Basic Graphs

Alexander Golovnev

Outline

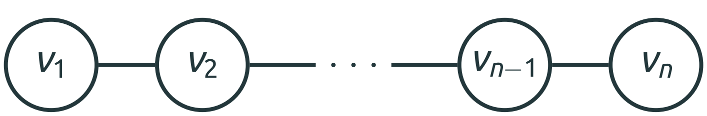
Paths, Cycles and Complete Graphs

Trees

Bipartite Graphs

Path Graphs

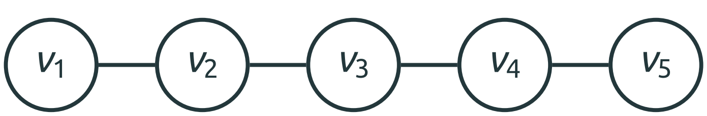
The Path Graph P_n, $n \geq 2$, consists of n vertices $v_1, \ldots, v_n$ and $n-1$ edges $\{v_1, v_2\}, \ldots, \{v_{n-1}, v_n\}$



Path Graphs

The Path Graph P_n, $n \geq 2$, consists of n vertices $v_1, \ldots, v_n$ and $n-1$ edges $\{v_1, v_2\}, \ldots, \{v_{n-1}, v_n\}$

The Graph P_5



Path Graphs

The Path Graph P_n, $n \geq 2$, consists of n vertices $v_1, \ldots, v_n$ and $n-1$ edges $\{v_1, v_2\}, \ldots, \{v_{n-1}, v_n\}$

The Graph P_2

Path Graphs

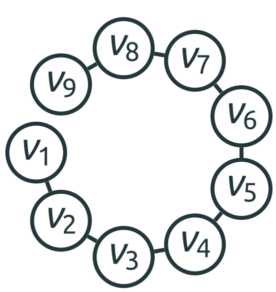
The Path Graph P_n, $n \geq 2$, consists of n vertices $v_1, \ldots, v_n$ and $n-1$ edges $\{v_1, v_2\}, \ldots, \{v_{n-1}, v_n\}$

The Graph P_9

Path Graphs

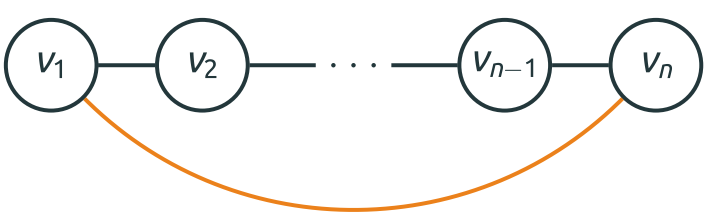
The Path Graph P_n, $n \geq 2$, consists of n vertices $v_1, \ldots, v_n$ and $n-1$ edges $\{v_1, v_2\}, \ldots, \{v_{n-1}, v_n\}$

The Graph P_9



Cycle Graphs

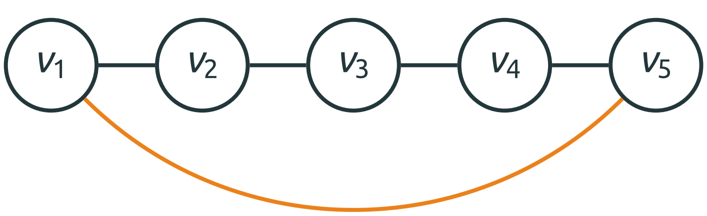
The Cycle Graph C_n, $n \geq 3$, consists of n vertices $v_1, \ldots, v_n$ and n edges $\{v_1, v_2\}, \ldots, \{v_{n-1}, v_n\}$, $\{v_n, v_1\}$



Cycle Graphs

The Cycle Graph C_n, $n \geq 3$, consists of n vertices $v_1, \ldots, v_n$ and n edges $\{v_1, v_2\}, \ldots, \{v_{n-1}, v_n\}$, $\{v_n, v_1\}$

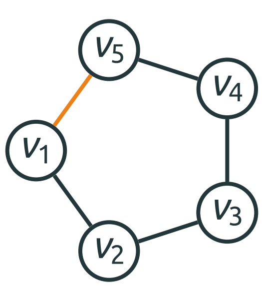
The Graph C_5



Cycle Graphs

The Cycle Graph C_n, $n \geq 3$, consists of n vertices $v_1, \ldots, v_n$ and n edges $\{v_1, v_2\}, \ldots, \{v_{n-1}, v_n\}$, $\{v_n, v_1\}$

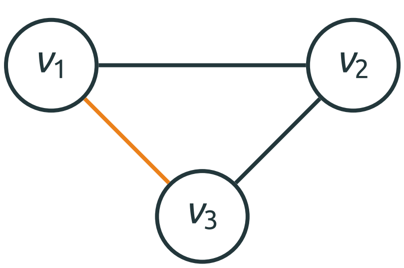
The Graph C_5



Cycle Graphs

The Cycle Graph C_n, $n \geq 3$, consists of n vertices $v_1, \ldots, v_n$ and n edges $\{v_1, v_2\}, \ldots, \{v_{n-1}, v_n\}$, $\{v_n, v_1\}$

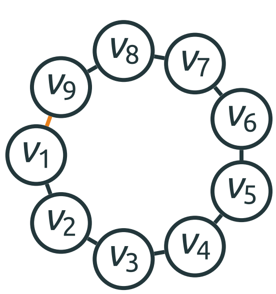
The Graph C_3



Cycle Graphs

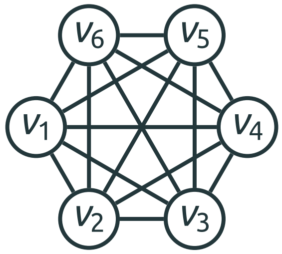
The Cycle Graph C_n, $n \geq 3$, consists of n vertices $v_1, \ldots, v_n$ and n edges $\{v_1, v_2\}, \ldots, \{v_{n-1}, v_n\}$, $\{v_n, v_1\}$

The Graph C_9



Complete Graphs

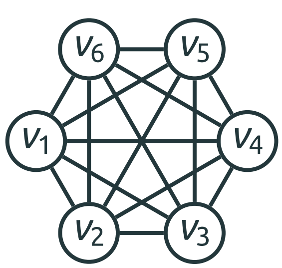
The Complete Graph (Clique) K_n, $n \geq 2$, contains n vertices $v_1, \ldots, v_n$ and all edges between them ($n(n-1)/2$ edges)



Complete Graphs

The Complete Graph (Clique) K_n, $n \geq 2$, contains n vertices $v_1, \ldots, v_n$ and all edges between them ($n(n-1)/2$ edges)

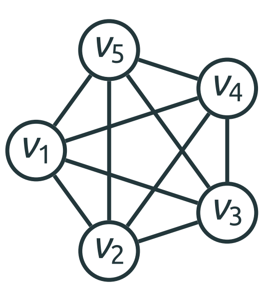
The Graph K_6



Complete Graphs

The Complete Graph (Clique) K_n, $n \geq 2$, contains n vertices $v_1, \ldots, v_n$ and all edges between them ($n(n-1)/2$ edges)

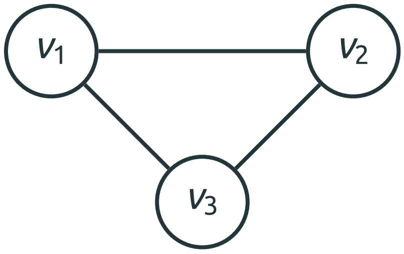
The Graph K_5



Complete Graphs

The Complete Graph (Clique) K_n, $n \geq 2$, contains n vertices $v_1, \ldots, v_n$ and all edges between them ($n(n-1)/2$ edges)

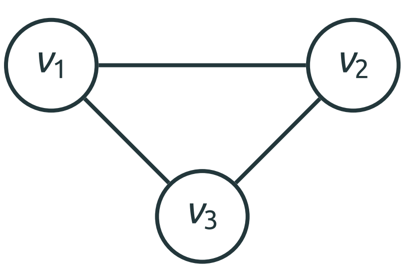
The Graph K_3



Complete Graphs

The Complete Graph (Clique) K_n, $n \geq 2$, contains n vertices $v_1, \ldots, v_n$ and all edges between them ($n(n-1)/2$ edges)

The Graph $K_3 = C_3$



Complete Graphs

The Complete Graph (Clique) K_n, $n \geq 2$, contains n vertices $v_1, \ldots, v_n$ and all edges between them ($n(n-1)/2$ edges)

The Graph K_2

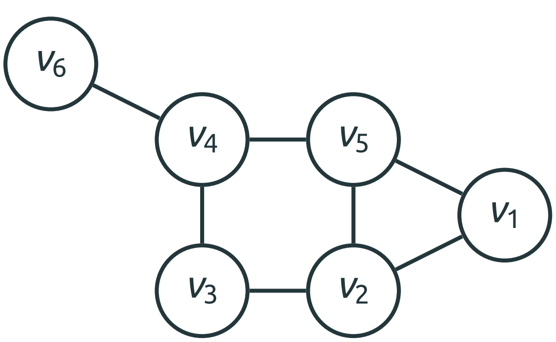
Complete Graphs

The Complete Graph (Clique) K_n, $n \geq 2$, contains n vertices $v_1, \ldots, v_n$ and all edges between them ($n(n-1)/2$ edges)

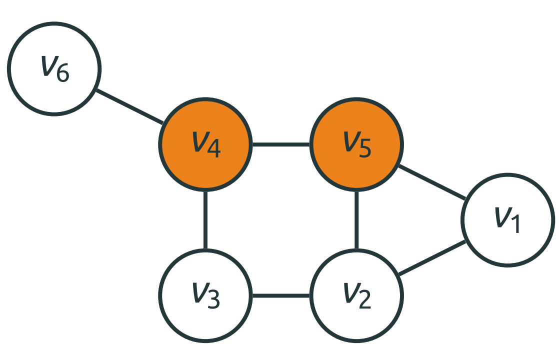
The Graph $K_2 = P_2$

Outline

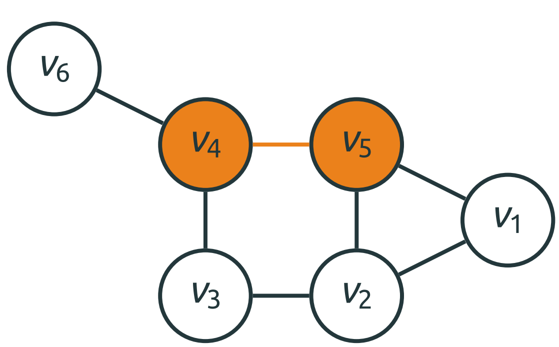
Trees



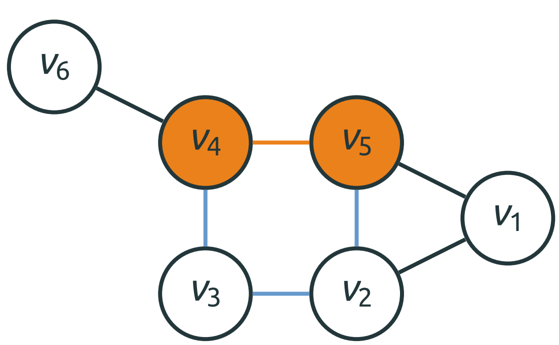
Trees



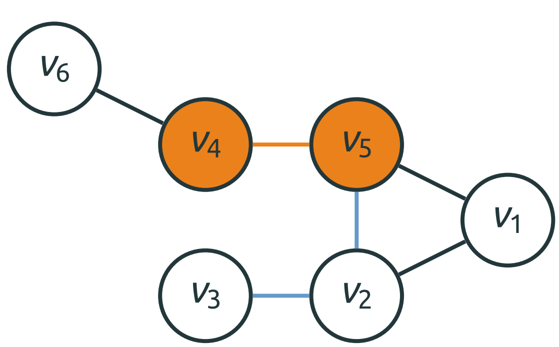
Trees



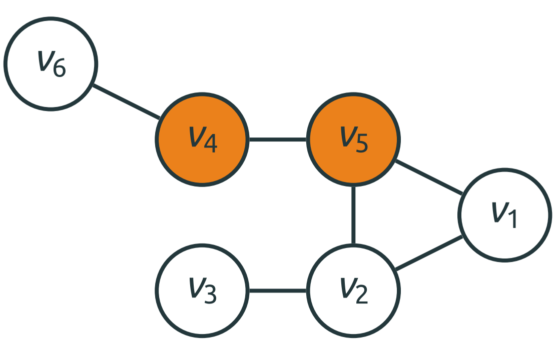
Trees



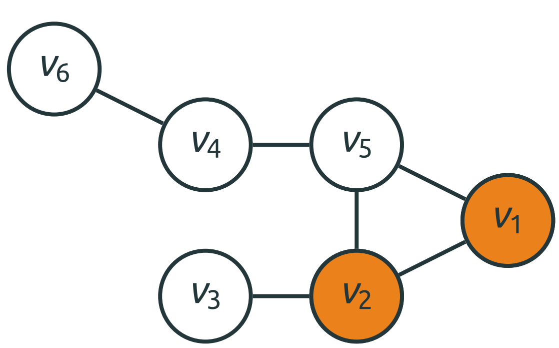
Trees



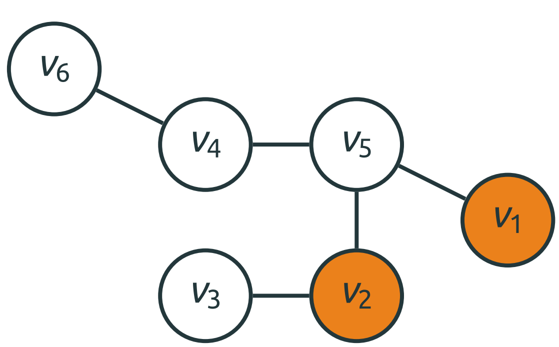
Trees



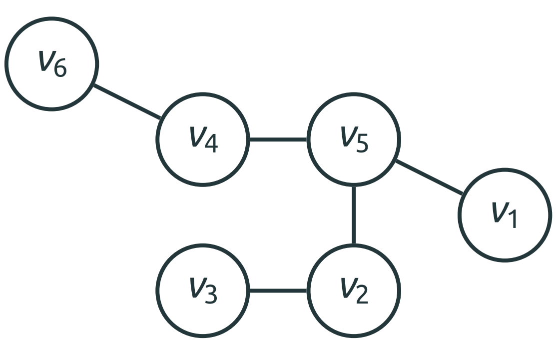
Trees



Trees



Trees



Definition

- A tree is a connected graph without cycles

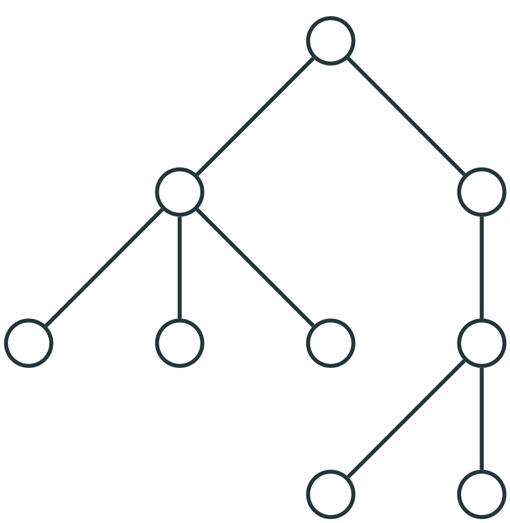
Definition

- A tree is a connected graph without cycles
- A tree is a connected graph on n vertices with $n - 1$ edges

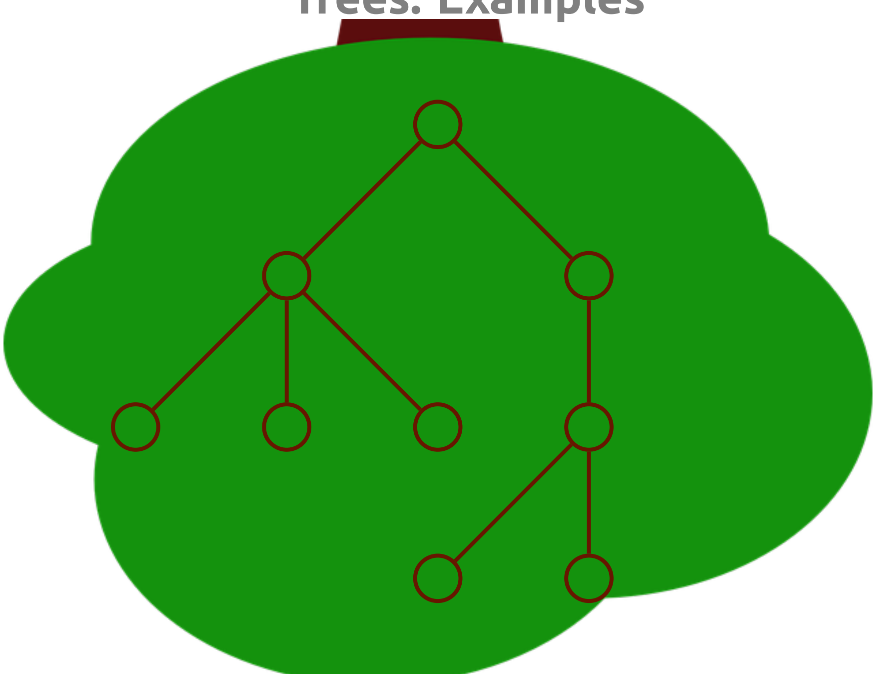
Definition

- A tree is a connected graph without cycles
- A tree is a connected graph on n vertices with $n - 1$ edges
- A graph is a tree if and only if there is a unique simple path between any pair of its vertices

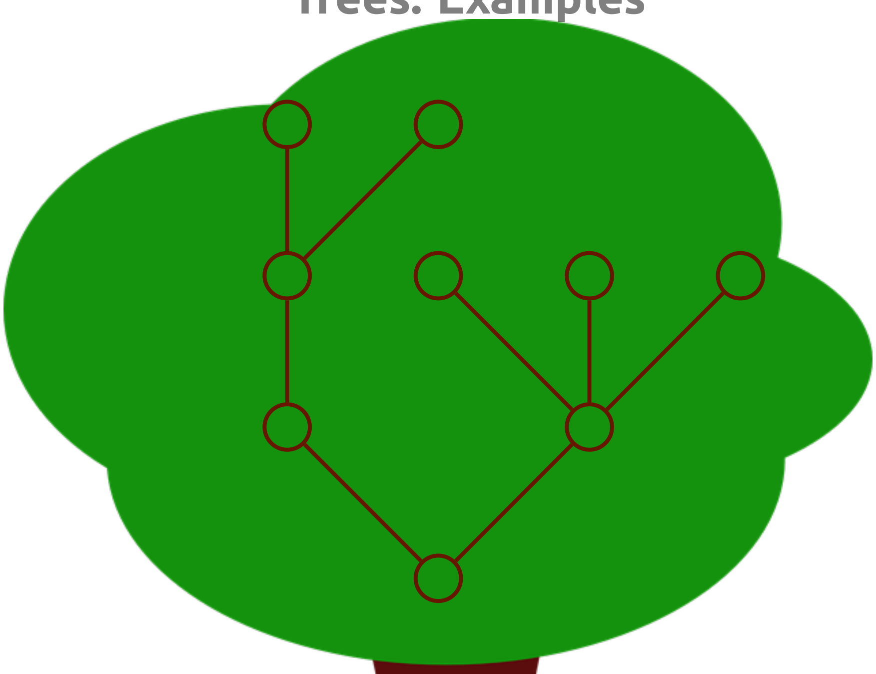
Trees: Examples



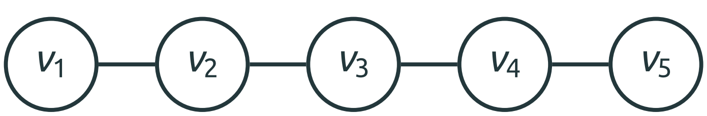
Trees: Examples



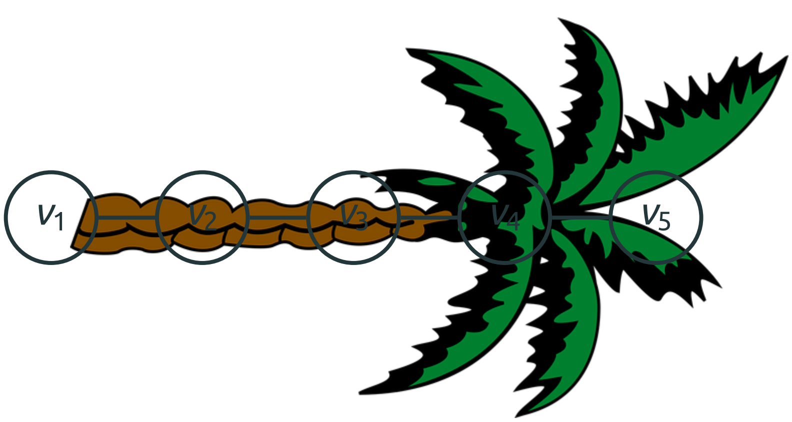
Trees: Examples



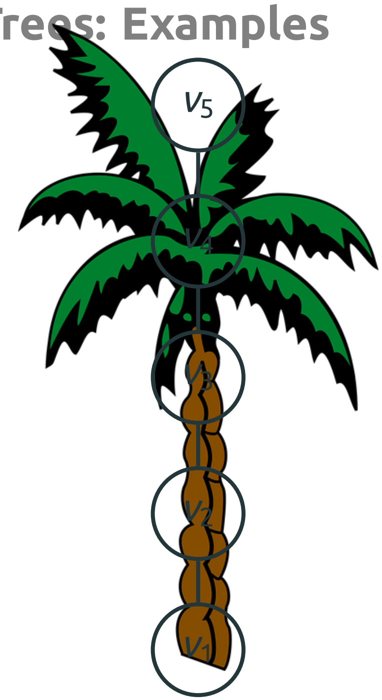
Trees: Examples



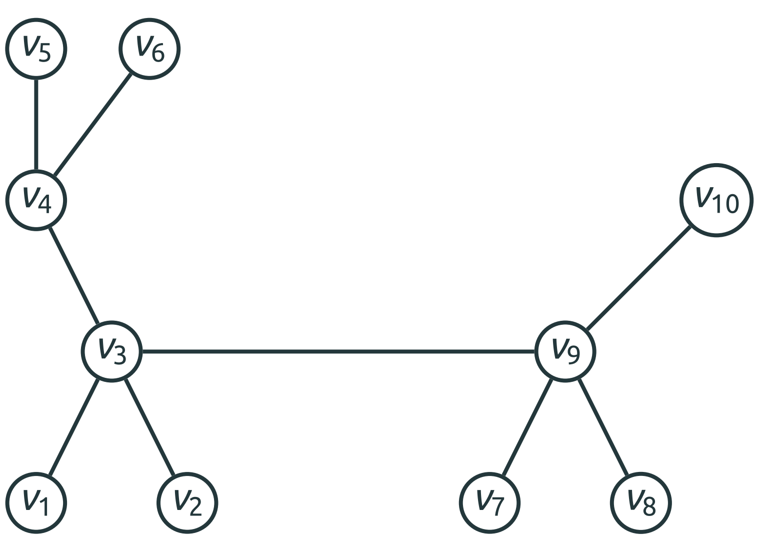
Trees: Examples



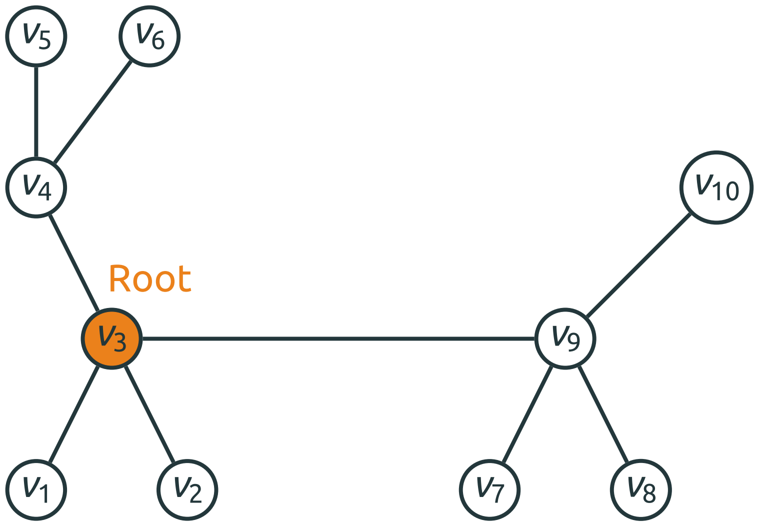
Trees: Examples



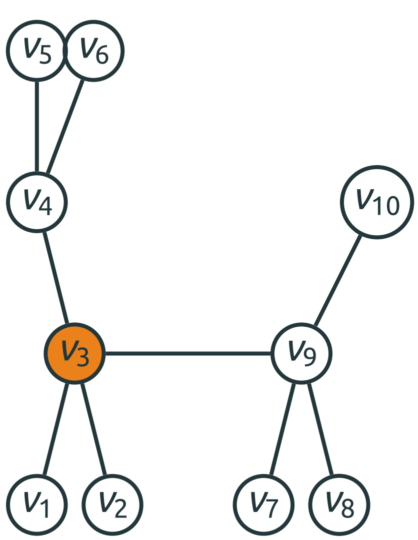
Drawing a Tree



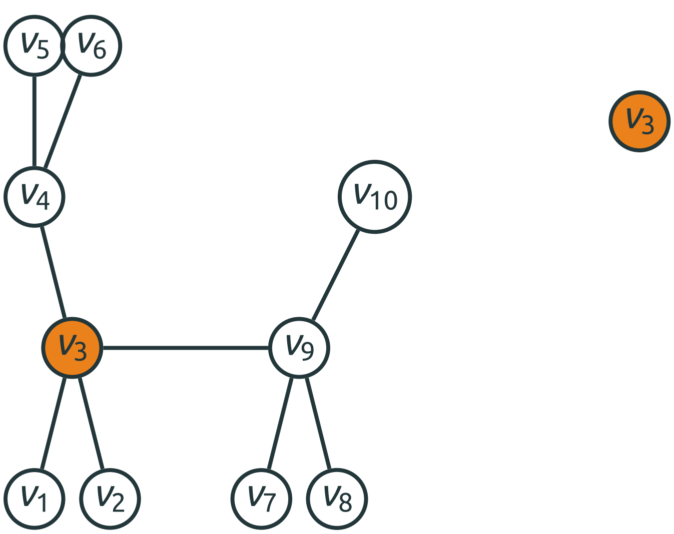
Drawing a Tree



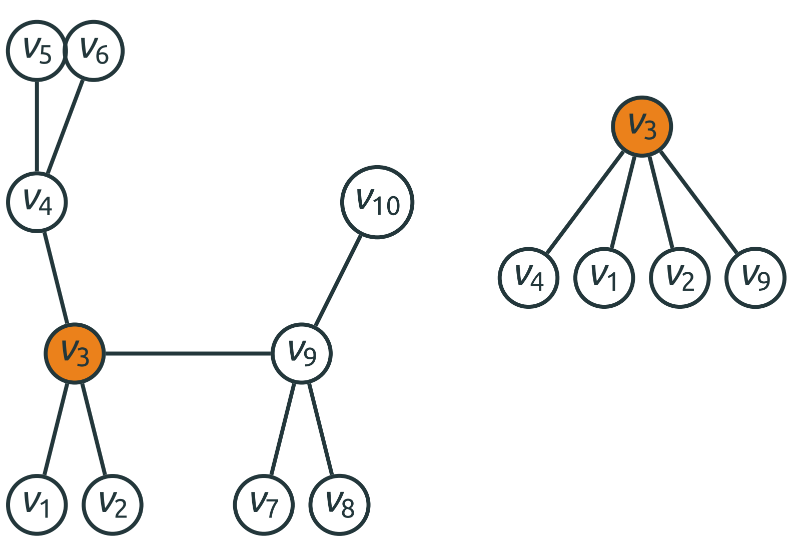
Drawing a Tree



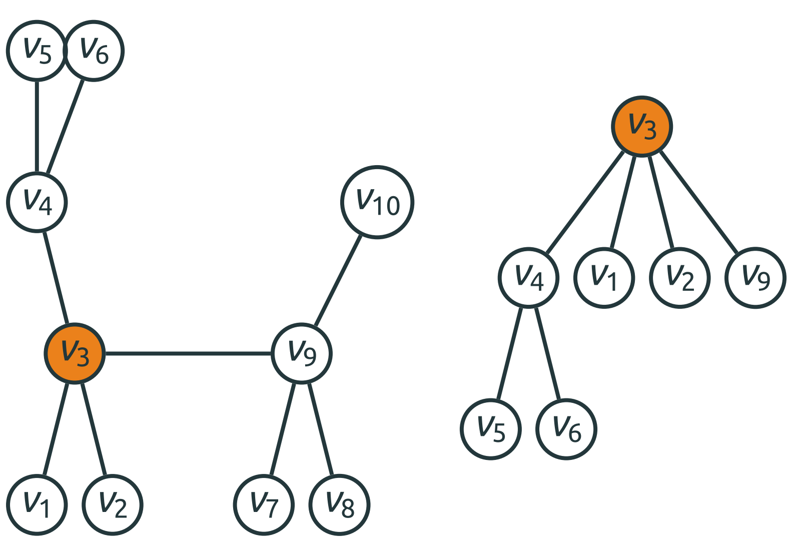
Drawing a Tree



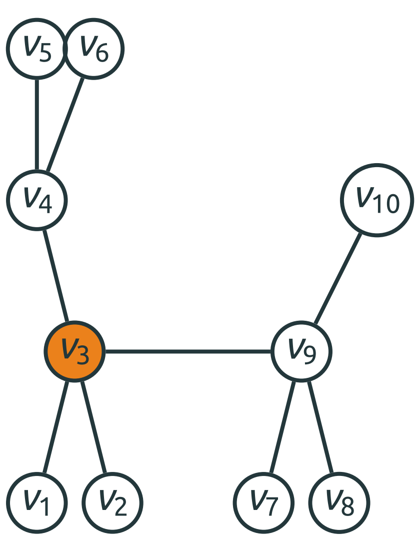
Drawing a Tree

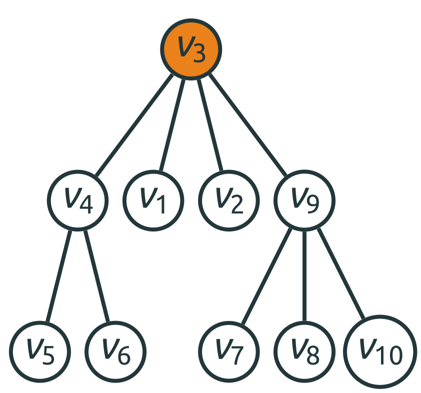


Drawing a Tree

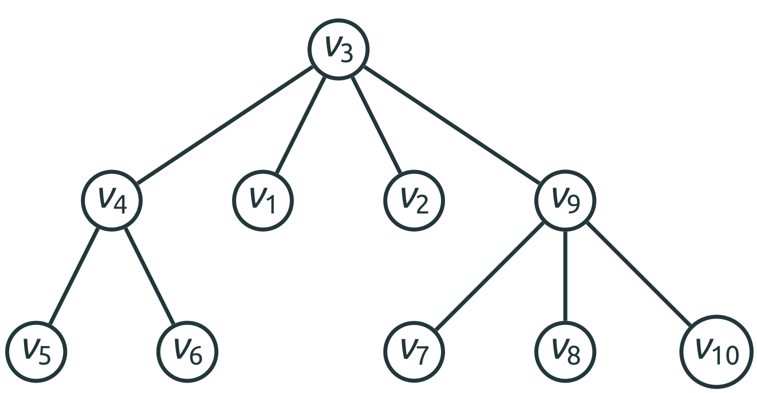


Drawing a Tree



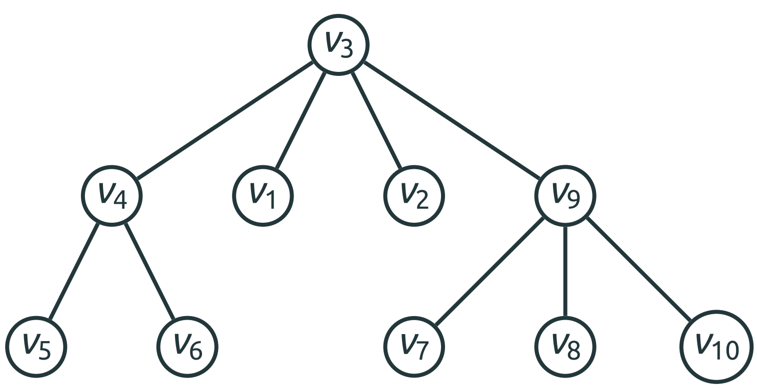


Drawing a Tree

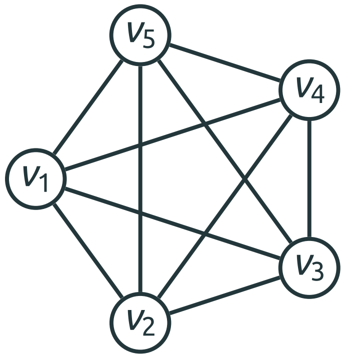


Drawing a Tree

Connected; the number of edges is $n - 1$

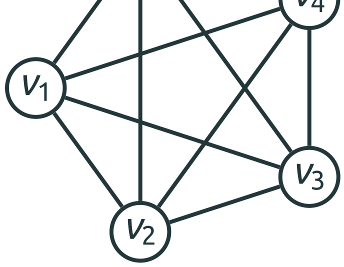


Make a Tree



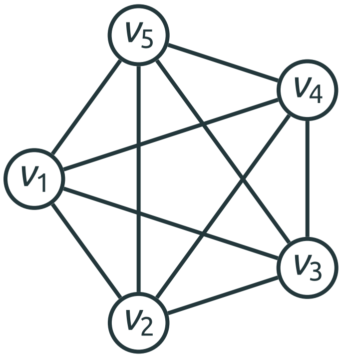
Make a Tree

Remove any edge, keeping the graph connected



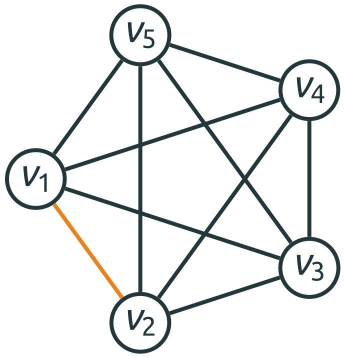
Make a Tree

Remove any edge, keeping the graph connected
Stop when only $n - 1$ edges left



Make a Tree

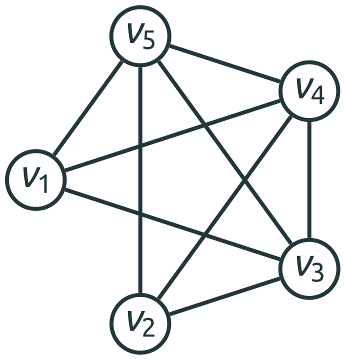
Remove any edge, keeping the graph connected
Stop when only $n - 1$ edges left



Make a Tree

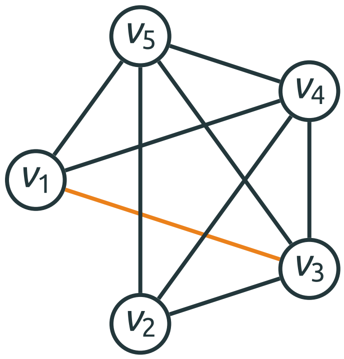
Remove any edge, keeping the graph connected

Stop when only $n-1$ edges left



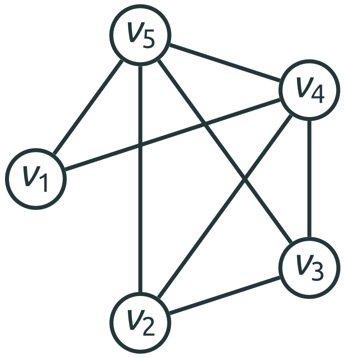
Make a Tree

Remove any edge, keeping the graph connected
Stop when only $n-1$ edges left



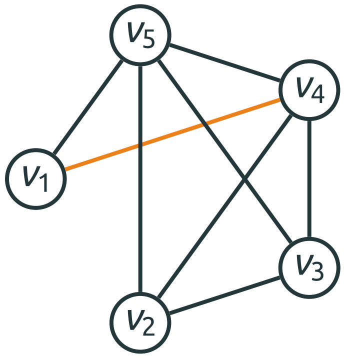
Make a Tree

Remove any edge, keeping the graph connected
Stop when only $n - 1$ edges left



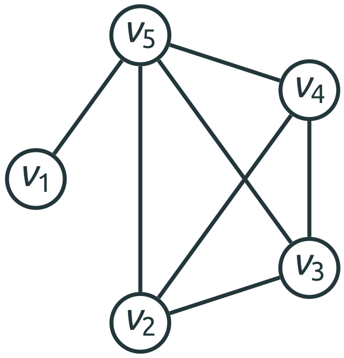
Make a Tree

Remove any edge, keeping the graph connected
Stop when only $n - 1$ edges left



Make a Tree

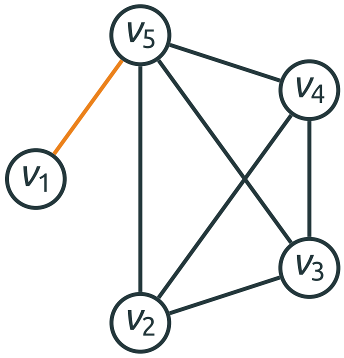
Remove any edge, keeping the graph connected
Stop when only $n - 1$ edges left



Make a Tree

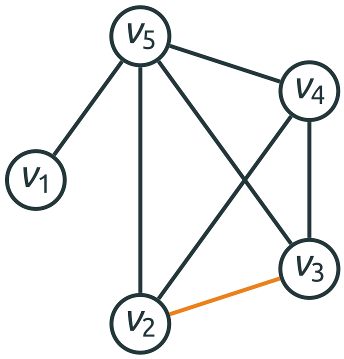
Remove any edge, keeping the graph connected

Stop when only $n - 1$ edges left



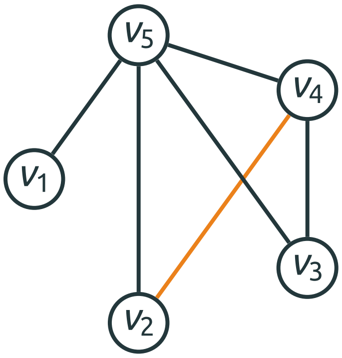
Make a Tree

Remove any edge, keeping the graph connected
Stop when only $n - 1$ edges left



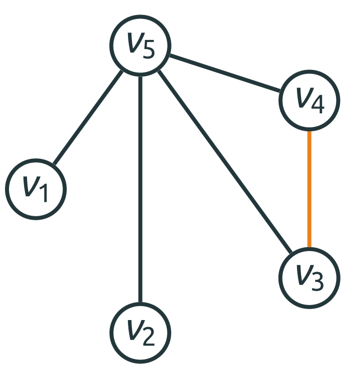
Make a Tree

Remove any edge, keeping the graph connected
Stop when only $n-1$ edges left



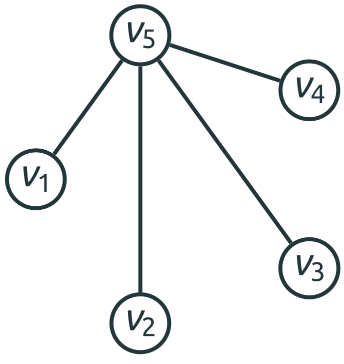
Make a Tree

Remove any edge, keeping the graph connected
Stop when only $n-1$ edges left



Make a Tree

Remove any edge, keeping the graph connected
Stop when only $n-1$ edges left



Outline

Paths, Cycles and Complete Graphs

Trees

Bipartite Graphs

Bipartite Graphs

- A graph G is Bipartite if its vertices can be partitioned into two disjoint sets L and R such that

Bipartite Graphs

- A graph G is Bipartite if its vertices can be partitioned into two disjoint sets L and R such that
  - Every edge of G connects a vertex in L to a vertex in R

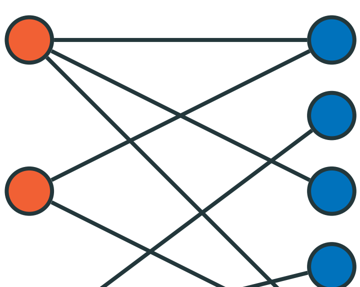
Bipartite Graphs

- A graph G is Bipartite if its vertices can be partitioned into two disjoint sets L and R such that
  - Every edge of G connects a vertex in L to a vertex in R
  - I.e., no edge connects two vertices from the same part

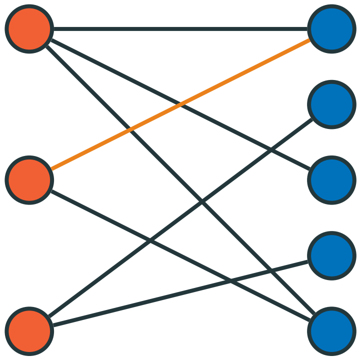
Bipartite Graphs

- A graph G is Bipartite if its vertices can be partitioned into two disjoint sets L and R such that
    - Every edge of G connects a vertex in L to a vertex in R
    - I.e., no edge connects two vertices from the same part
- L and R are called the parts of G

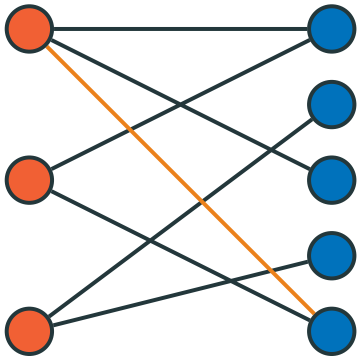
Bipartite Graphs: Examples



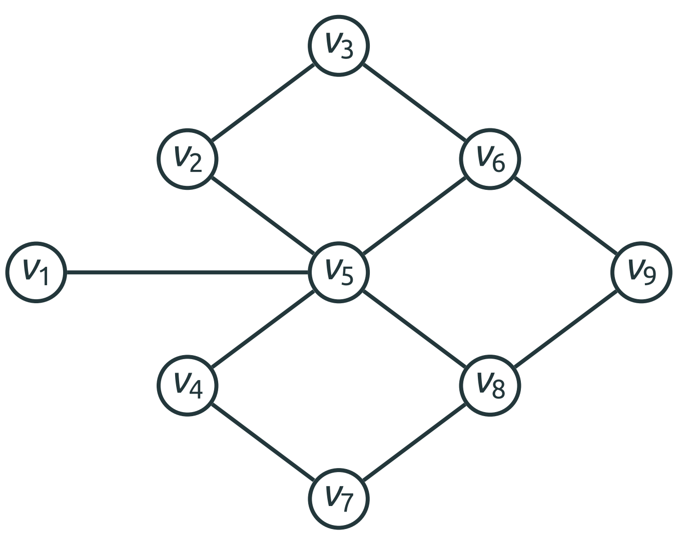
Bipartite Graphs: Examples



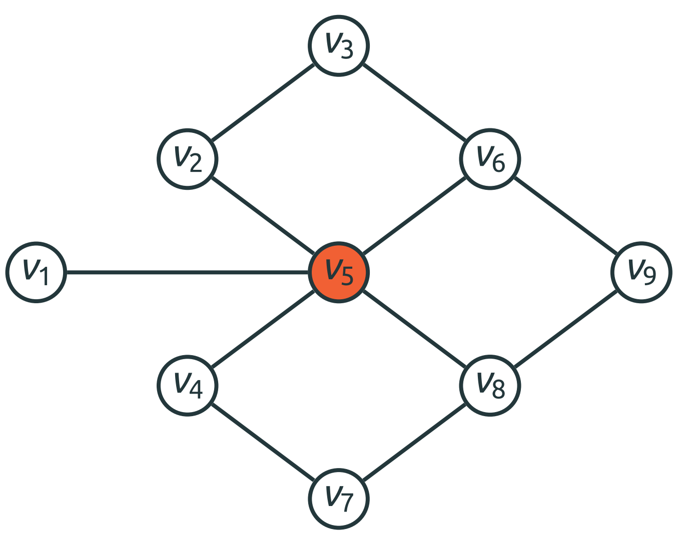
Bipartite Graphs: Examples



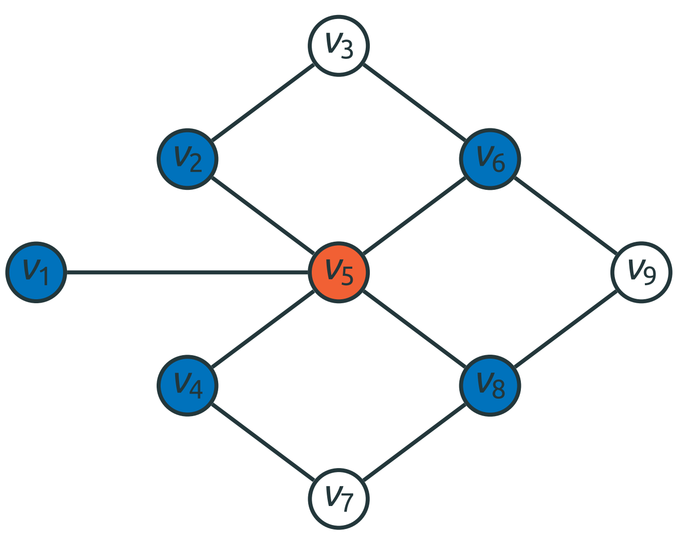
Bipartite Graphs: Examples



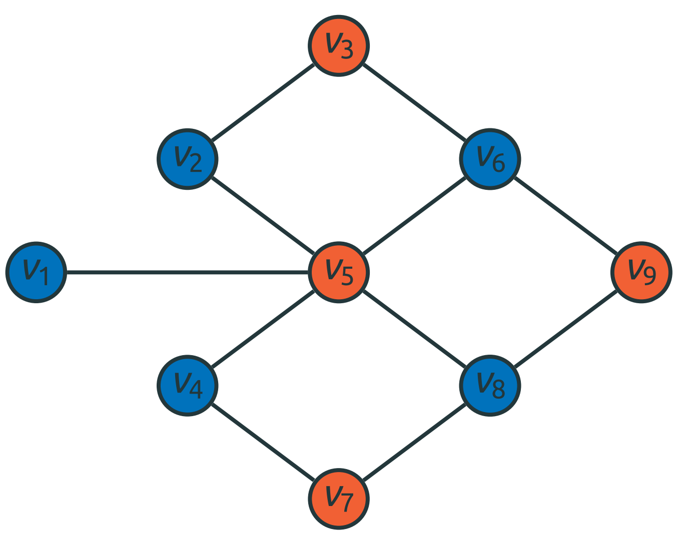
Bipartite Graphs: Examples



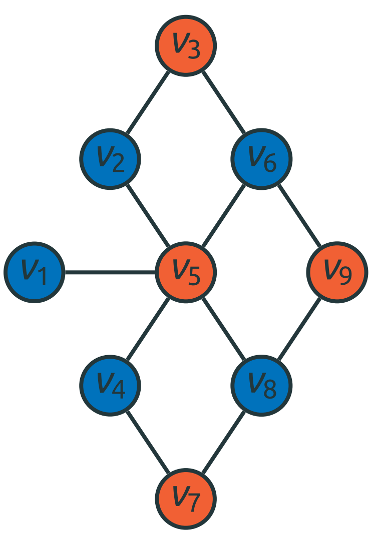
Bipartite Graphs: Examples



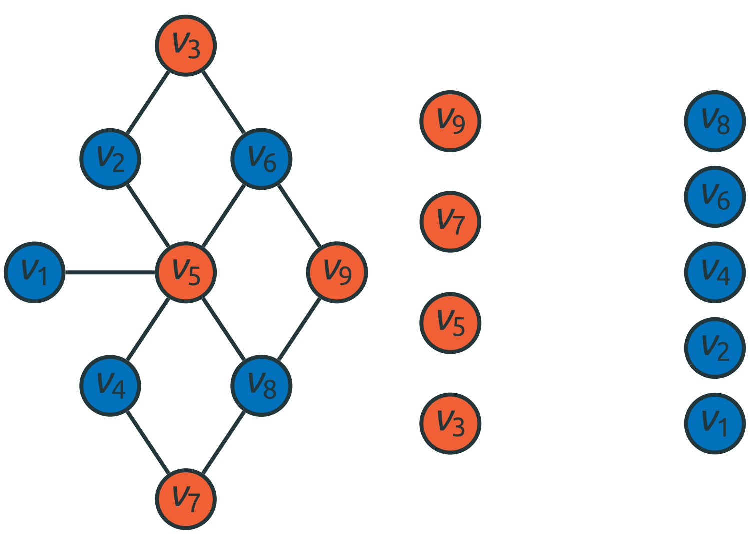
Bipartite Graphs: Examples



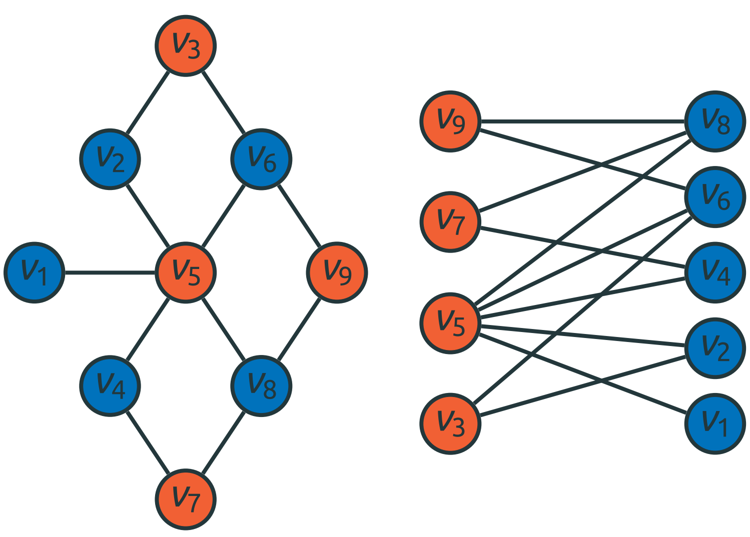
Bipartite Graphs: Examples



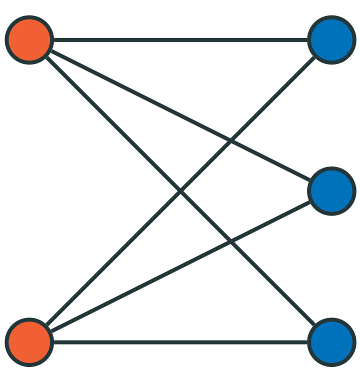
Bipartite Graphs: Examples



Bipartite Graphs: Examples

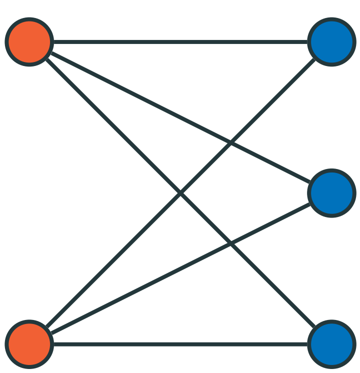


Complete Bipartite Graphs



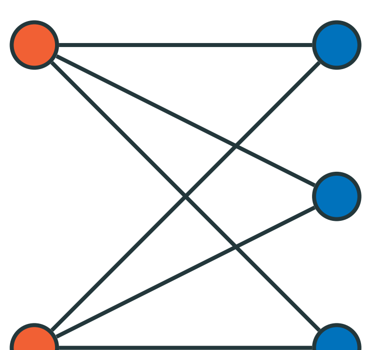
Complete Bipartite Graphs

Complete bipartite graph

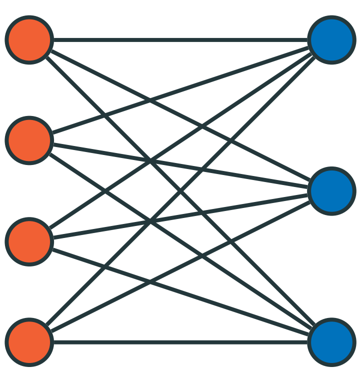


Complete Bipartite Graphs

Complete bipartite graph $K_{2,3}$

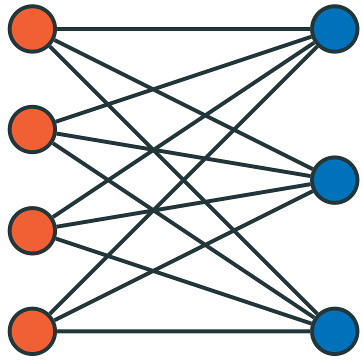


Complete Bipartite Graphs



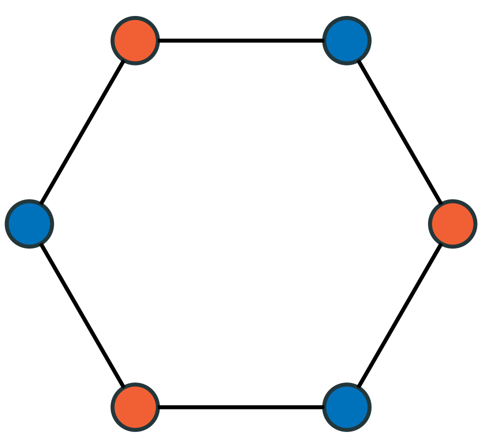
Complete Bipartite Graphs

Complete bipartite graph $K_{4,3}$



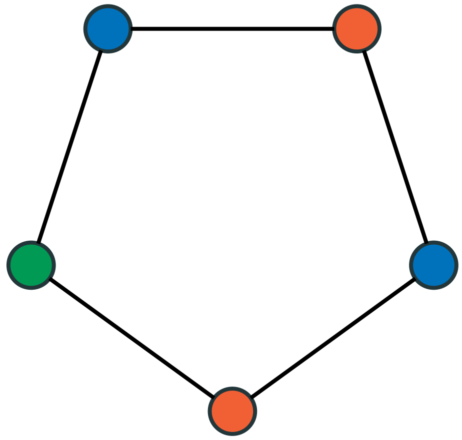
Cycle Graphs

For even n, C_n is bipartite

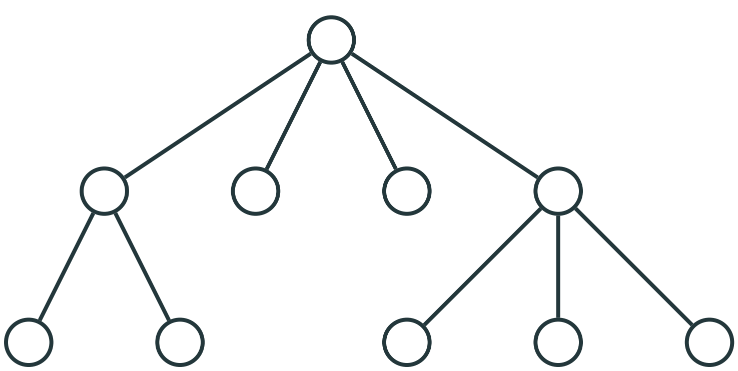


Cycle Graphs

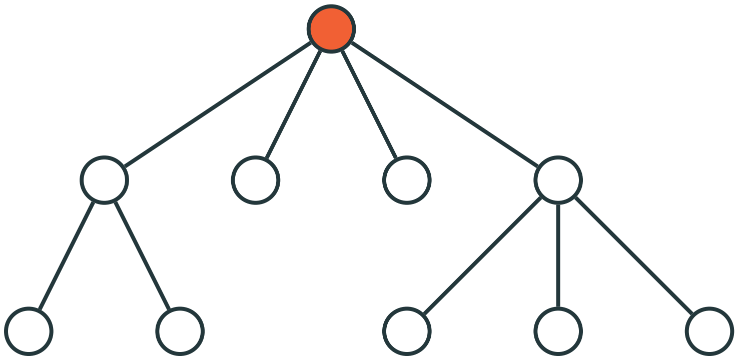
For odd $n > 2$, C_n is not bipartite



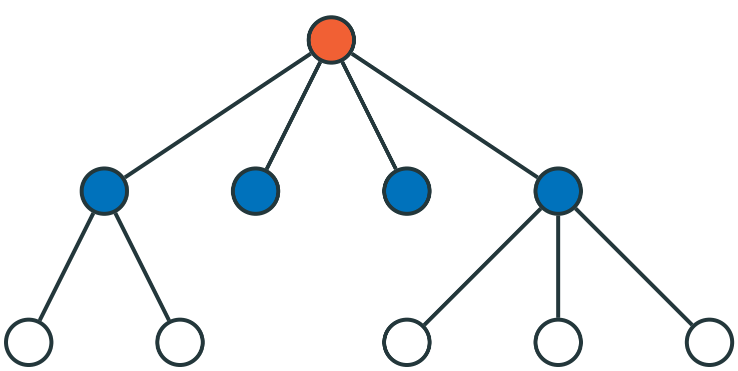
Trees are bipartite



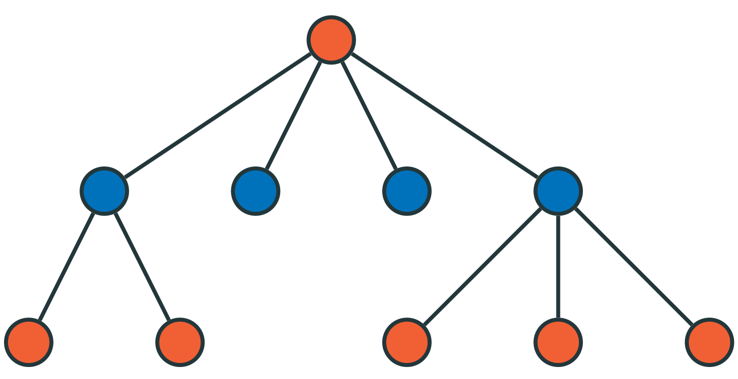
Trees are bipartite



Trees are bipartite



Trees are bipartite



Trees are bipartite