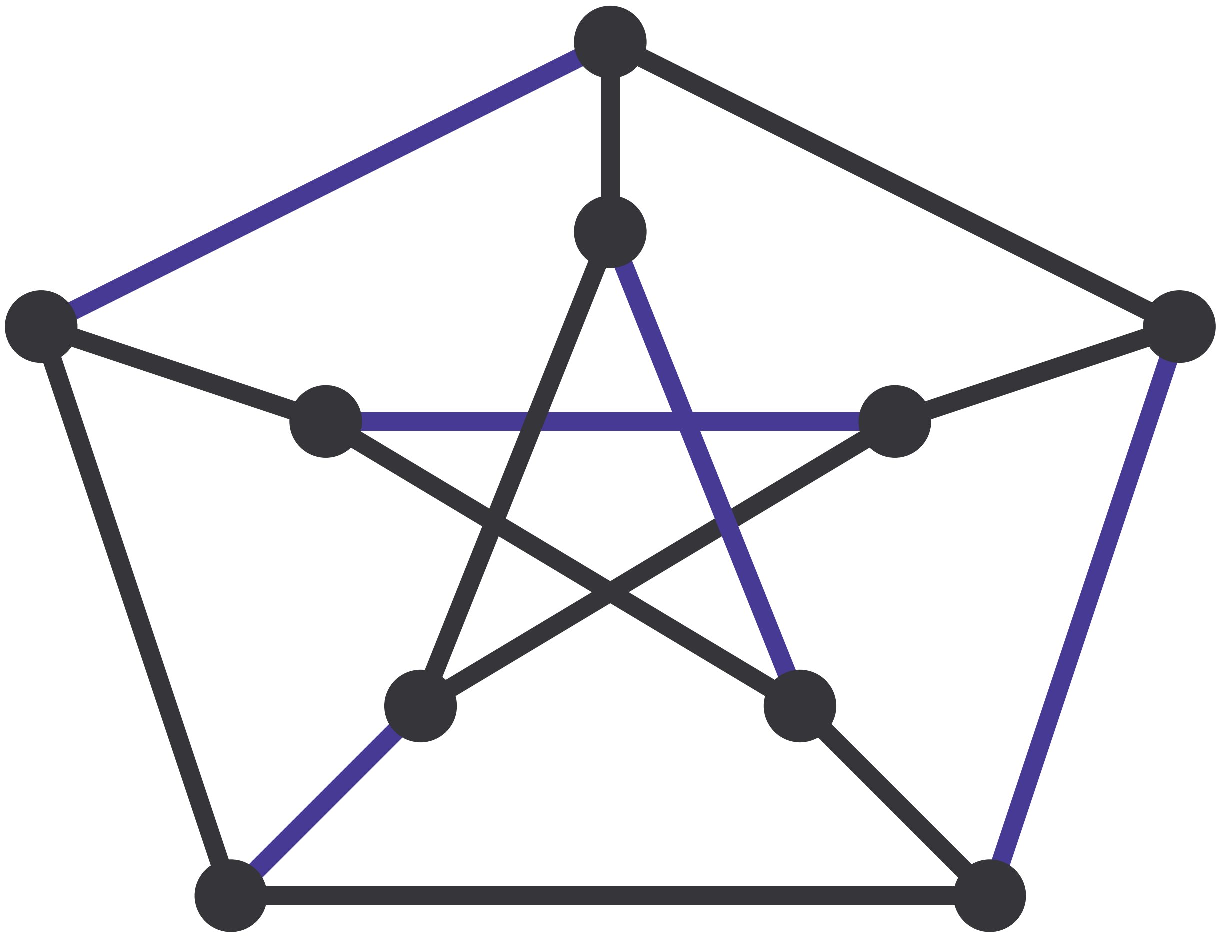
Example: Maximal Matching



Example: Maximum Size Matching



Example: Perfect Matching

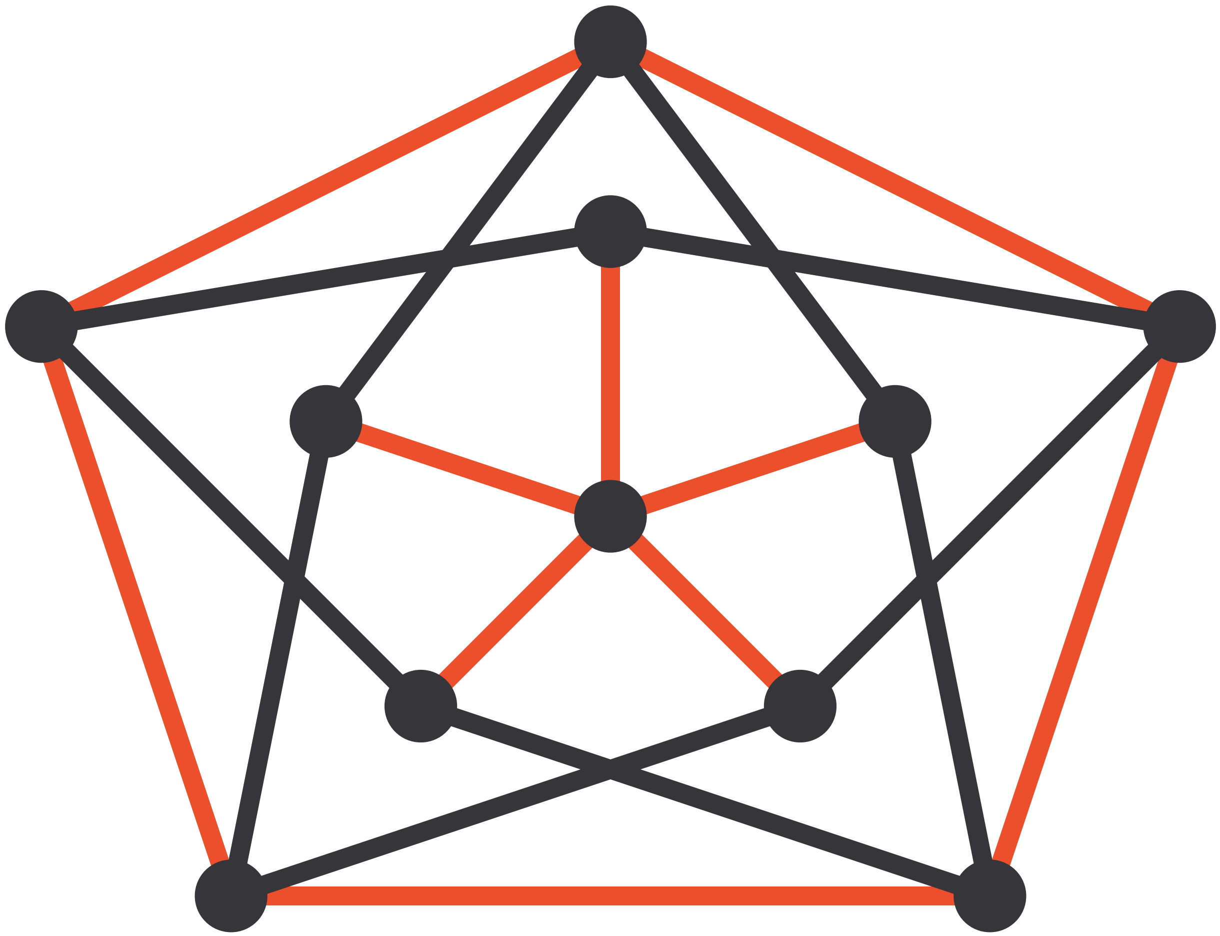


Edge Covering

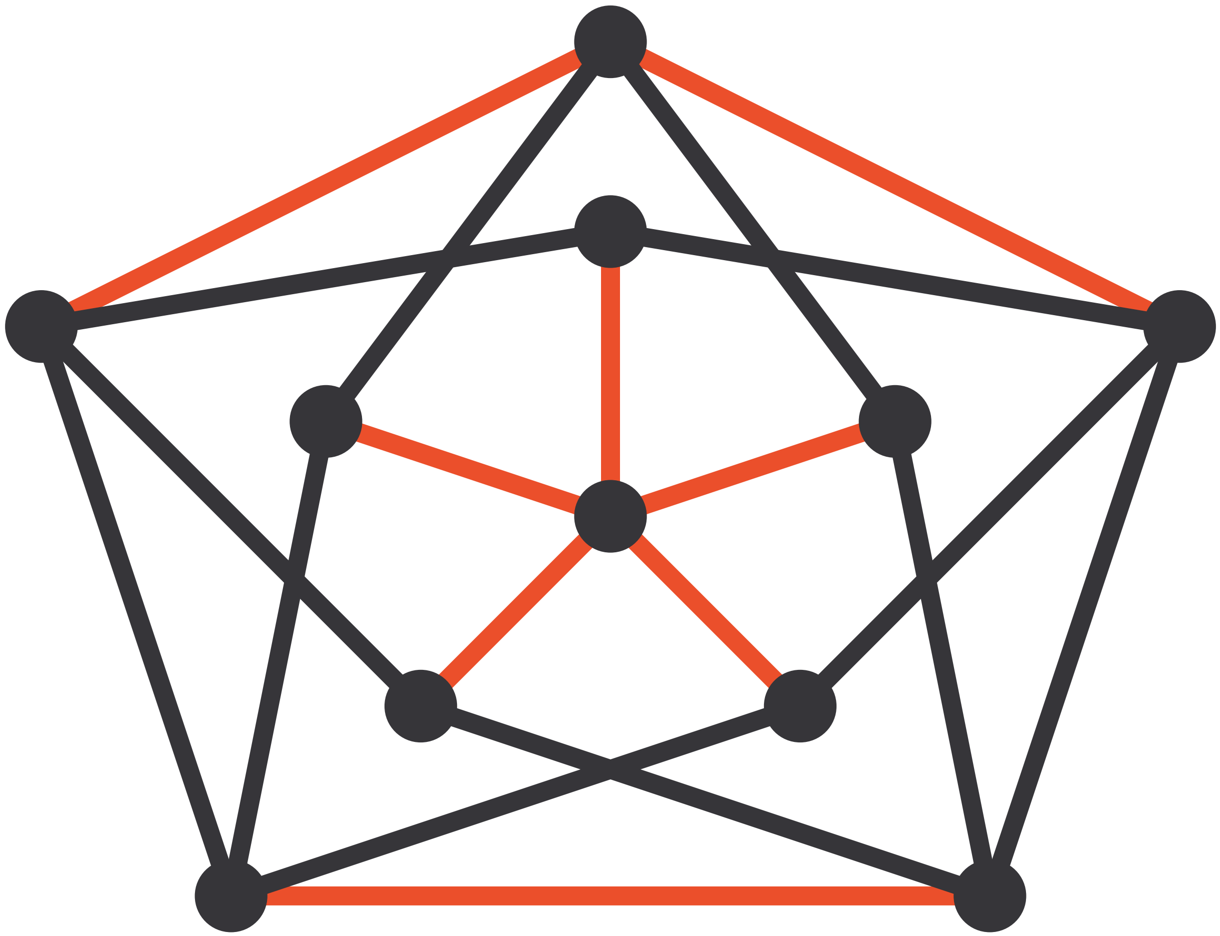
⋆ An edge covering, $EC \subseteq E$, is a set of edges such that any vertex in V belongs to at least one of the edges in EC.

– $\forall_{v \in V} \exists_{e \in EC} (e = (u, v))$.

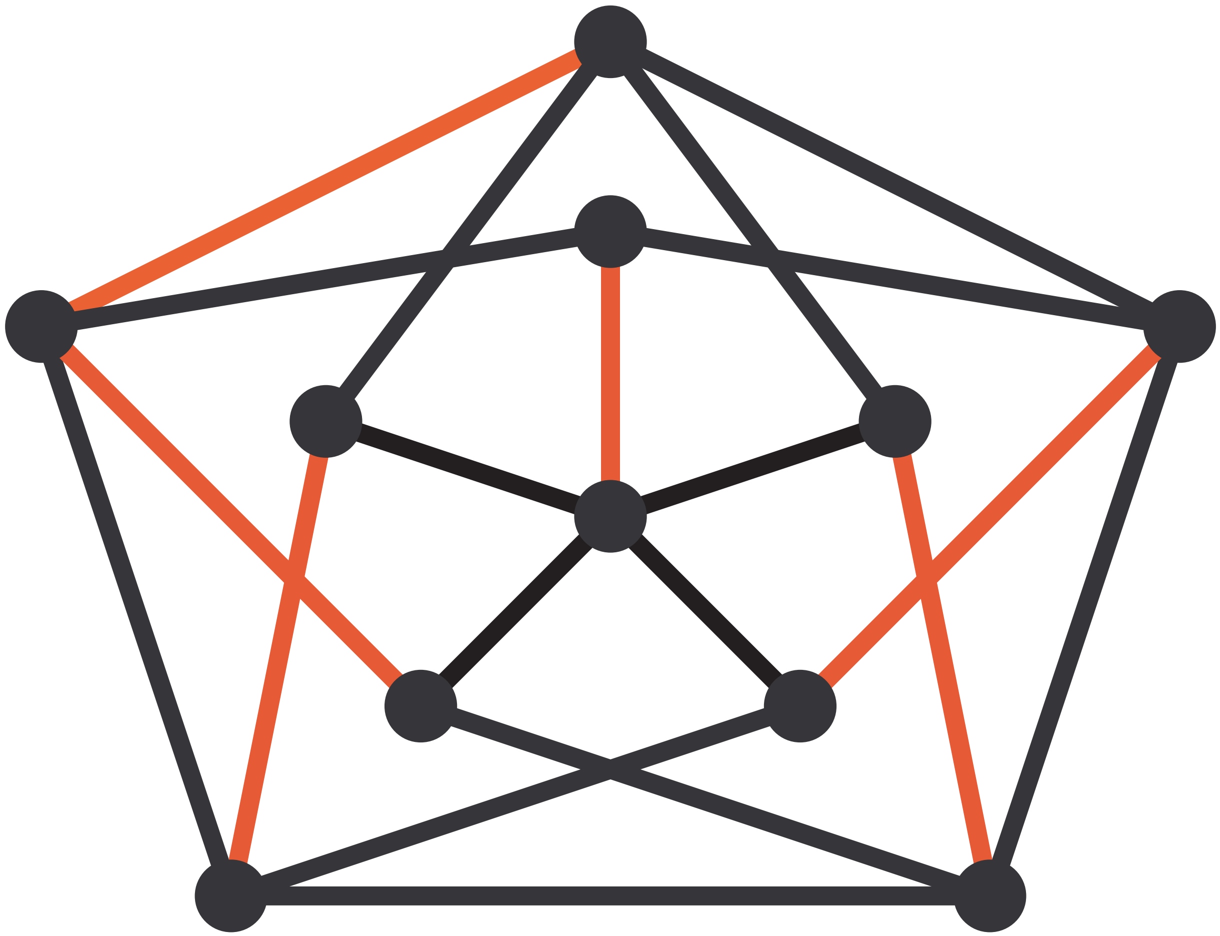
Example: Edge Covering



Example: Minimal Edge Covering



Example: Minimum Size Edge Covering



Matching and Edge Covering

Definition An isolated vertex is a vertex with no neighbors.

Proposition: $EC + M = n$ for G with no isolated vertices.

Proof outline: Show that $EC + M \geq n$ and $EC + M \leq n$ which imply $EC + M = n$.

$EC + M \geq n$

- ⋆ Construct a matching M'.
- ⋆ Consider the edges of EC in any order.
- ⋆ Add an edge to M' if it does not intersect any edge that is already in M'.
- ⋆ The rest of the edges in EC connect a vertex from M' to a vertex that is not in M'.
- ⋆ Therefore, the number of edges in EC is the number of edges in M' plus $n - 2M'$ additional edges.

$$\boxed{EC + M \geq n}$$

$$\begin{aligned}
EC &= M' + (n - 2M') \\
&= n - M' \\
&\geq n - M \qquad (* \text{ since } M \geq M' \ *) \\
&\Rightarrow EC + M \geq n \ .
\end{aligned}$$

$EC + M \leq n$

- Construct an edge covering EC'.
- EC' contains all the edges of the maximum matching M.
- For each vertex that is not covered by M, add an edge that contains it to EC'.
- Due to the maximality of M, there is no edge that covers 2 vertices that are not covered by M.
- Therefore, the number of edges in EC' is the number of edges in M plus $n - 2M$ additional edges.

$EC + M \leq n$

$$
\begin{aligned}
EC' &= M + (n - 2M) \\
&= n - M \\
&\Rightarrow EC \leq n - M \qquad (* \text{ since } EC' \geq EC *) \\
&\Rightarrow EC + M \leq n \; .
\end{aligned}
$$

Edge Coloring

Definition I:

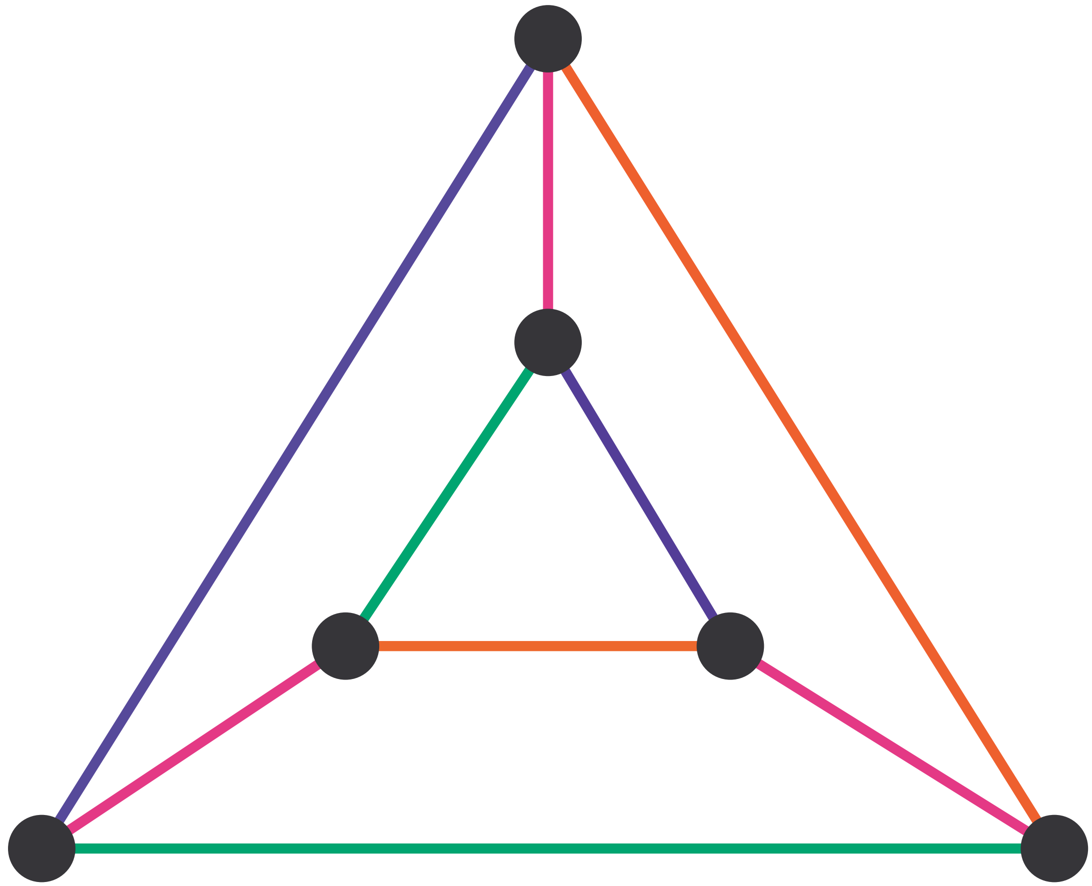
- ⋆ A disjoint collection of matchings that cover all the edges in the graph.
- ⋆ A partition $E = M_1 \cup\ M_2 \cup \cdots \cup M_\psi$ such that M_j is a matching for all $1 \leq j \leq \psi$.

Definition II:

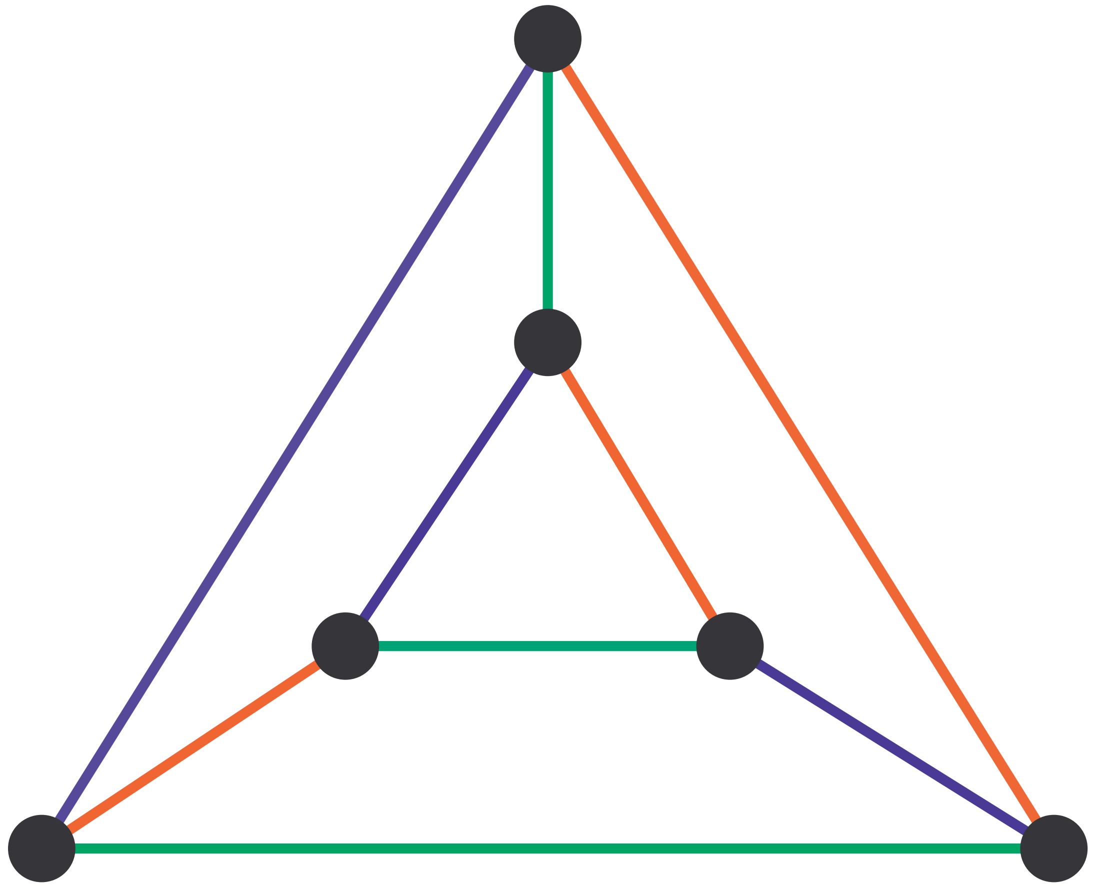
- ⋆ An assignment of colors to the edges such that two intersecting edges are assigned different colors.
- ⋆ A function $c : E \longrightarrow \{1, \ldots, \psi\}$ such that if $v \neq w$ and $(u, v), (u, w) \in E$ then $c(u, v) \neq c(u, w)$.

Observation: Both definitions are equivalent.

Example: Edge Coloring



Example: Edge Coloring with Minimum Number of Colors



The Edge Coloring Problem

The optimization problem: Find an edge coloring with minimum number of colors.

Notation: $\psi(G)$ – the chromatic index of G – the minimum number of colors required to color all the edges of G.

Hardness: A Hard problem to solve.

- $\star$ It is NP-Hard to decide if $\psi(G) = \Delta$ or $\psi(G) = \Delta + 1$ where Δ is the maximum degree in G.

Bounds on the Chromatic Index

- ⋆ Let Δ be the maximum degree in G.
- ⋆ Any edge coloring must use at least Δ colors.
  - $\psi(G) \geq \Delta$.
- ⋆ A greedy first-fit algorithm colors the edges of any graph with at most $2\Delta - 1$ colors.
  - $\psi(G) \leq 2\Delta - 1$.
- ⋆ There exists a polynomial time algorithm that colors any graph with at most $\Delta + 1$ colors.
  - $\psi(G) \leq \Delta + 1$.

More Bounds on the Chromatic Index

Notation: $M(G)$ – size of the maximum matching in G.

- $\star$ $M(G) \leq \lfloor n/2 \rfloor$.

Observation: $\psi(G) \geq \left\lceil \frac{m}{M(G)} \right\rceil$.

- $\star$ A pigeon hole argument: the size of each color-set is at most $M(G)$.

Corollary: $\psi(G) \geq \left\lceil \frac{m}{\lfloor n/2 \rfloor} \right\rceil$.

Vertex Coloring vs. Edge Coloring

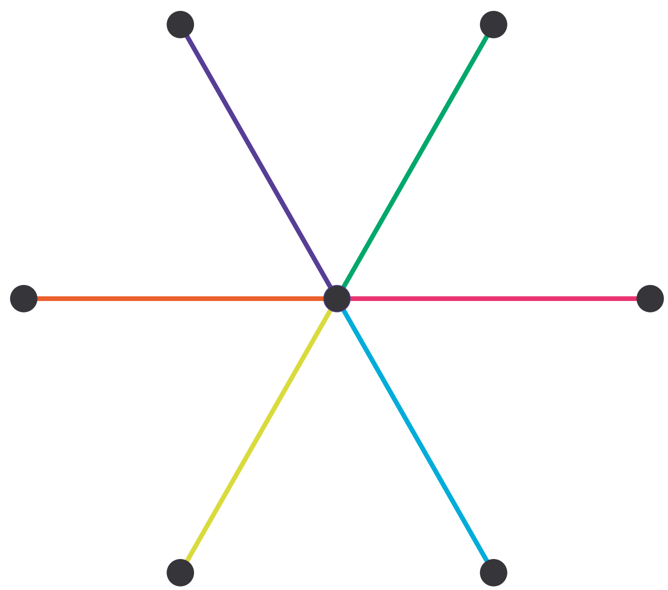
⋆ Matching is an easy problem while Independent Set is a hard problem.

⋆ Edge Coloring is a hard problem while Vertex Coloring is a very hard problem.

Coloring the Edges the Star Graph S_n

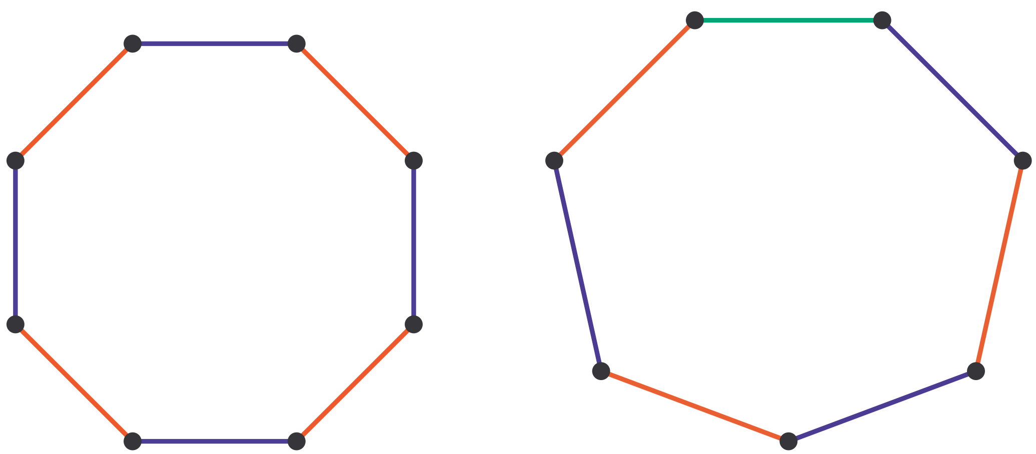
- ⋆ In a star graph $\Delta = n - 1$.
- ⋆ Each edge must be colored with a different color

$\Rightarrow$ $\psi(S_n) = \Delta$.



Coloring the Edges of the Cycle C_n

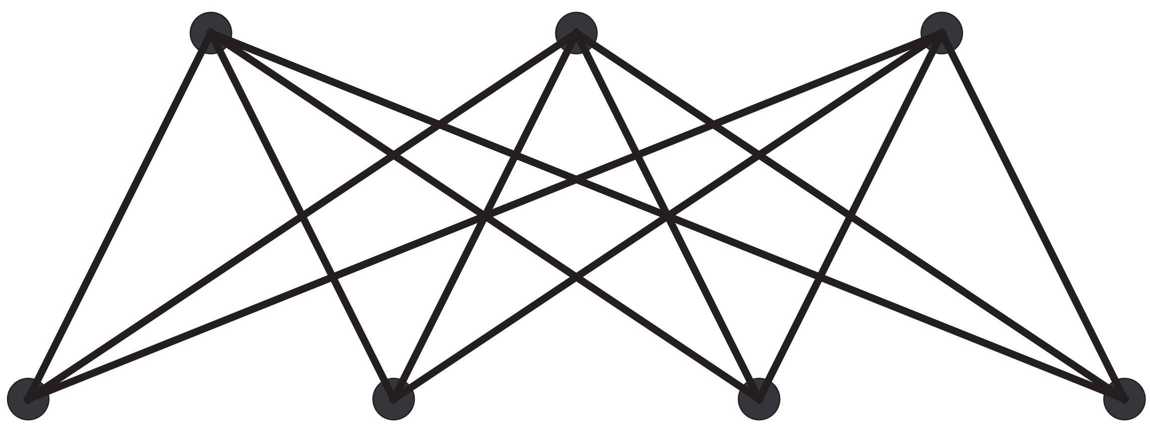
- $\star$ In any cycle $\Delta = 2$.
- $\star$ For even $n = 2k$, edges alternate colors and 2 colors are enough $\Rightarrow \psi(C_{2k}) = \Delta$.
- $\star$ For odd $n = 2k + 1$, at least 1 edge is colored with a third color $\Rightarrow \psi(C_{2k+1}) = \Delta + 1$.



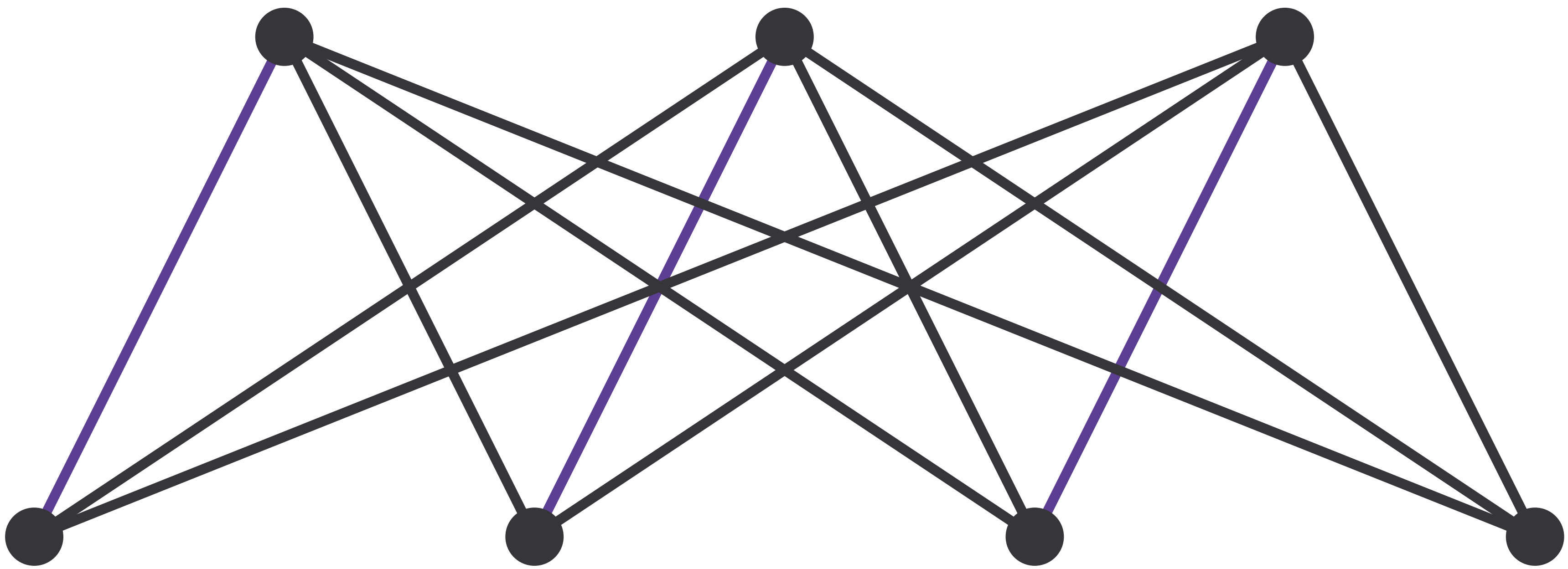
Complete Bipartite Graphs

Bipartite graphs: $V = A \cup B$ and each edge is incident to one vertex from A and one vertex from B.

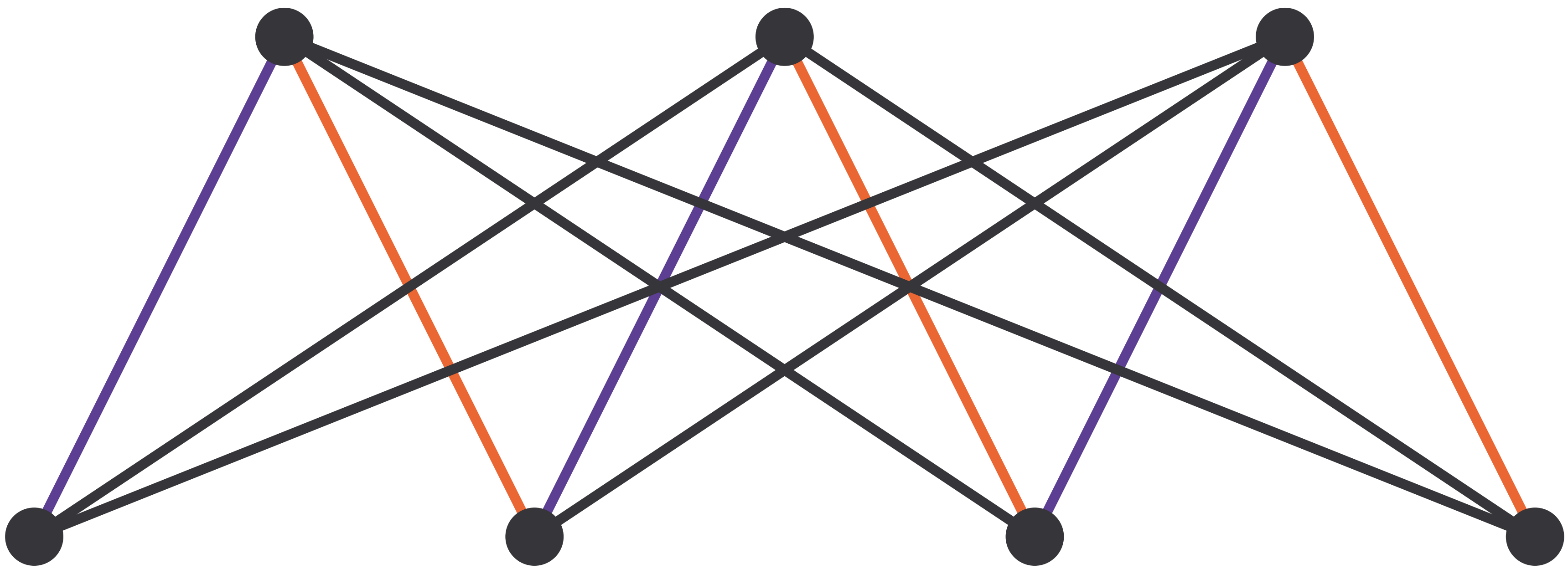
Complete bipartite graphs $K_{a,b}$: There are a vertices in A, b vertices in B, and all possible $a \cdot b$ edges exist.



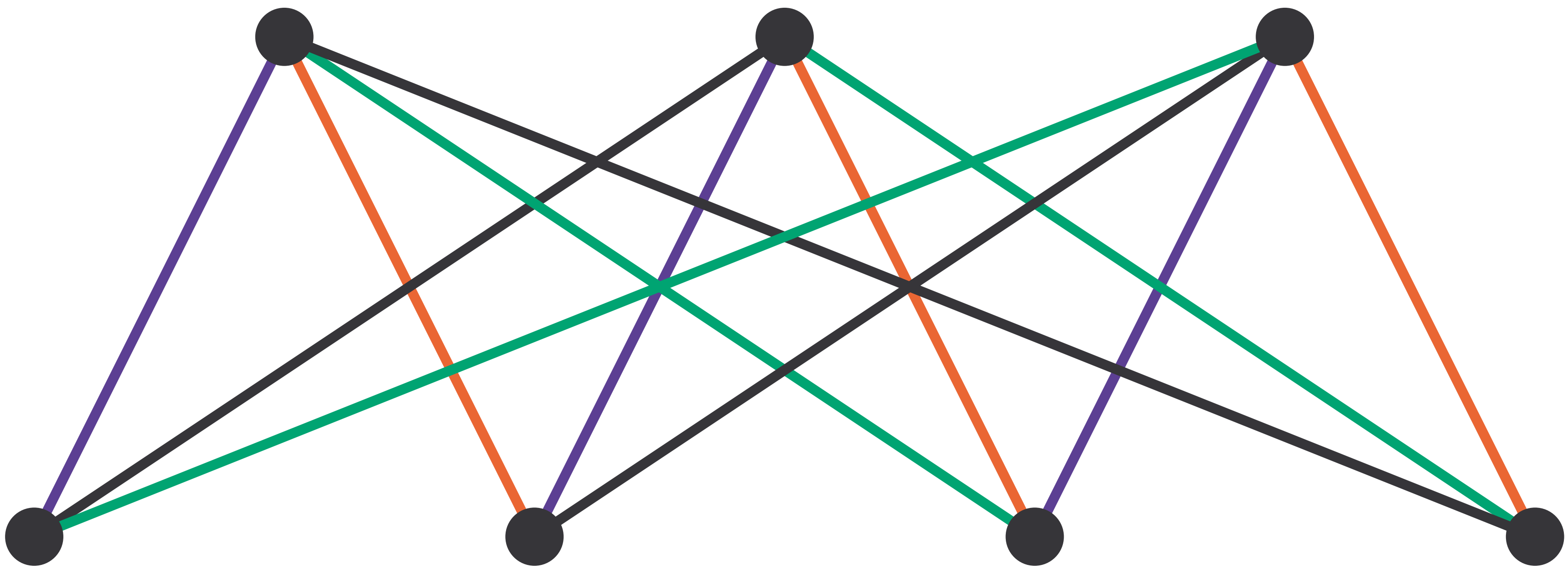
Coloring the Edges of $K_{3,4}$



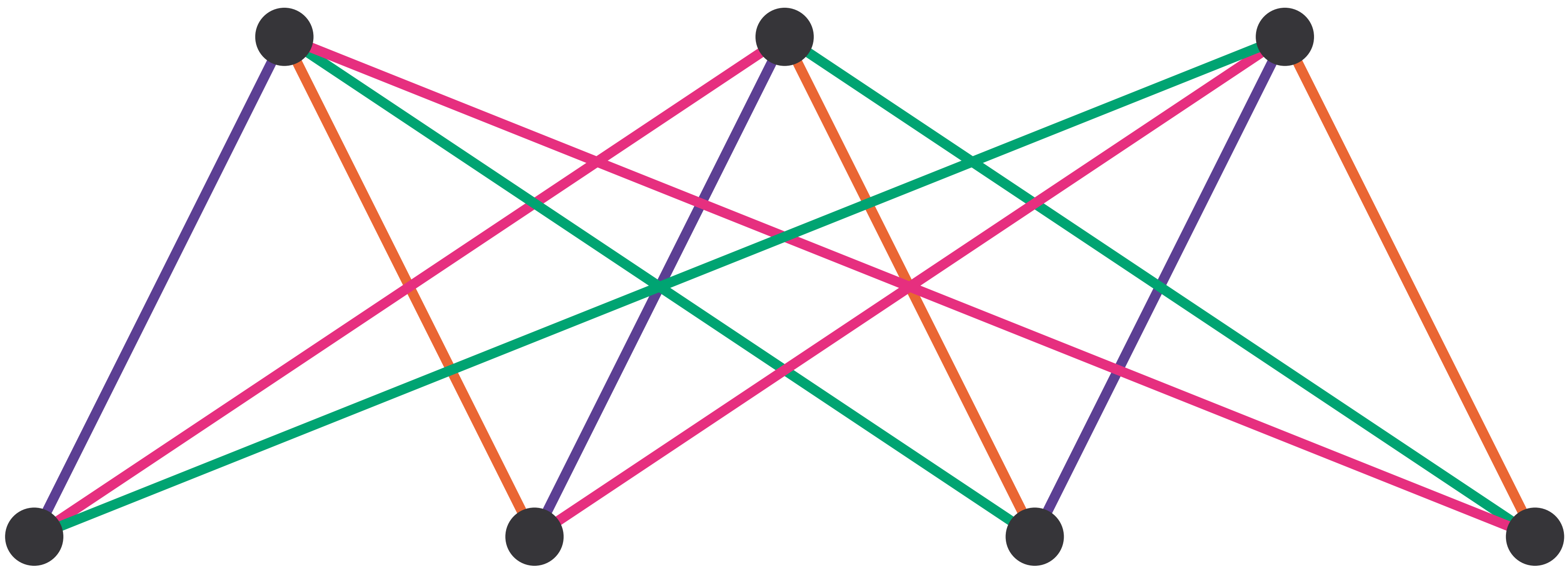
Coloring the Edges of $K_{3,4}$



Coloring the Edges of $K_{3,4}$



Coloring the Edges of $K_{3,4}$

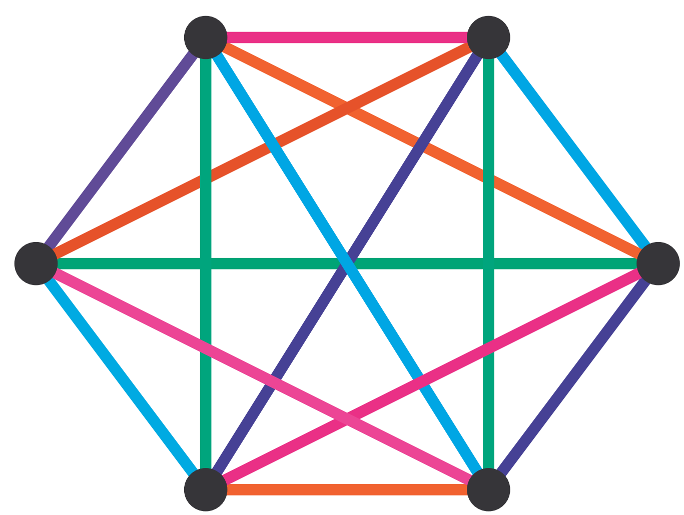


Coloring the Edges of Complete Bipartite Graphs

- $\Delta = \max\{a, b\}$ in the complete bipartite graph $K_{a,b}$.
- Let the vertices be $v_0, \ldots, v_{a-1} \in A$ and $u_0, \ldots, u_{b-1} \in B$.
- Assume $a \le b$.
- Color the edges in $b = \Delta$ rounds with the colors $0, \ldots, \Delta-1$.
- In round $0 \le i \le \Delta - 1$, color edges $(v_j, u_{(j+i) \bmod b})$ with color i for $0 \le j \le a$.

Complete Graphs – Even n

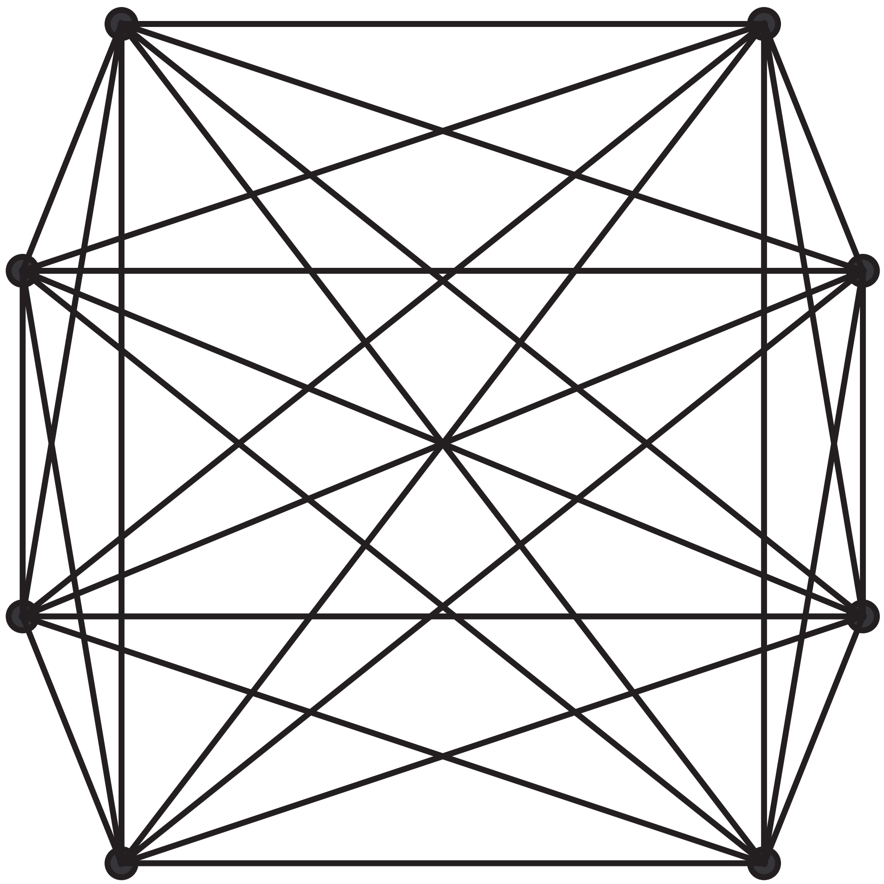
- ⋆ $\psi(K_n) = n - 1 = \Delta$ for an even n.
  - – $n - 1$ disjoint perfect matchings each of size $n/2$.
  - – $m = (n-1)(n/2)$ in a complete graph with n vertices.



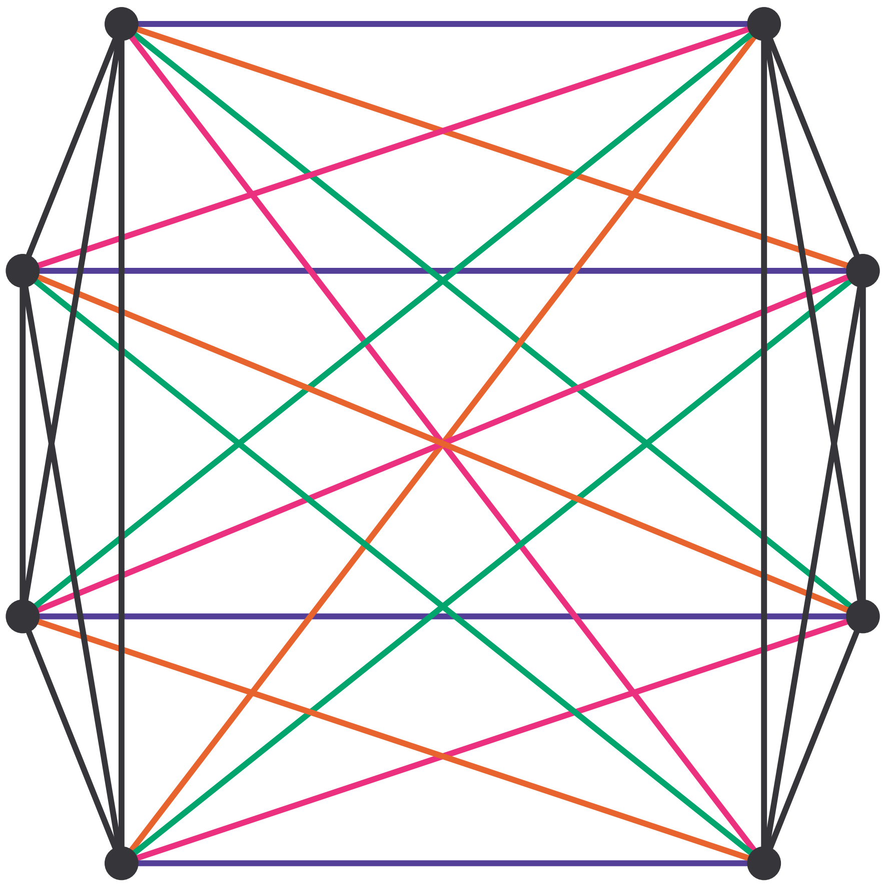
Complete Graphs – $n = 2^k$ power of 2

- $\star$ If $k = 1$ color the only edge with $1 = \Delta$ color.
- $\star$ If $k > 1$, partition the vertices into two $K_{n/2}$ cliques A and B each with 2^{k-1} vertices.
- $\star$ Color the complete bipartite $K_{n/2,n/2}$ implied by the partition $V = A \cup B$ with $n/2$ colors.
- $\star$ Recursively and in parallel, color both cliques A and B with $n/2 - 1 = \Delta(K_{n/2})$ colors.
- $\star$ All together, $n - 1 = \Delta$ colors were used.

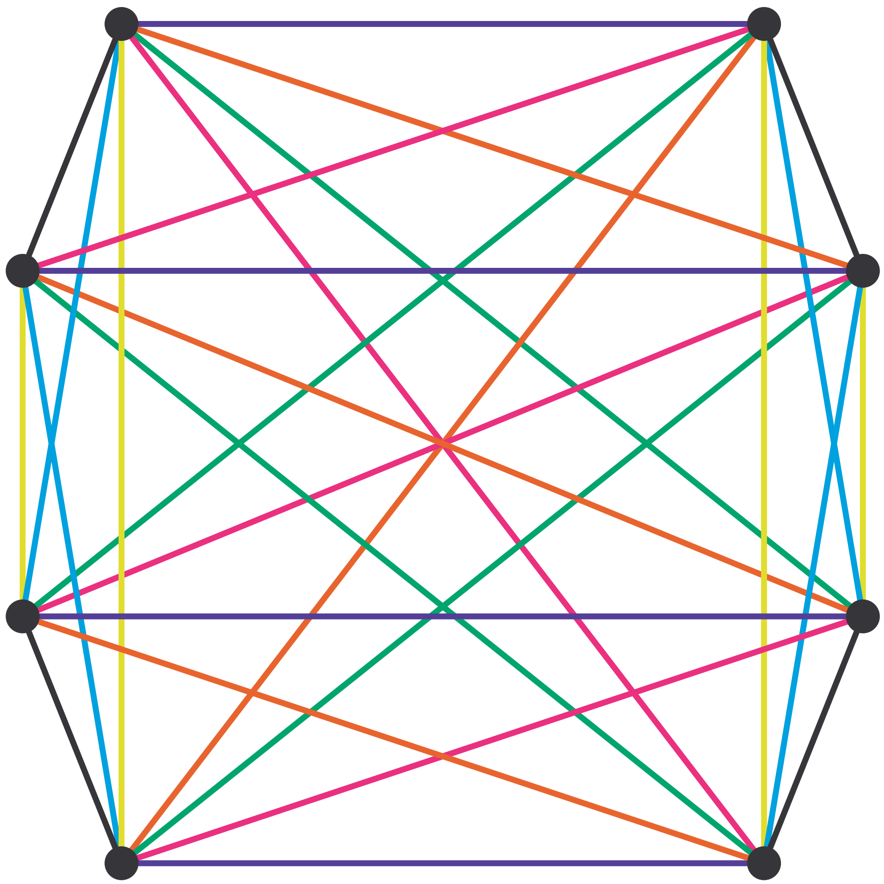
Coloring the edges of K_8



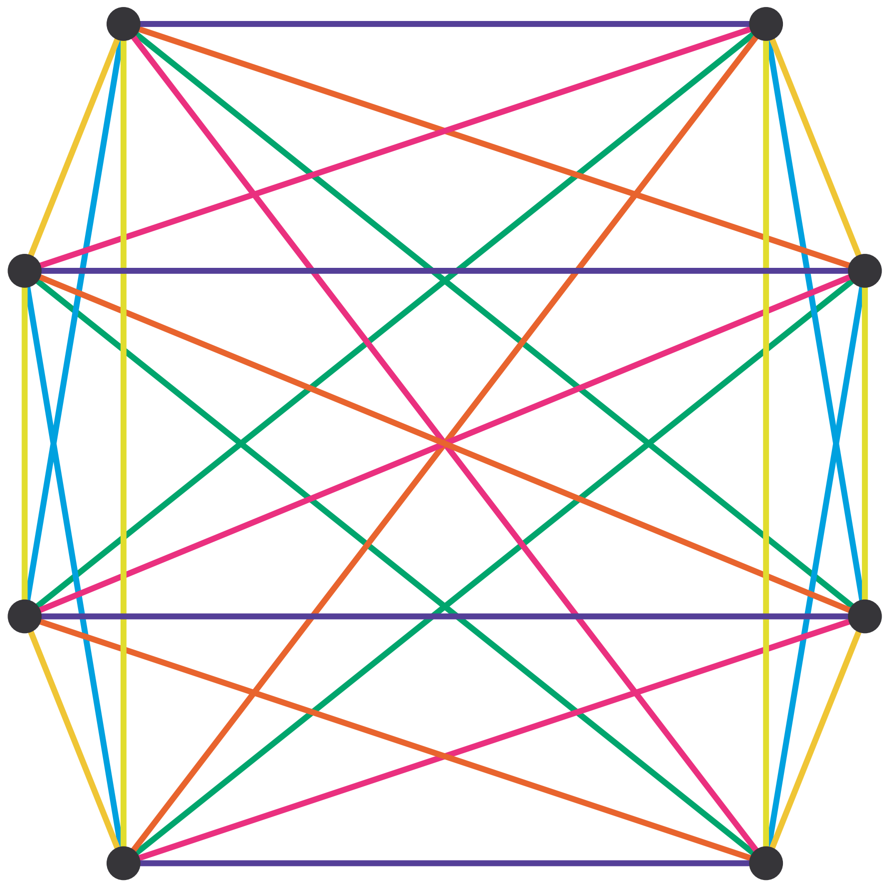
Coloring the edges of K_8



Coloring the edges of K_8

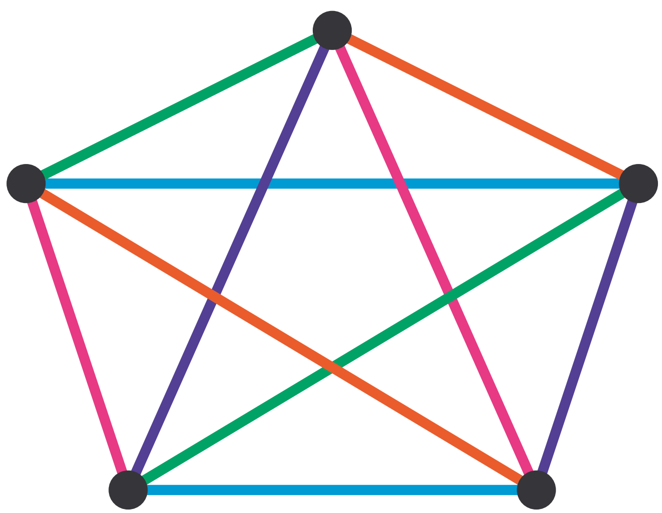


Coloring the edges of K_8



Complete Graphs – Odd n

- ⋆ $\psi(K_n) = n = \Delta + 1$ for an odd n.
  - $\psi(K_n) \leq \psi(K_{n+1}) = n$ because $n+1$ is even.
  - $\psi(K_n) \geq \frac{(n(n-1)/2)}{((n-1)/2)} = n$ because there are $\frac{n(n-1)}{2}$ edges in K_n and the size of the maximum matching is $\frac{n-1}{2}$.



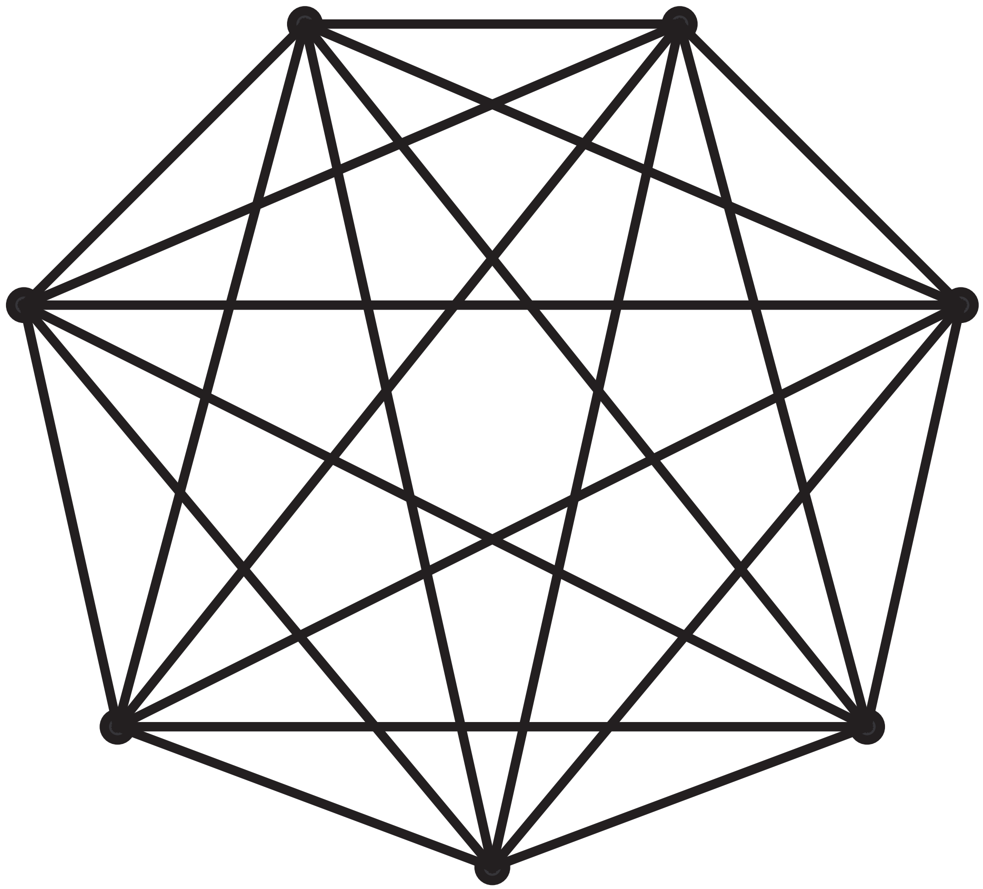
Coloring the Edges of Odd Complete Graphs

⋆ Arrange the vertices as a regular n-polygon.

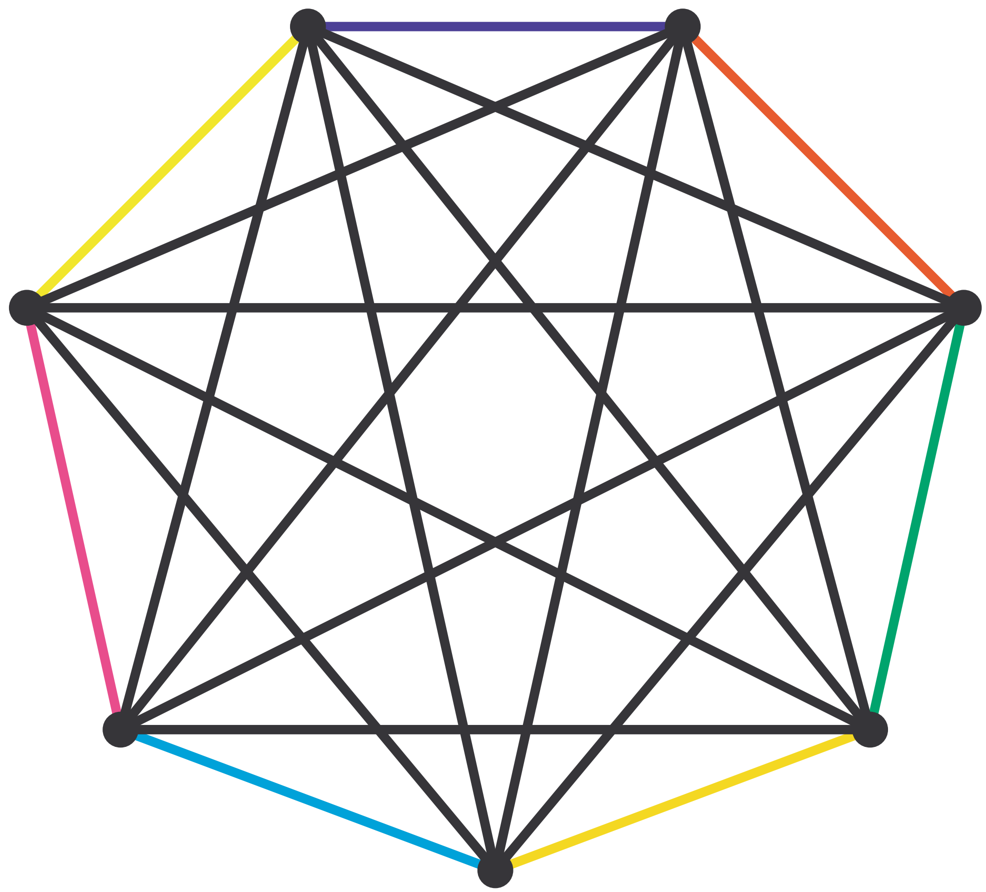
⋆ Color the n edges on the perimeter of the polygon using n colors.

⋆ Color an inside edge with the color of its parallel edge on the perimeter of the polygon.

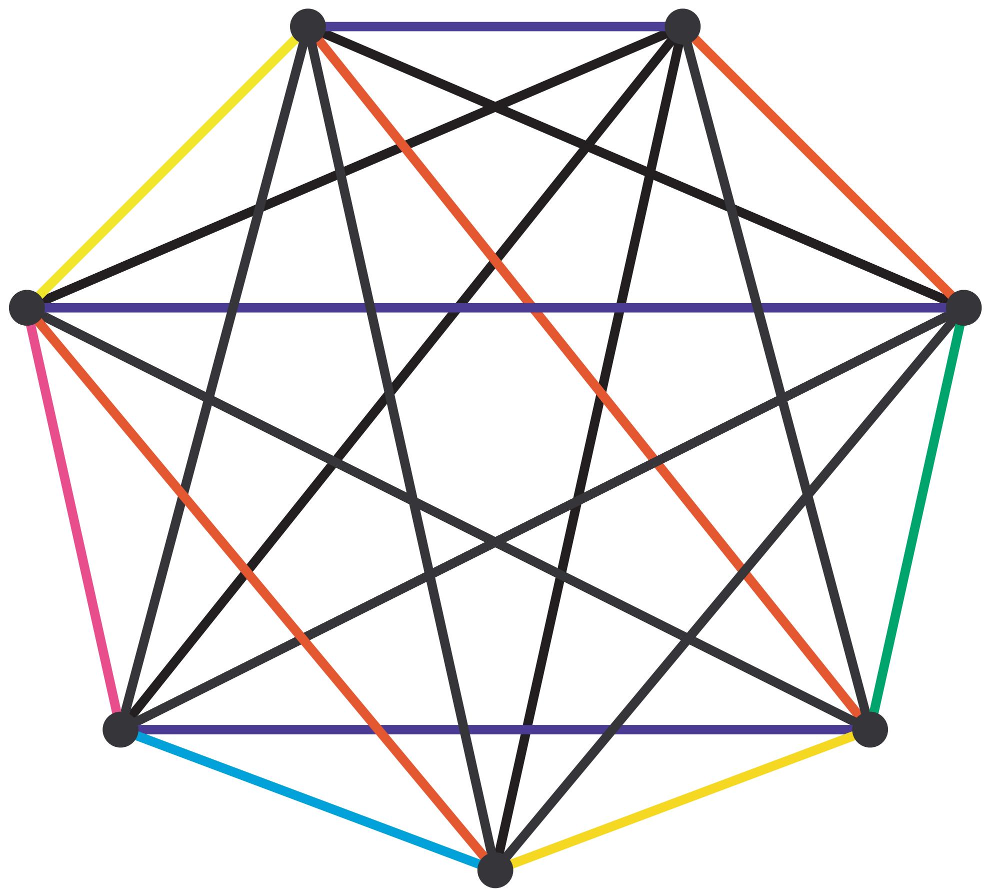
Coloring the Edges of K_7



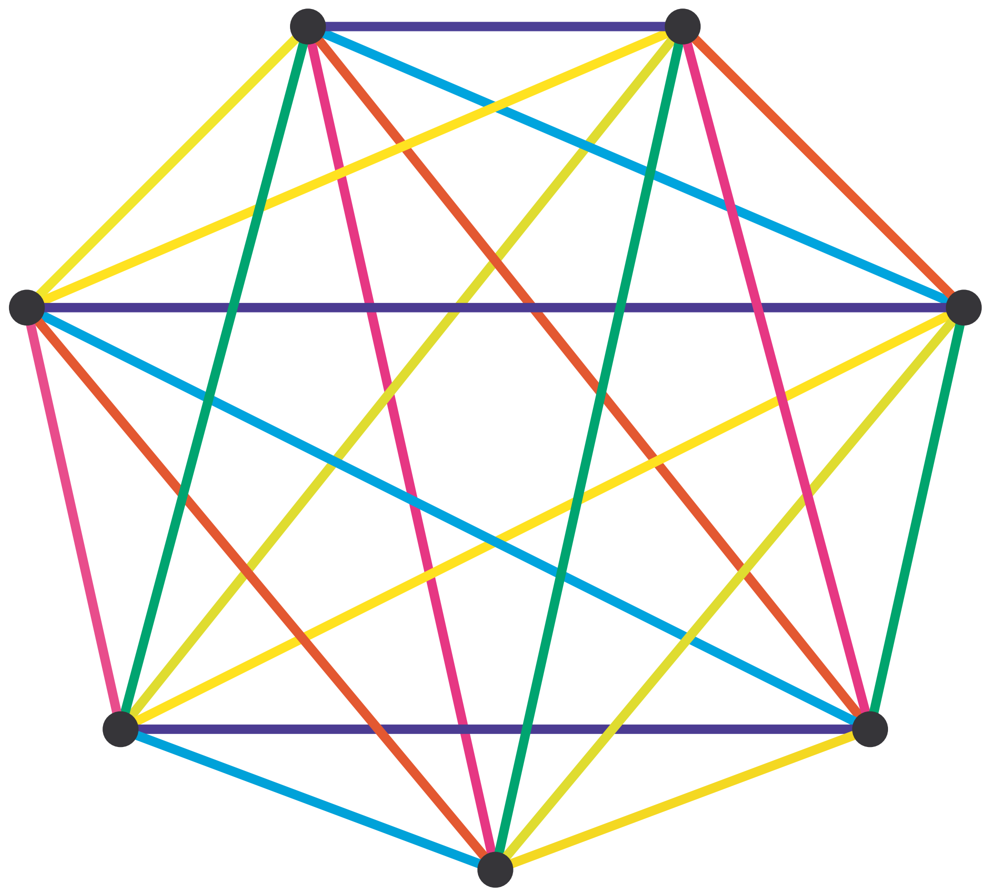
Coloring the Edges of K_7



Coloring the Edges of K_7



Coloring the Edges of K_7



Coloring the Edges of Odd Complete Graphs

Correctness: The coloring is legal:

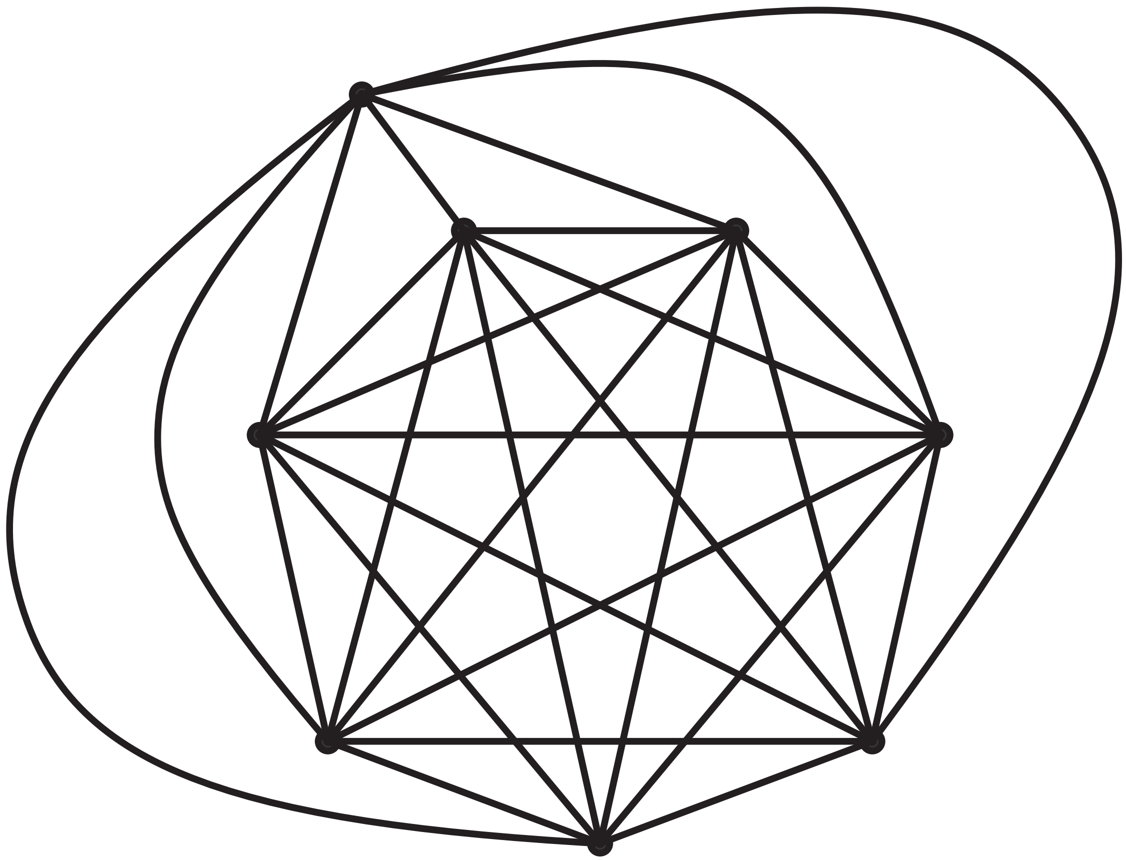
- ⋆ Parallel edges do not intersect.
- ⋆ Each edge is parallel to exactly one perimeter edge.

Number of colors: $\psi(K_{2k+1}) = 2k + 1 = \Delta + 1.$

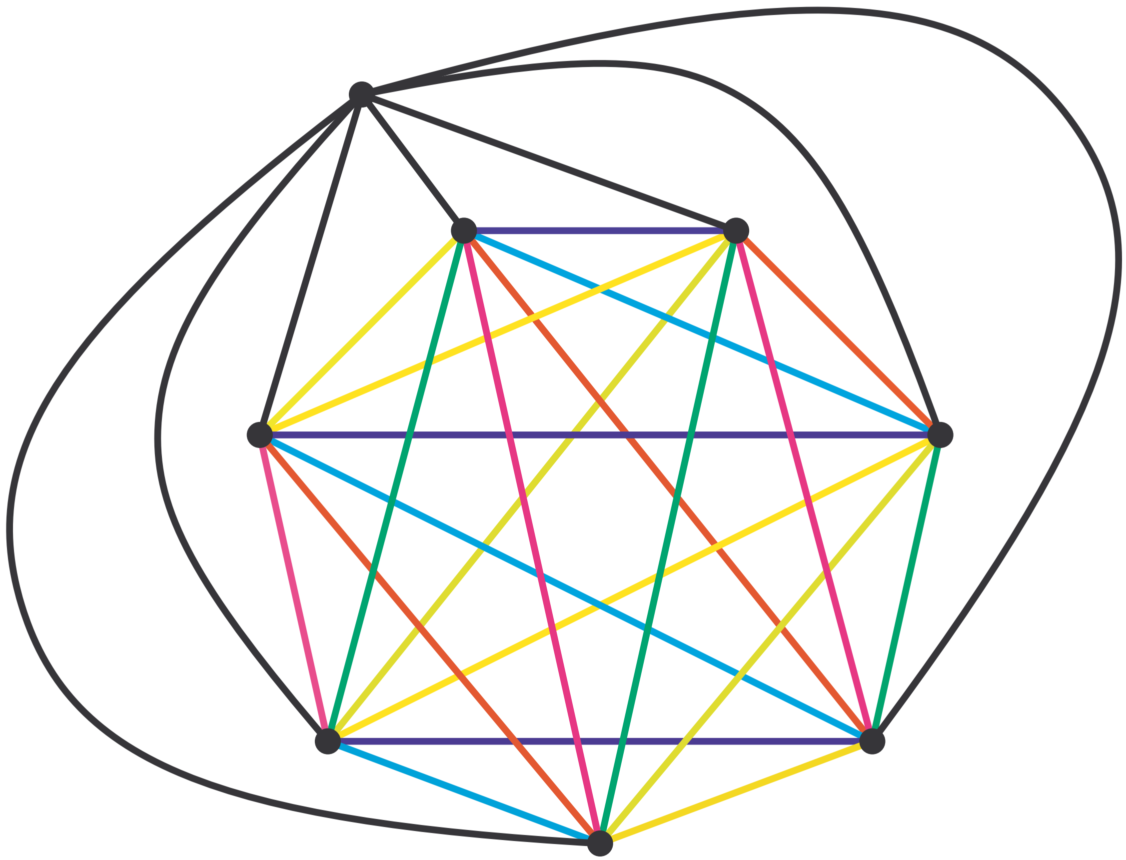
Coloring the Edges of Even Complete Graphs – Algorithm I

- ⋆ Let the vertices be $0, \ldots, n-1$.
- ⋆ Color the edges of K_{n-1} on the vertices $0, \ldots, n-2$ using the polygon algorithm.
- ⋆ For $0 \leq x \leq n-2$, color the edge $(n-1, x)$ with the only color missing at x.

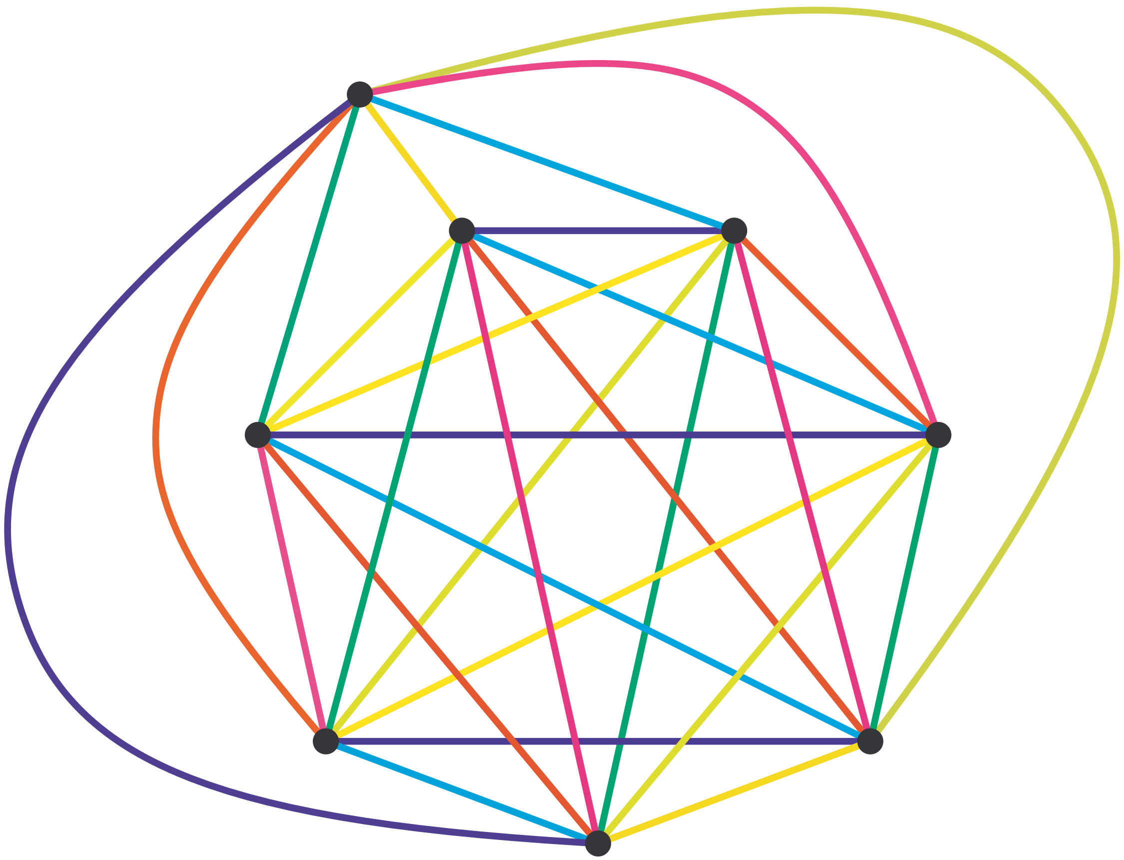
Coloring the Edges of K_8



Coloring the Edges of K_8



Coloring the Edges of K_8



Coloring the Edges of Even Complete Graphs – Algorithm I

Correctness: The coloring is legal:

- ⋆ The coloring of K_{n-1} implies a legal coloring for all the edges except those incident to vertex $n-1$.
- ⋆ Exactly 1 color is missing at a vertex $x \in \{0, \ldots, n-2\}$.
- ⋆ This is the color of the perimeter edge opposite to it.
- ⋆ The $n-1$ perimeter edges are colored with different colors.

Number of colors: $\psi(K_{2k}) = 2k - 1 = \Delta$.

Coloring the Edges of Even Complete Graphs – Algorithm II

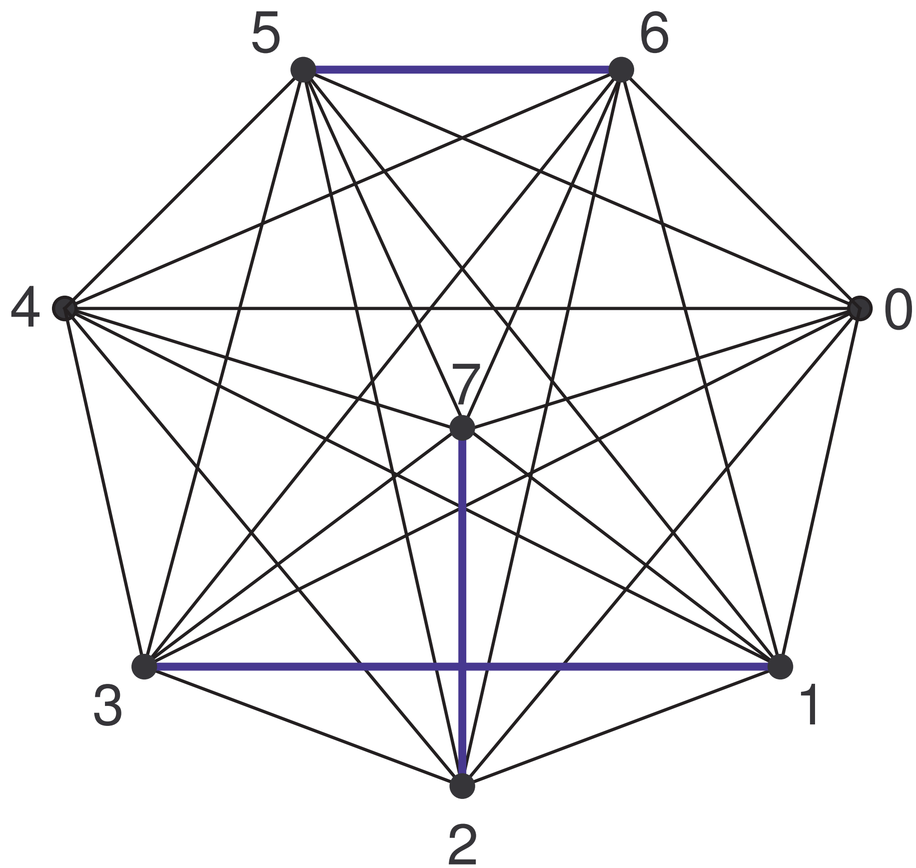
- ⋆ Let the vertices be $0, \ldots, n-1$.
- ⋆ In round $0 \le x \le n-2$, color the following edges with the color x:
  - $(n-1, x)$.
  - $(((x-1) \bmod (n-1)), ((x+1) \bmod (n-1)))$.
  - $(((x-2) \bmod (n-1)), ((x+2) \bmod (n-1)))$.

$$\vdots \qquad\qquad\qquad \vdots$$

  - $(((x-(\frac{n}{2}-1)) \bmod (n-1), ((x+(\frac{n}{2}-1)) \bmod (n-1)))$.

Correctness and number of colors: The same as Algorithm I since both algorithms are equivalent!

Coloring the Edges of Even Complete Graphs – Algorithm II



Coloring the Edges of Even Complete Graphs – Algorithm III

- ⋆ Let $0 \le i < j \le n-1$ be 2 vertices.
- ⋆ Case $j = n-1$:

  color the edge (i,j) with the color i.
- ⋆ Case $i+j$ is even:

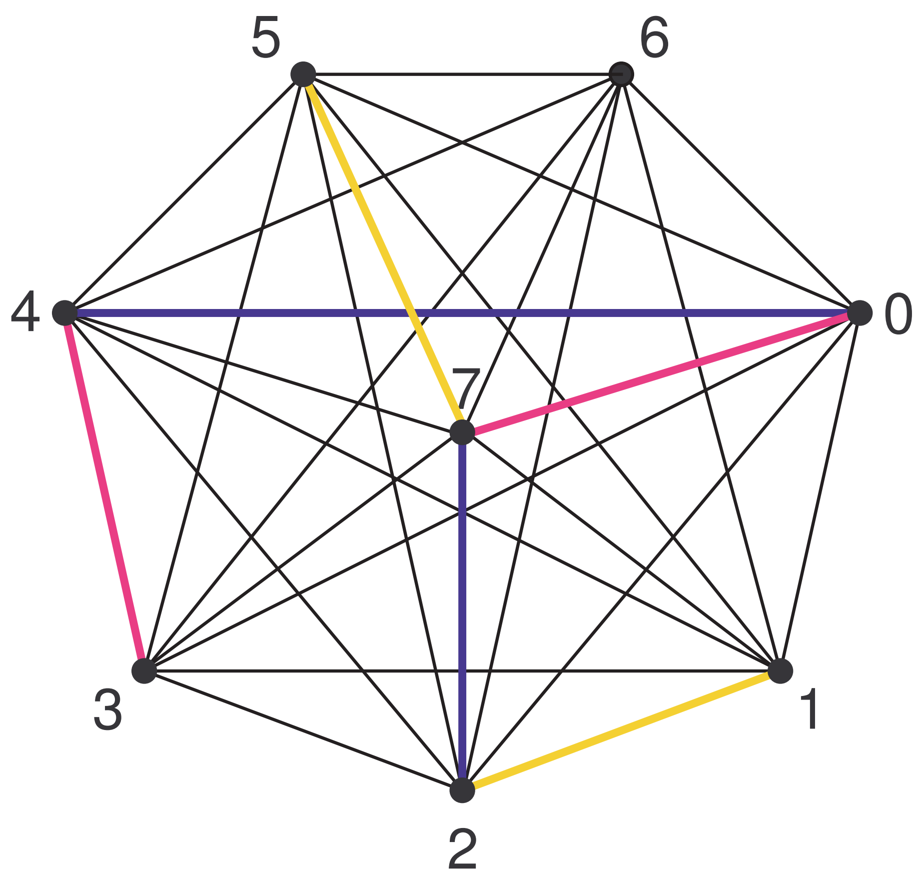
  color the edge (i,j) with the color $\frac{i+j}{2}$.
- ⋆ Case $i+j$ is odd and $i+j < n-1$:

  color the edge (i,j) with the color $\frac{i+j+(n-1)}{2}$.
- ⋆ Case $i+j$ is odd and $i+j \ge n-1$:

  color the edge (i,j) with the color $\frac{i+j-(n-1)}{2}$.

Correctness and number of edges: The same as Algorithm I and Algorithm II since all three algorithms are equivalent!

Coloring the Edges of Even Complete Graphs – Algorithm III

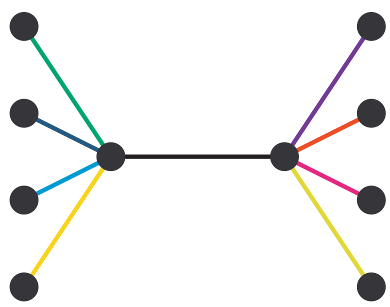


Greedy First Fit Edge Coloring

Algorithm: Color the edges of G with $2\Delta - 1$ colors.

- ⋆ Consider the edges in an arbitrary order.
- ⋆ Color an edge with the first available color among $\{1, 2, \ldots, 2\Delta - 1\}$.

Correctness: At the time of coloring, at most $2\Delta - 2$ colors are not available. By the pigeon hole argument there exists an available color.



Complexity

- m edges to color.
- $O(\Delta)$ to find the available color for any edge.
- Overall, $O(\Delta m)$ complexity.

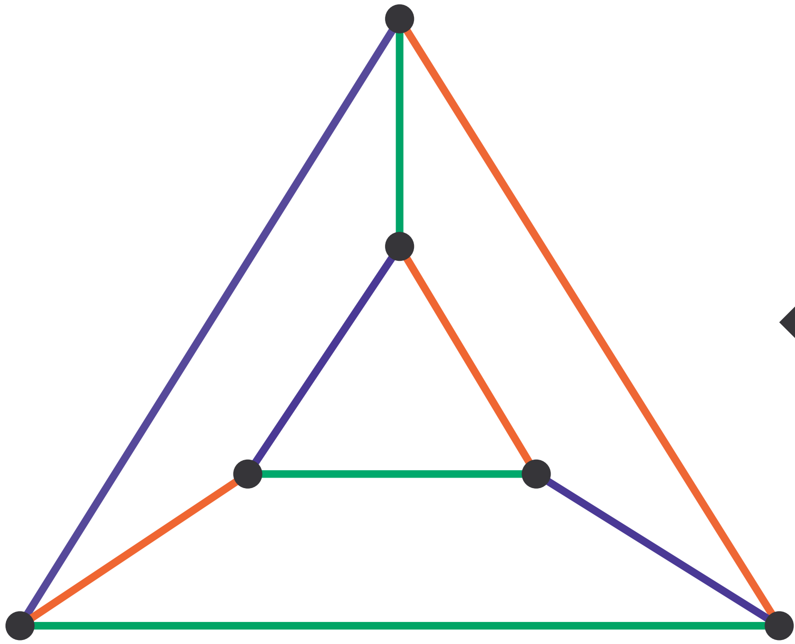
Coloring the Vertices of the Line Graph $L(G)$

- ⋆ Coloring the edges of G is equivalent to coloring the vertices of the line graph $L(G)$ of G.

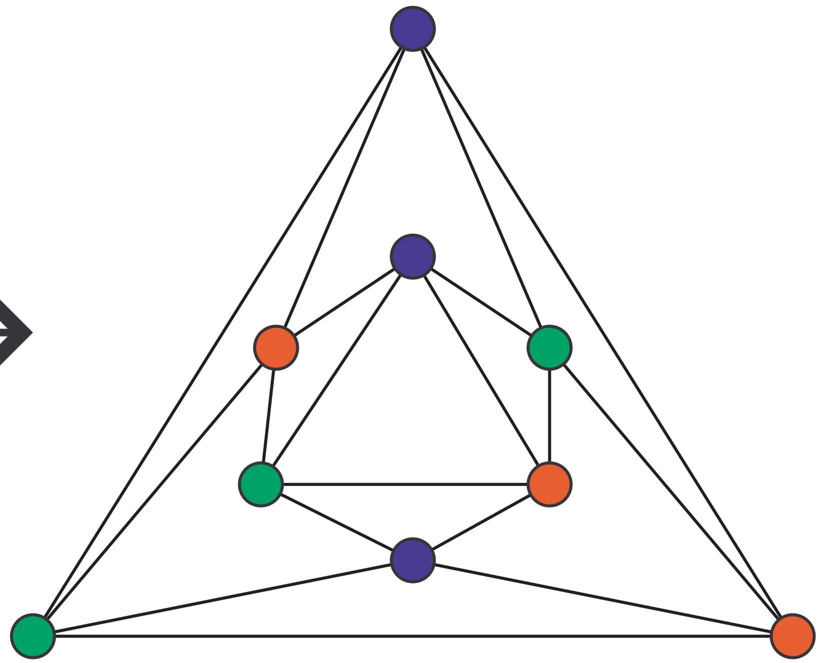
- ⋆ If the maximum degree in G is Δ then the maximum degree in $L(G)$ is $\Delta(L(G)) = 2\Delta - 2$.

- ⋆ The greedy coloring of the edges of G with $2\Delta - 1$ colors is equivalent to the greedy coloring of the vertices of $L(G)$ with $\Delta(L(G)) + 1 = 2\Delta - 1$ colors.

Coloring the Vertices of the Line Graph $L(G)$



$\longleftrightarrow$



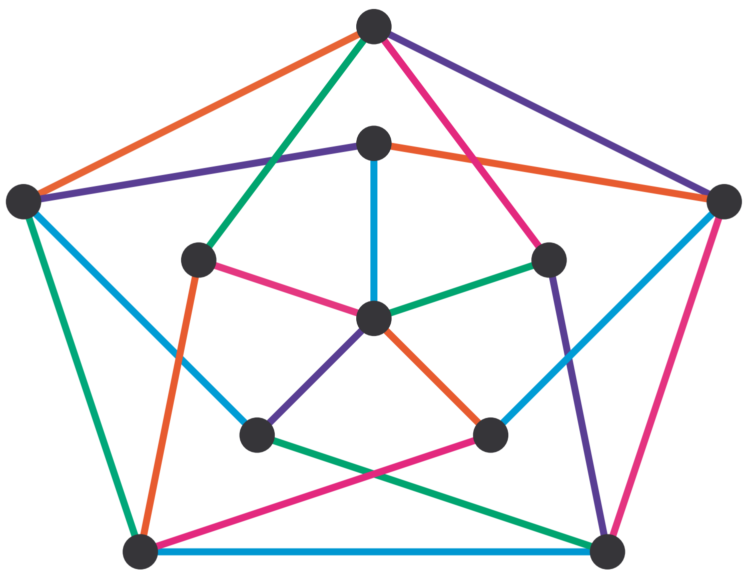
Subgraphs Definitions

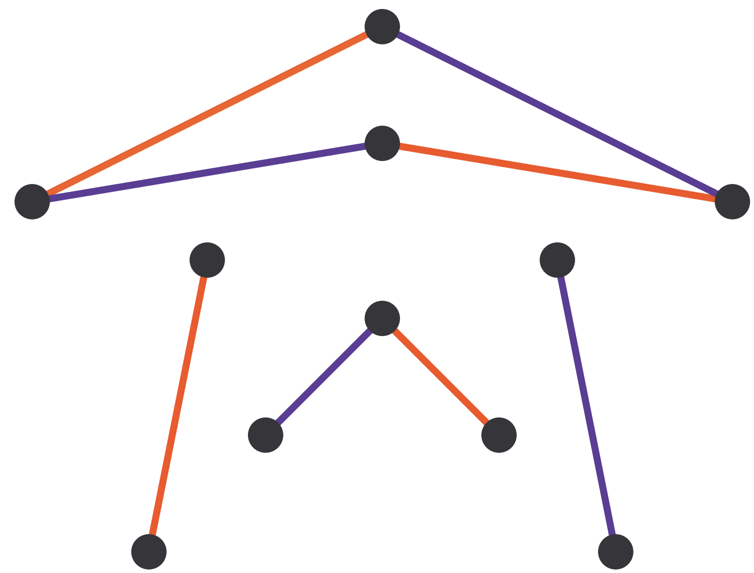
- ⋆ For colors x and y, let $G(x, y)$ be the subgraph of G containing all the vertices of G and only the edges whose colors are x or y.

- ⋆ For a vertex w, let $G_w(x, y)$ be the connected component of $G(x, y)$ that contains w.

Observation

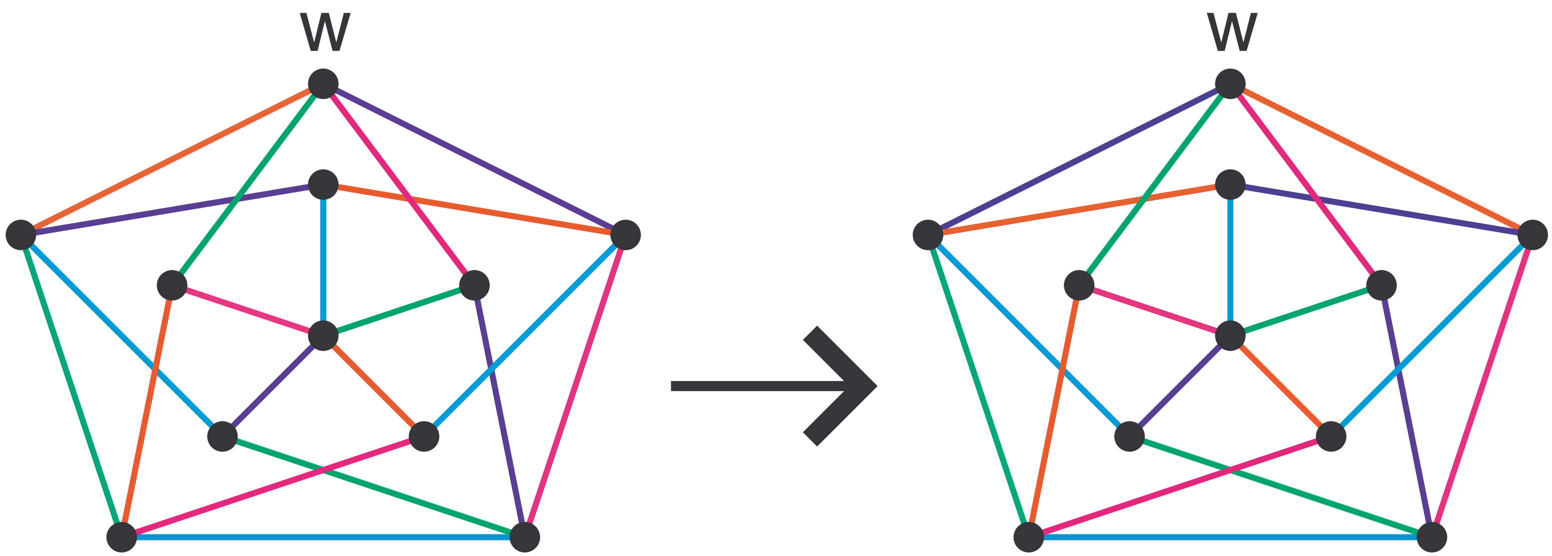
⋆ $G(x, y)$ is a collection of even size cycles and paths.





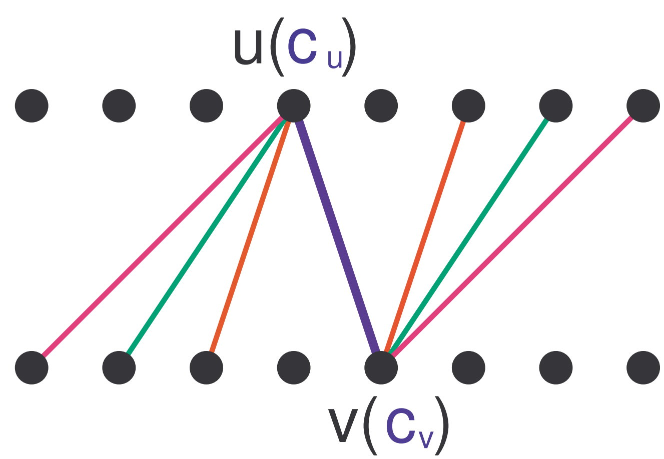
Exchanging Colors Tool

⋆ Exchanging between the colors x and y in the connected component $G_w(x, y)$ of $G(x, y)$ results with another legal coloring.



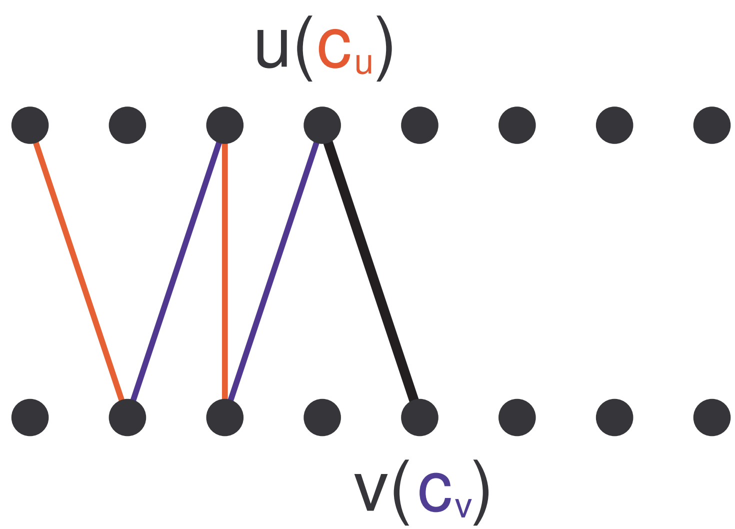
Coloring the edges of Bipartite Graphs with Δ Colors

- ⋆ Color the edges with the colors $\{1, 2, \ldots, \Delta\}$ following an arbitrary order. Let (u, v) be the next edge to color.
- ⋆ At most $\Delta - 1$ edges containing u or v are colored $\Rightarrow$ one color c_u is **missing** at u and one color c_v is **missing** at v.
- ⋆ If $c_u = c_v$ then color the edge (u, v) with the color $c_u = c_v$.



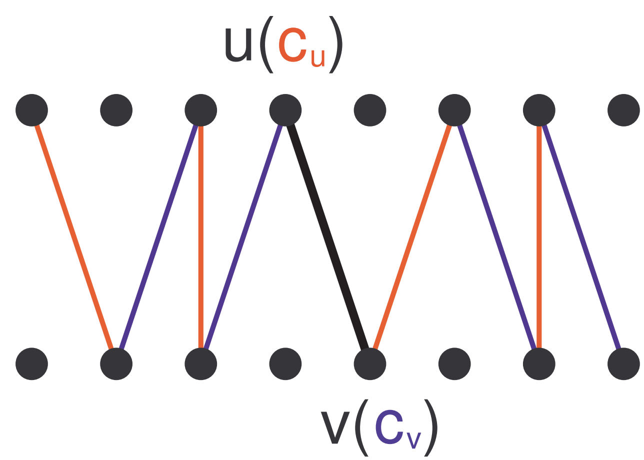
Coloring the Edges of Bipartite Graphs with Δ Colors

- Assume $c_u \neq c_v$.
- Let $G_u(c_u, c_v)$ be the connected component of the subgraph of G containing only edges colored with c_u and c_v.
- $G_u(c_u, c_v)$ is a path starting at u with a c_v colored edge.



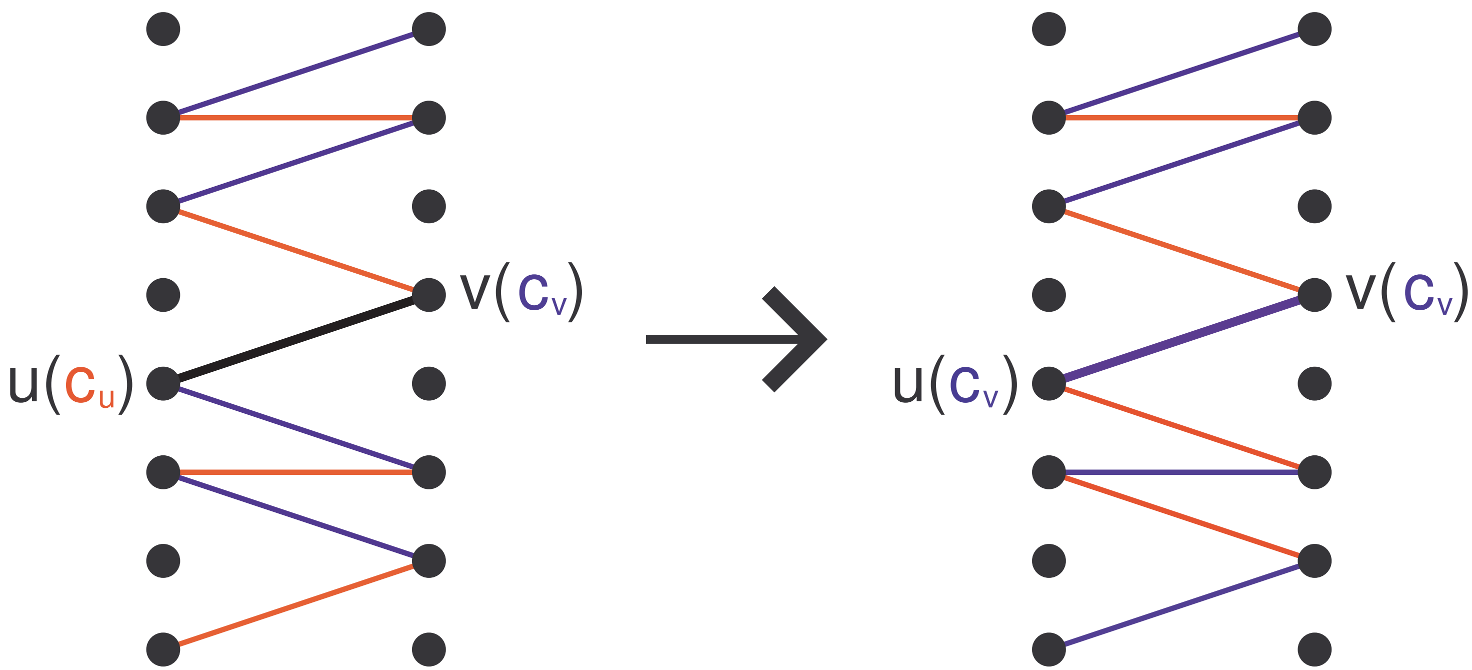
Coloring the Edges of Bipartite Graphs with Δ Colors

- The colors in $G_u(c_u, c_v)$ alternate between c_v and c_u.
- The first edge in the path starting at u is colored c_v $\Rightarrow$ any edge in the path that starts at the side of u must be colored with c_v.
- v does not belong to $G_u(c_u, c_v)$ because c_v is missing at v.



Coloring the Edges of Bipartite Graphs with $\triangle$ Colors

- ⋆ Exchange between the colors c_u and c_v in $G_u(c_u, c_v)$.
- ⋆ c_v is missing at both u and v: color the edge (u, v) with the color c_v.

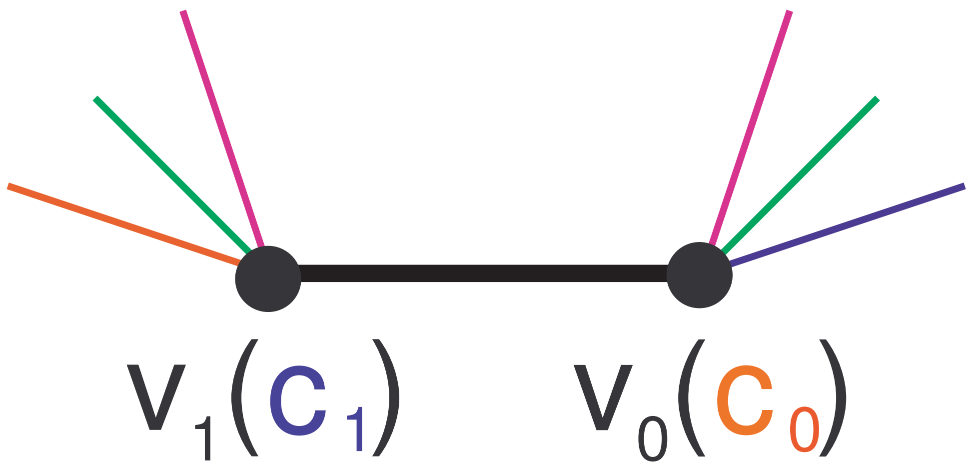


Complexity

- ⋆ m edges to color.
- ⋆ $O(\Delta)$ to find the missing colors at u and v.
- ⋆ $O(n)$ to change the colors in $G_u(c_u, c_v)$.
- ⋆ Overall, $O(nm)$ complexity.

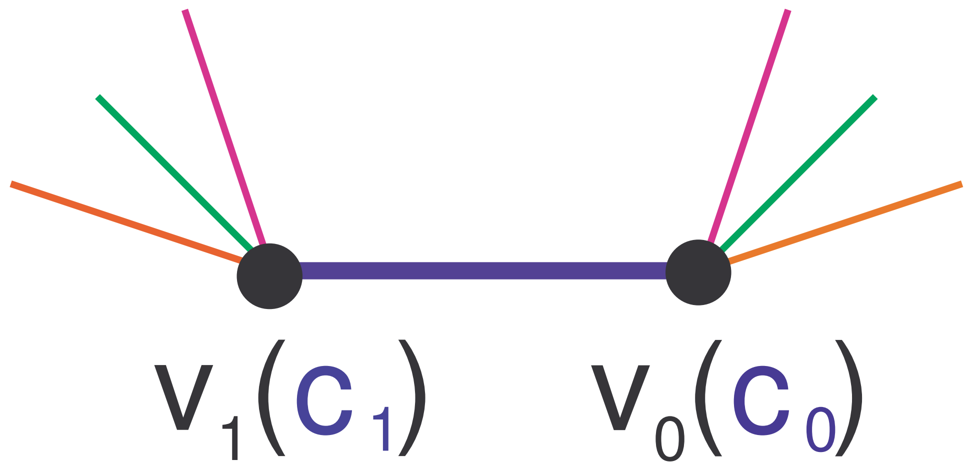
Coloring the Edges of Any Graph with $\Delta + 1$ Colors

- ⋆ Color the edges with the colors $\{1, 2, \ldots, \Delta + 1\}$ following an arbitrary order. Let (v_0, v_1) be the next edge to color.
- ⋆ At most $\Delta - 1$ edges containing v_0 or v_1 are colored $\Rightarrow$ one color c_0 is missing at v_0 and one color c_1 is missing at v_1.



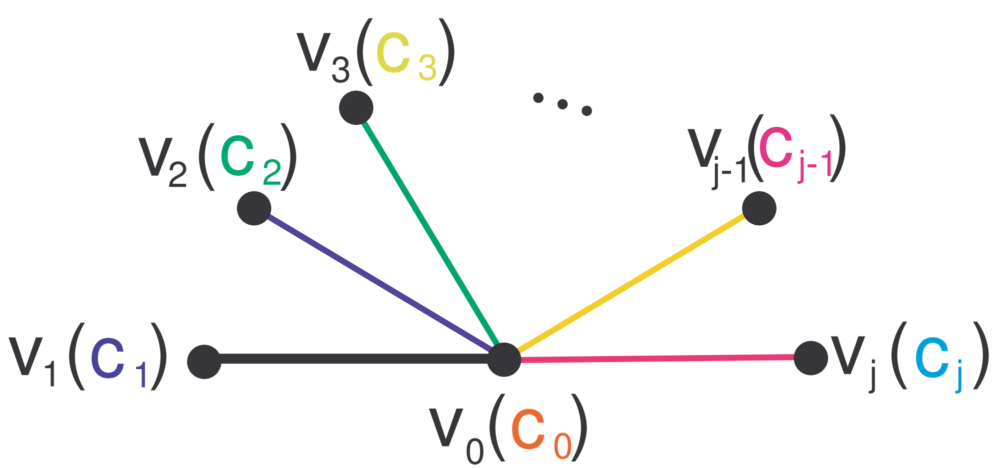
Coloring the edges of Any Graph with $\Delta + 1$ Colors

⋆ If $c_0 = c_1$ then color the edge (v_0, v_1) with the color $c_0 = c_1$.



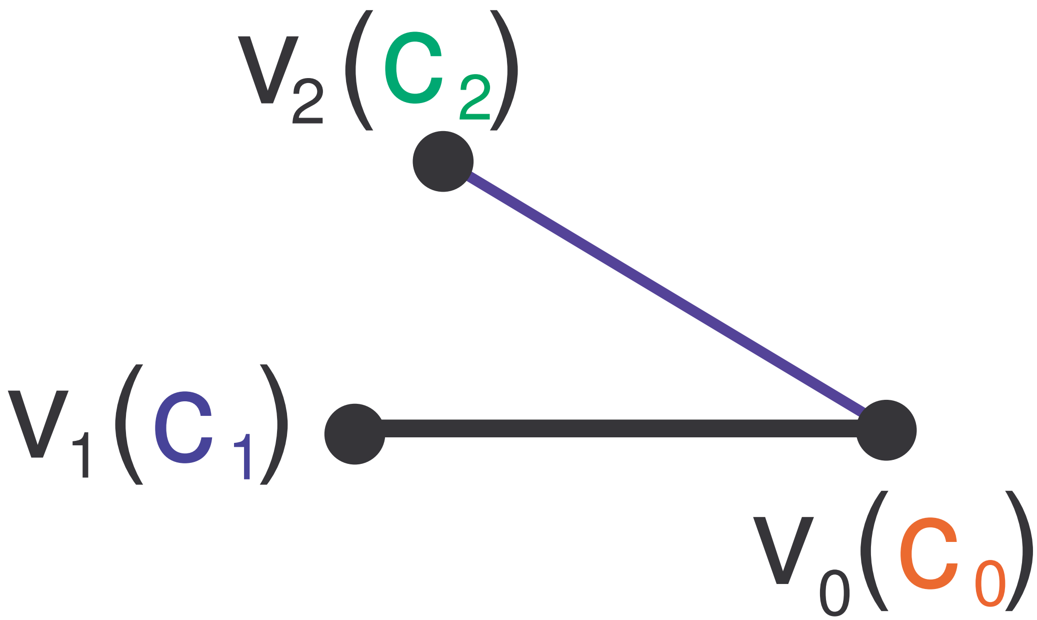
Coloring the Edges Any Graph with $\Delta + 1$ Colors

- Assume $c_0 \neq c_1$.
- Construct a sequence of distinct colors $c_0, c_1, c_2, \ldots, c_{j-1}, c_j$ and a sequence of edges $(v_0, v_1), (v_0, v_2), \ldots, (v_0, v_j)$.
  - Color c_i is missing at v_i for $0 \leq i \leq j$.
  - c_i is the color of the edge (v_0, v_{i+1}) for $1 \leq i \leq j$.

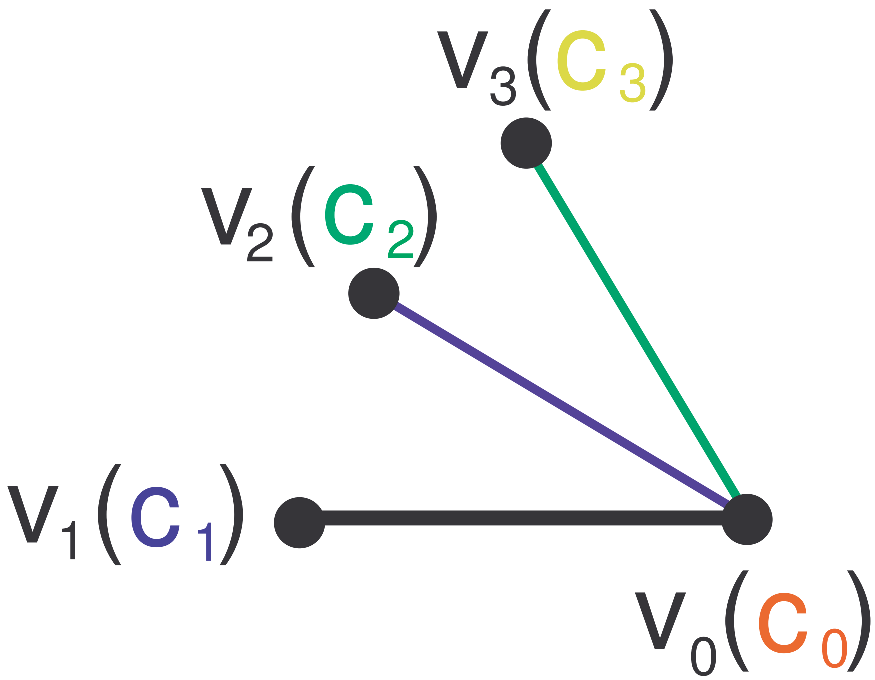


Constructing the Sequence

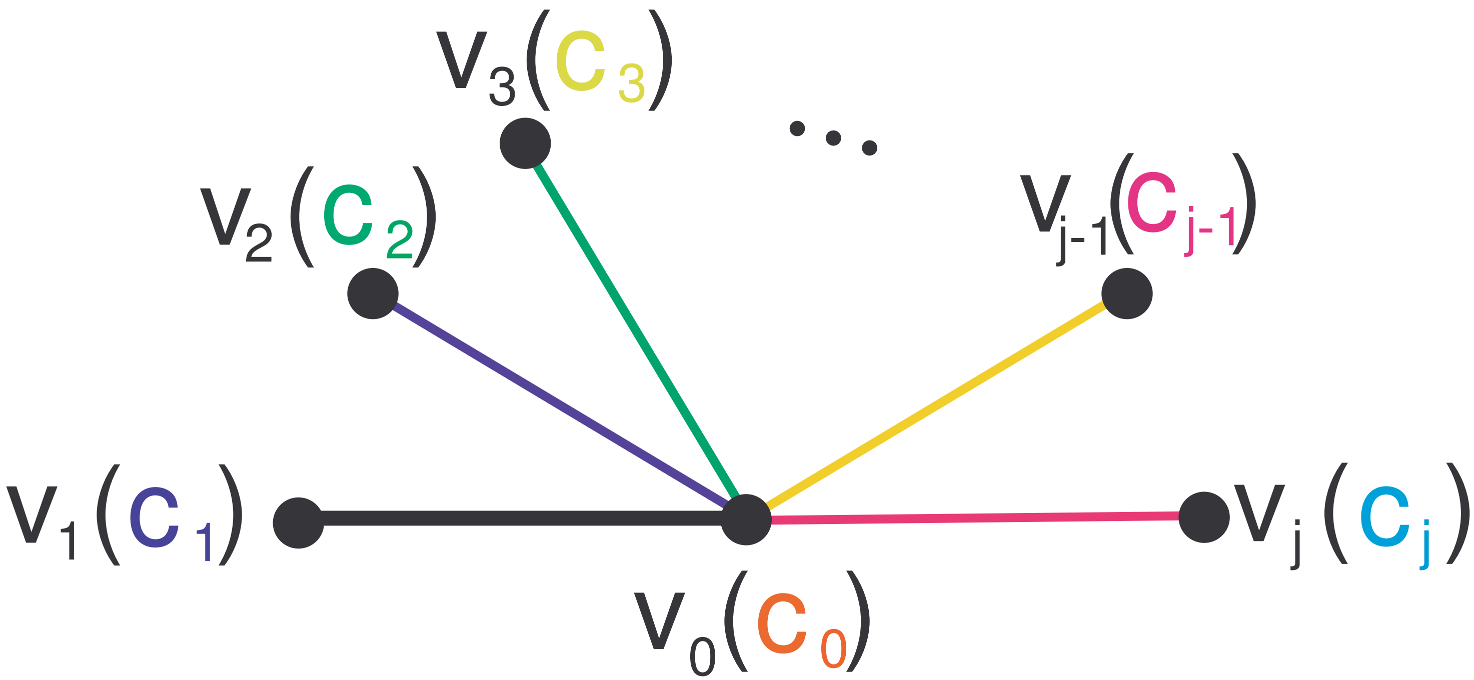
Constructing the Sequence



Constructing the Sequence



Constructing the Sequence



Constructing the Sequence

- ⋆ The edge (v_0, v_1) and the colors c_0, c_1 are initially defined.
- ⋆ Assume the colors $c_0, c_1, \ldots, c_{j-1}$ and the edges $(v_0, v_1), (v_0, v_2), \ldots, (v_0, v_j)$ are defined:
  - c_i is missing at v_i for $0 \leq i \leq j-1$.
  - c_i is the color of the edge (v_0, v_{i+1}) for $1 \leq i \leq j-1$.
- ⋆ Let c_j be a color missing at v_j.
- ⋆ If there exists an edge (v_0, v_{j+1}) colored with c_j, where $v_{j+1} \notin \{v_1, \ldots, v_j\}$, then continue constructing the sequence with the defined c_j and v_{j+1}.

The Process Terminates

⋆ Since v_0 has only Δ neighbors, the construction process stops with one of the following cases:

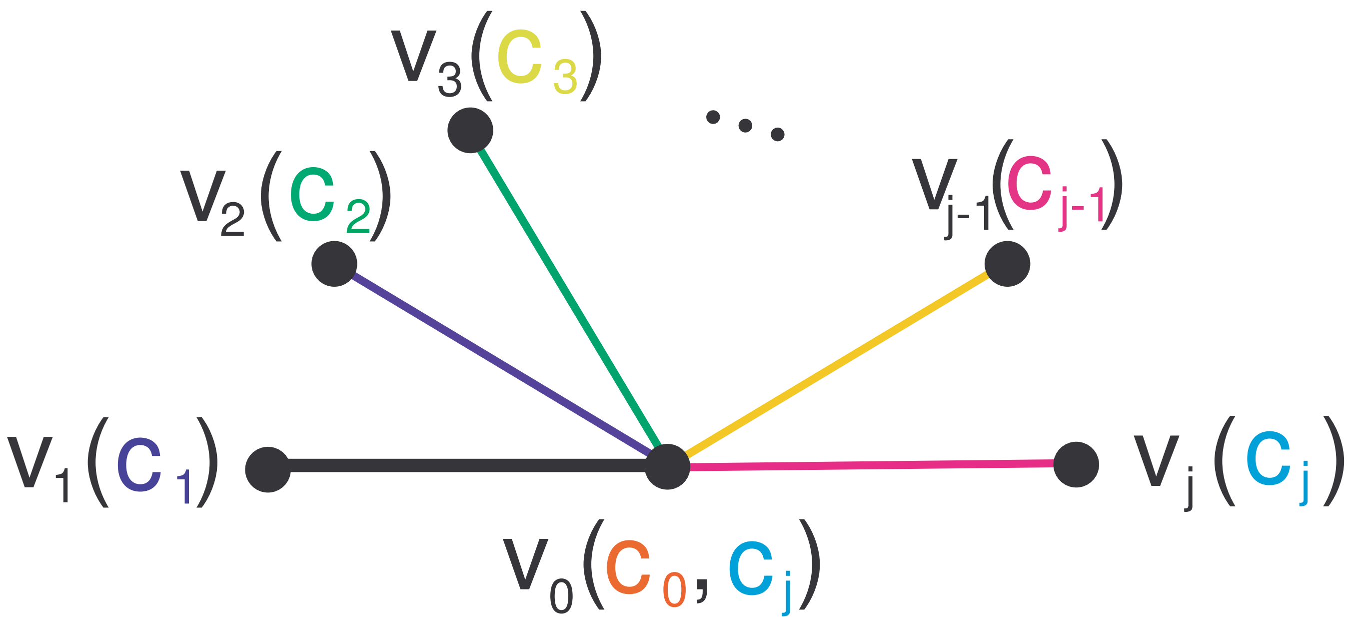
Case I: There is no edge (v_0, v) colored with c_j.

Case II: For some $2 \leq k < j$: $c_j = c_{k-1}$

$\Rightarrow$ the edge (v_0, v_k) is colored with c_j.

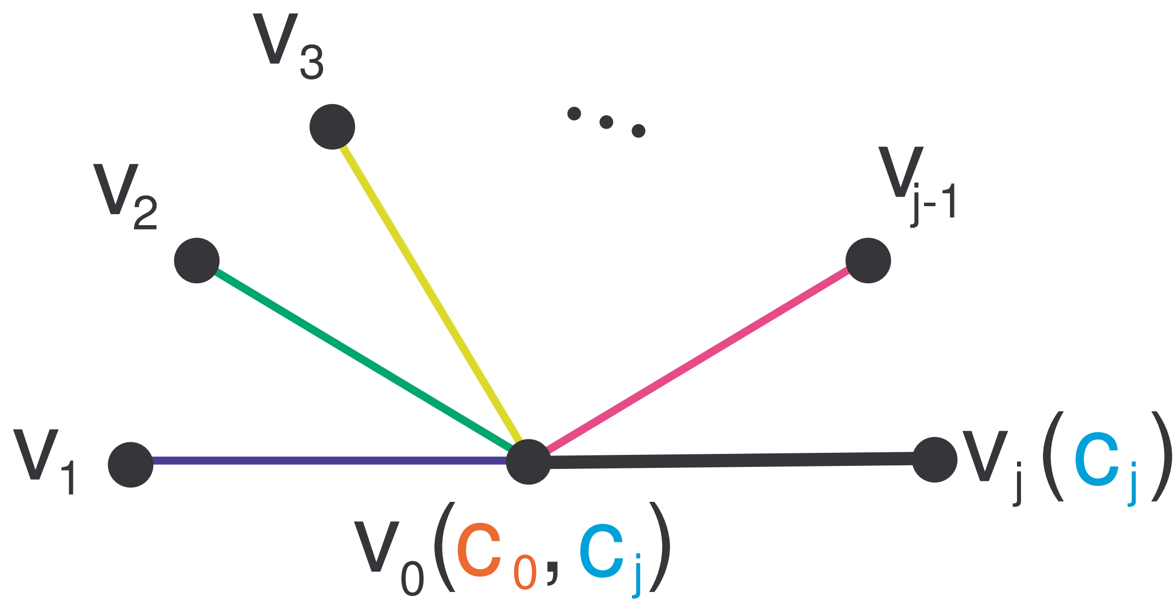
Case I

⋆ The color c_j is missing at v_0.



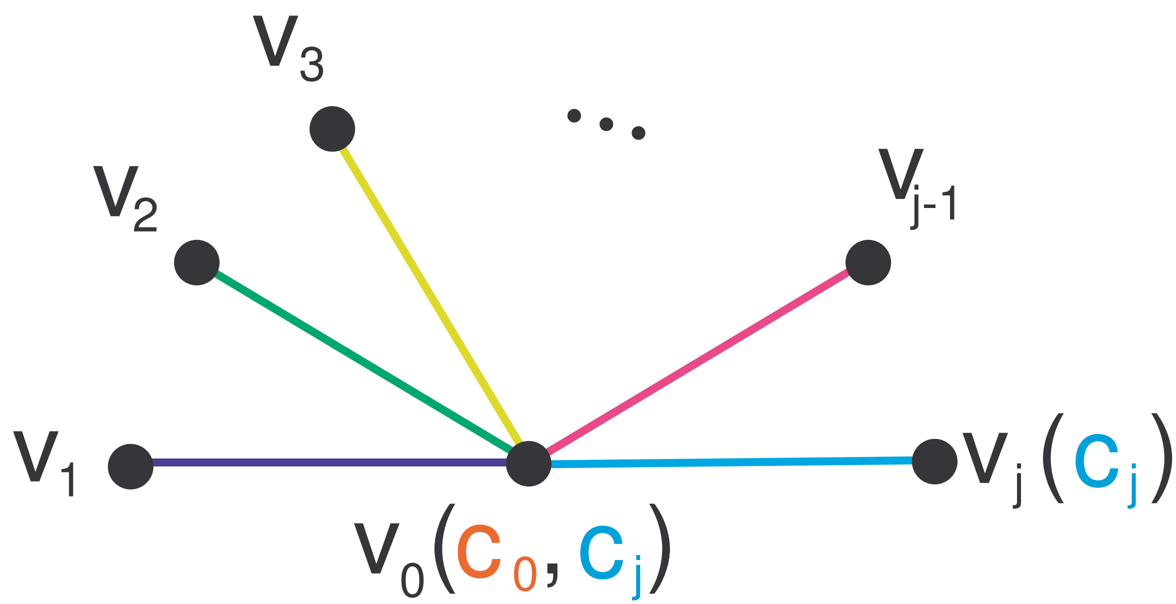
Case I

⋆ Shift colors: Color (v_0, v_i) with c_i for $1 \leq i \leq j-1$.



Case I

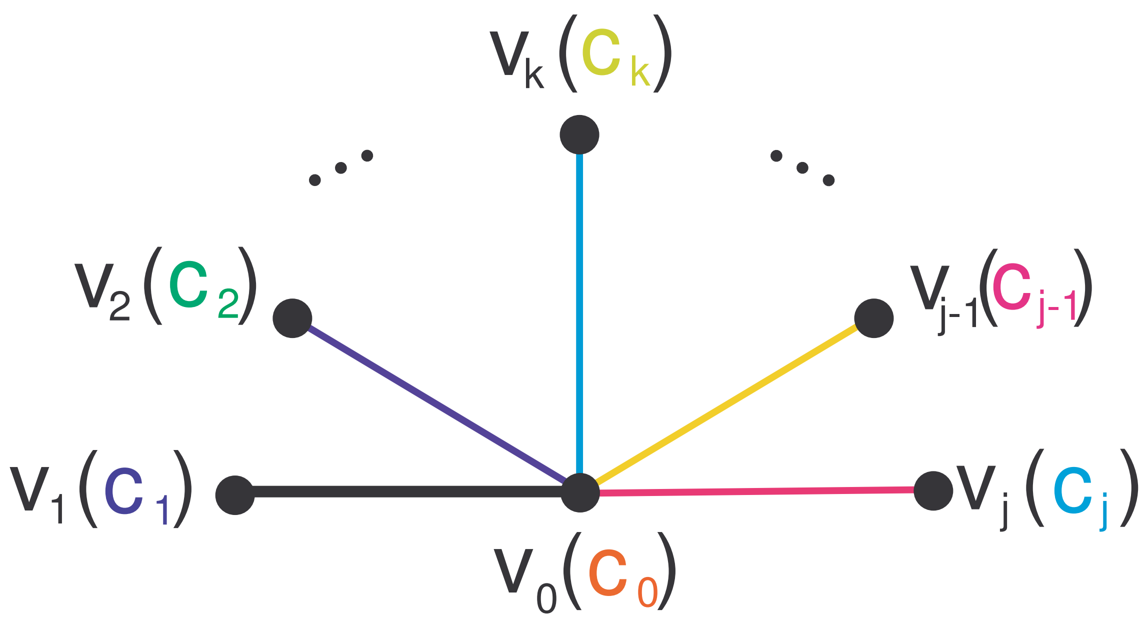
⋆ c_j is missing at both v_0 and v_j: color (v_0, v_j) with c_j.



Case II

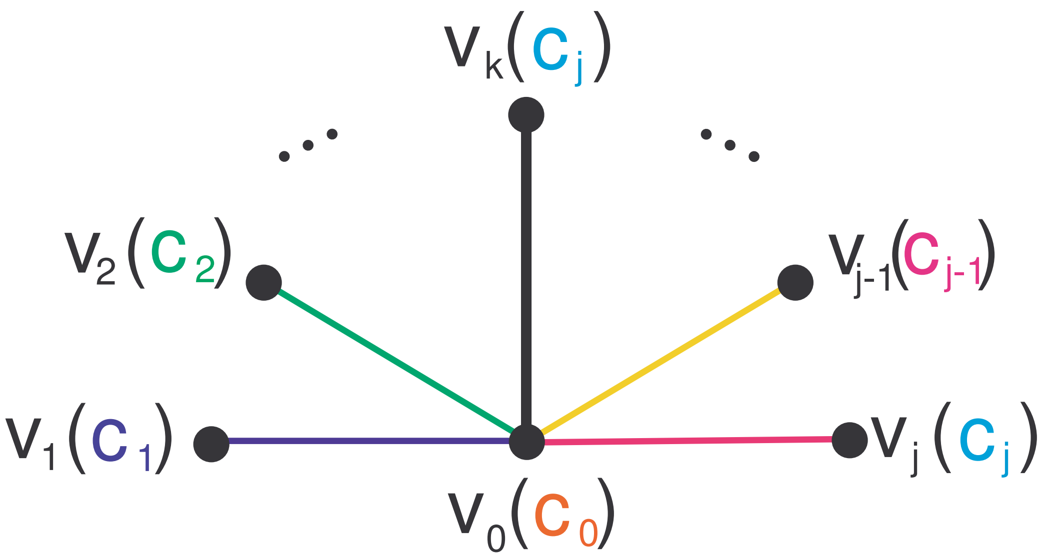
⋆ For some $2 \leq k < j$: $c_j = c_{k-1}$

$\Rightarrow (v_0, v_k)$ is colored with c_j.



Case II

- ⋆ Shift colors: color (v_0, v_i) with c_i for $1 \leq i \leq k-1$.
- ⋆ The edge (v_0, v_k) is not colored.
- ⋆ c_j is missing at both v_k and v_j.



Case II

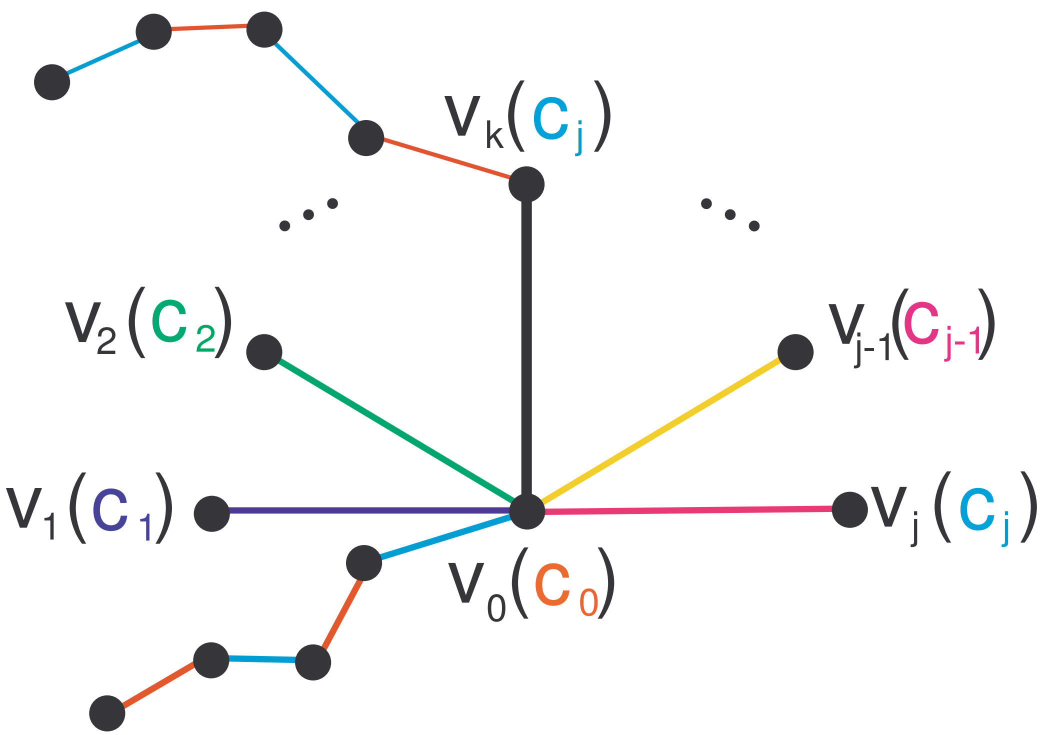
- ⋆ Consider the sub-graph $G(c_0, c_j)$ of G containing only the edges colored with c_0 and c_j.
- ⋆ $G(c_0, c_j)$ is a collection of paths and cycles; v_0, v_k, v_j are end-vertices of paths in $G(c_0, c_j)$.
- ⋆ Not all of the 3 vertices v_0, v_k, v_j are in the same connected component of $G(c_0, c_j)$.

Case II.I v_0 and v_k are in different connected components of $G(c_0, c_j)$.

Case II.II v_0 and v_j are in different connected components of $G(c_0, c_j)$.

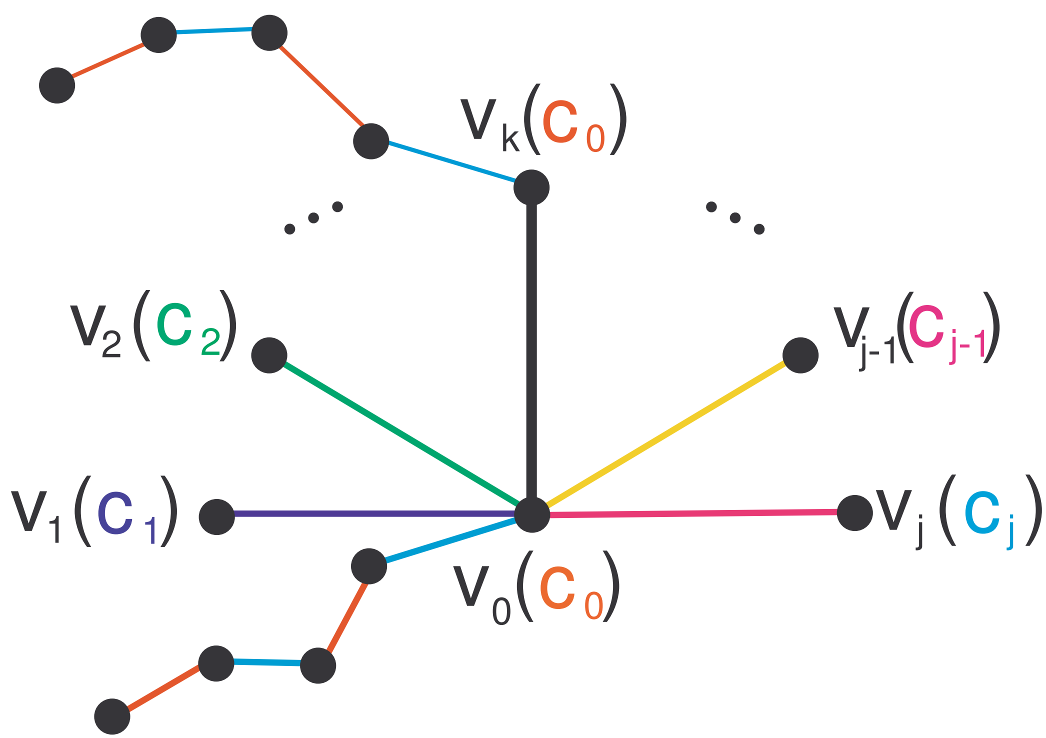
Case II.I

$\star$ v_0, v_k are in different connected components of $G(c_0, c_j)$.



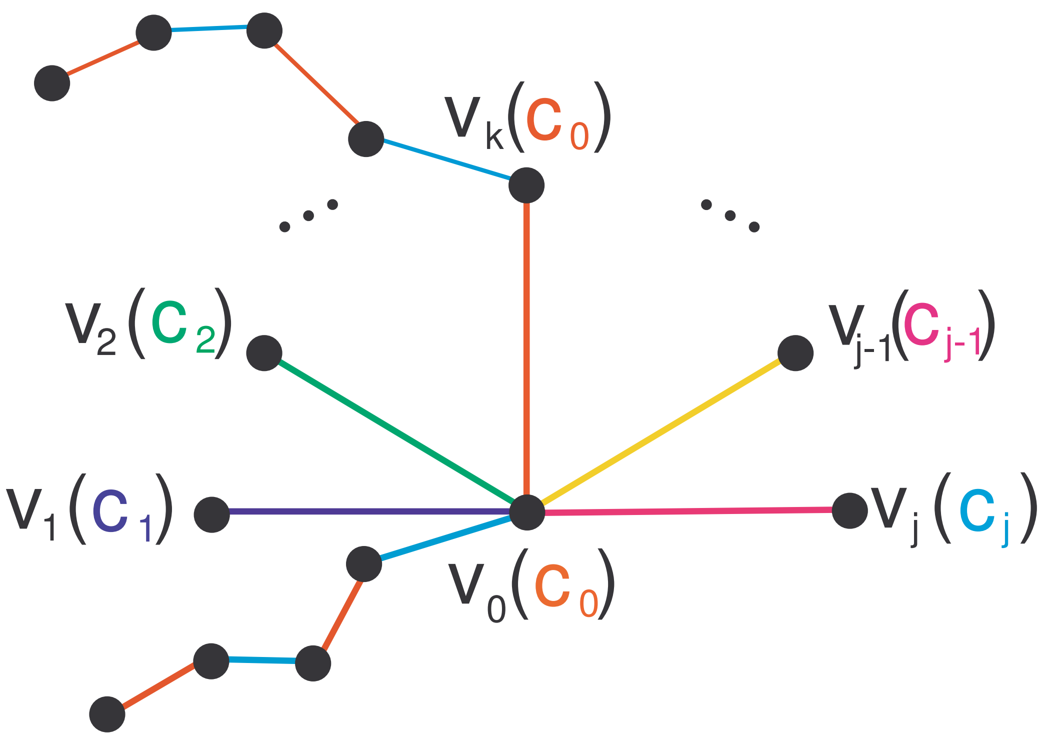
Case II.I

⋆ Exchange between c_0 and c_j in the v_k-path in $G(c_0, c_j)$.



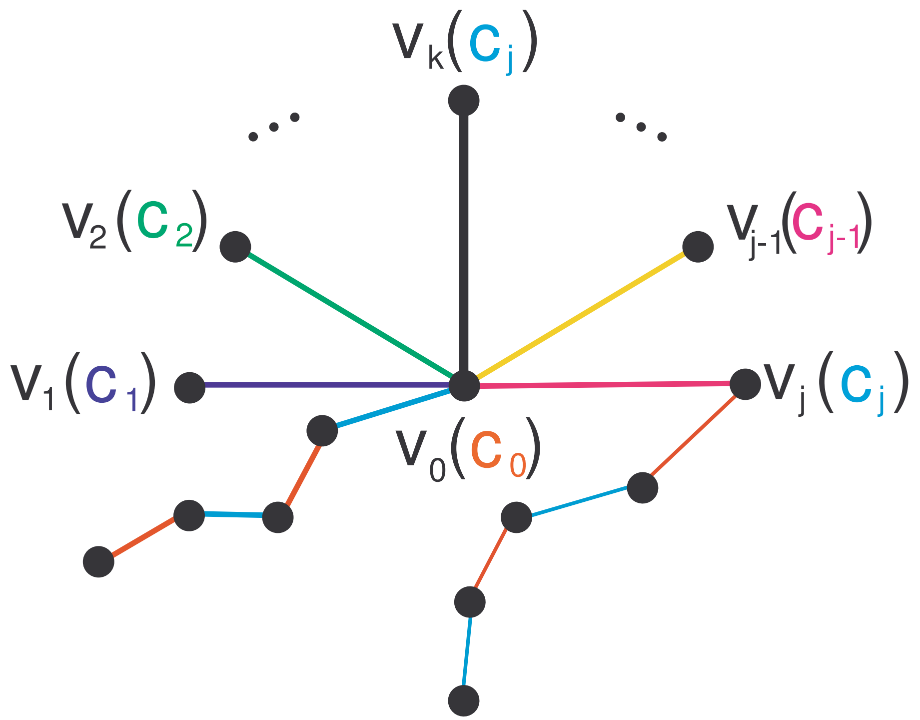
Case II.I

⋆ c_0 is missing at both v_0 and v_k: color (v_0, v_k) with c_0.



Case II.II

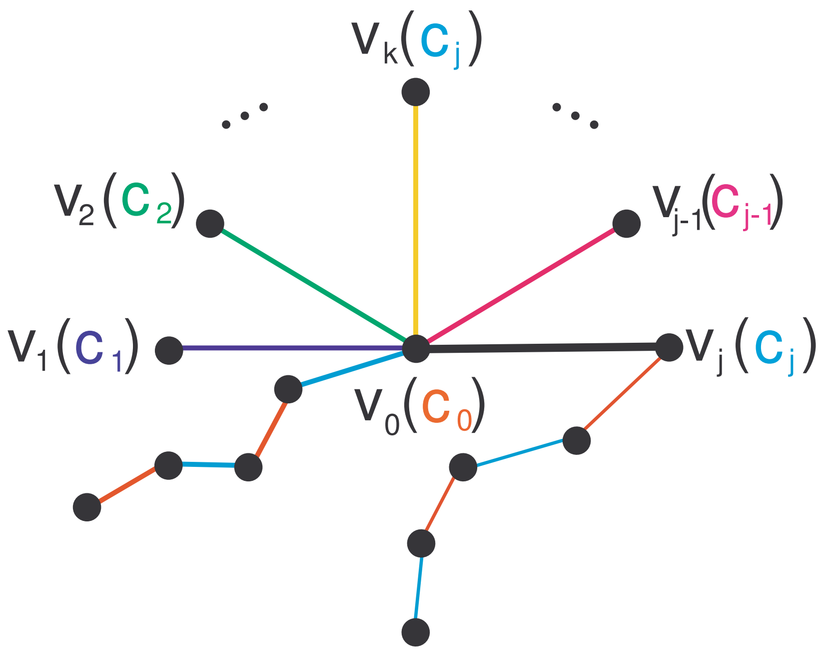
- $\star$ v_0, v_j are in different connected components of $G(c_0, c_j)$.



Case II.II

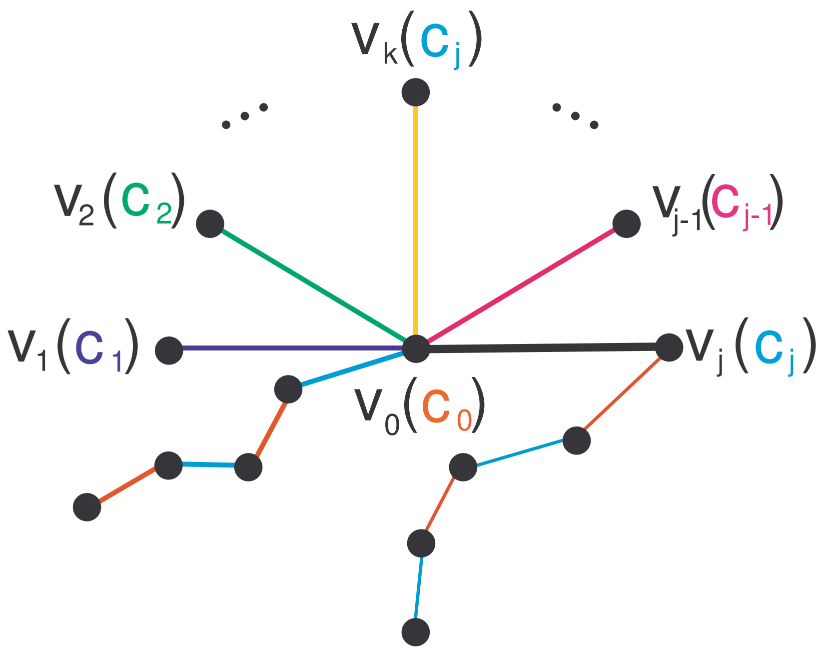
⋆ Shift colors: Color (v_0, v_i) with c_i for $k \leq i \leq j-1$.

⋆ The edge (v_0, v_j) is not colored.



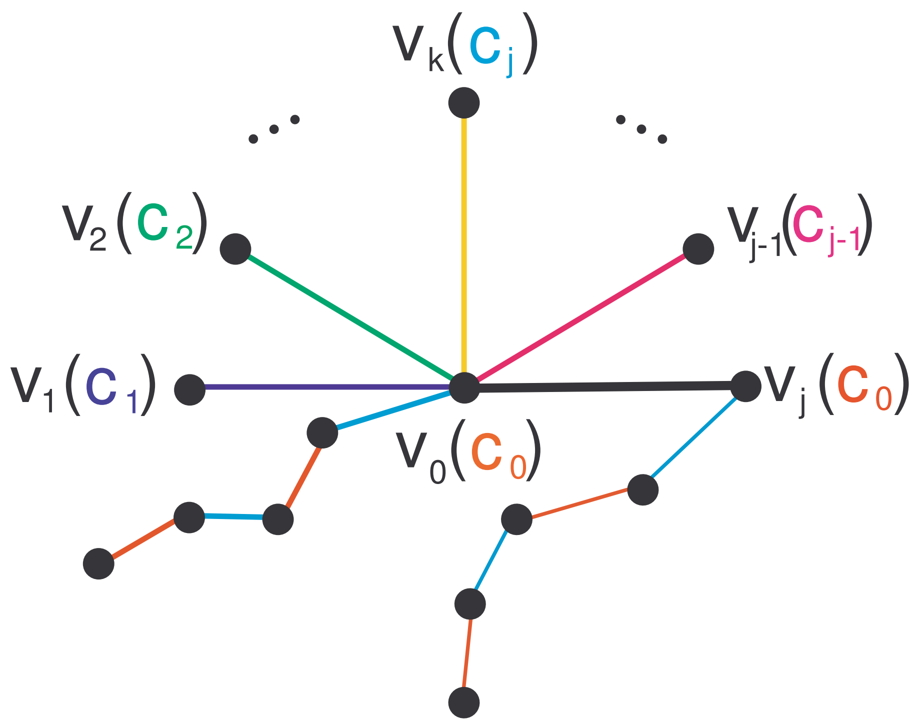
Case II.II

- ⋆ The shift process does not involve c_0 and $c_j \Rightarrow v_0$ and v_j are still in different connected components of $G(c_0, c_j)$.
- ⋆ c_j is missing at v_j and c_0 is missing in v_0.



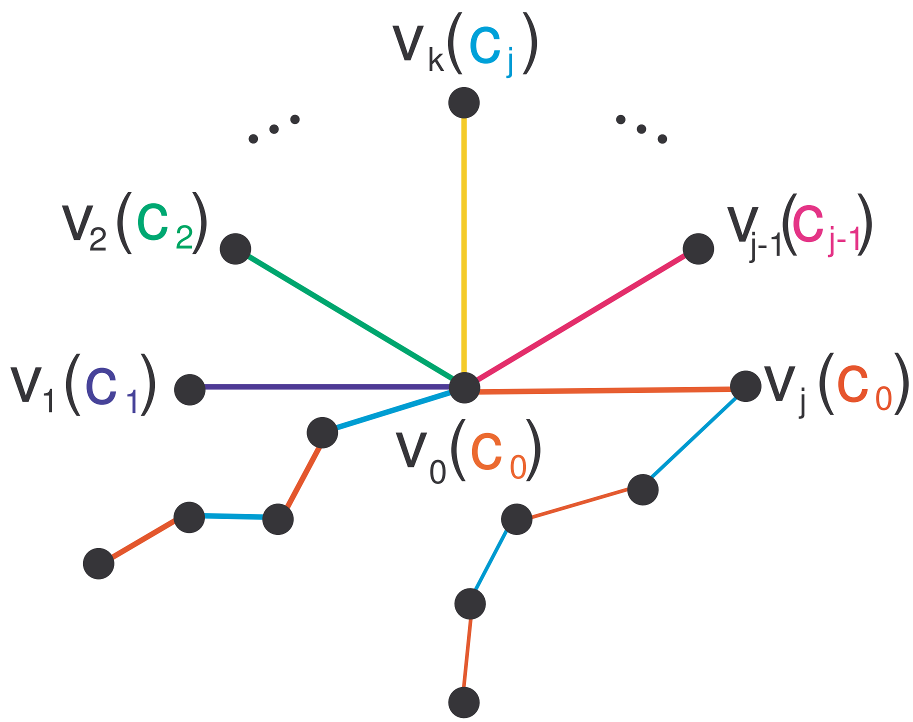
Case II.II

⋆ Exchange between c_0 and c_j in the v_j-path in $G(c_0, c_j)$.



Case II.II

$\star$ c_0 is missing at both v_0 and v_j: color (v_0, v_j) with c_0.



Complexity

- $\star$ m edges to color.
- $\star$ $O(\Delta^2)$ to find $v_1, \ldots, v_j$.
- $\star$ $O(n)$ to exchange colors.
- $\star$ Overall, $O(nm + \Delta^2 m)$ complexity.