A graph $G = (V, E)$ is said to be $k$-connected if $|V| > k$ and we cannot disconnect $G$ by removing $k - 1$ vertices from $V$.

Is the graph in the figure

- 1-connected? Yes.
- 2-connected? Yes.
- 3-connected? No!
1- and 2-Connected Graphs

• We characterized all of the graphs that are 1-connected.
  ◦ These are exactly the connected graphs.

• Can we characterize all of the graphs that are 2-connected?
  ◦ What is the simplest type of 2-connected graphs? Cycles.

G-paths

• Given a graph $G$, a $G$-path is a path that is not a cycle and meets $G$ only at its endpoints.
2-connected Graphs

- **Theorem.** A graph is 2-connected if and only if it can be constructed by repeatedly adding $G$-paths to a cycle.

- **Proof (easy direction).**
  - If a graph was built by repeatedly adding $G$-paths to a cycle, it cannot be disconnected by removing one vertex.

The Other Direction

- Assume for contradiction that a 2-connected graph $G = (V, E)$ cannot be obtained by repeatedly adding $C$-paths to a cycle $C$.
- **There is a cycle $C$ in $G$.**
  - Otherwise, $G$ is a tree, and thus not 2-connected.
- **We repeatedly add $C$-paths** to $C$ using edges of $G$, until no such paths remain.
  - By definition, we obtain a subgraph $G' \subset G$. 
Completing the Proof

- $G$ – 2-connected graph that cannot be obtained by adding $C$-paths to a cycle $C$.
- $G' \subseteq G$ – a maximal subgraph that can be obtained by adding $C$-paths to cycle $C$.
  - Since $G$ is connected, there is a vertex $v \in V - G'$ that is connected by an edge to a vertex of $G'$.
  - Since $G$ is 2-connected, there must be another path between $v$ and $G'$.
  - Contradicting the maximality of $G'$.

Blocks

- **Recall.** Any graph can be decomposed into connected components.

  ![Connected Components](image)

- A block* is a maximal subgraph that is 2-connected.
  - Can we decompose every graph into blocks?

* The correct definition is 3 slides ahead.
Block Properties

- Can two blocks share a vertex? **Yes**

- Can two blocks share two vertices?
  - Let \( B_1, B_2 \) be two blocks with at least two common vertices.
  - If we remove a vertex of \( B_1 \) from \( B_1 \cup B_2 \), by definition \( B_1 \) remains connected, and remains connected to \( B_2 \).
  - We cannot disconnect \( B_1 \cup B_2 \) by removing one vertex, so it is **one big block**.

The Decomposition

- **We decompose a graph into blocks.** Does every edge belong to block? **No**
  - We refer to edges between blocks as **bridges**.
  - We **extend the definition of a block** so that a bridge is also considered as a block.
The Accurate Definition of a Block

- A **block** is a maximal subgraph that cannot be disconnected by removing one vertex.

- Deciding whether an isolated vertex is a block is just a matter of definition.

---

**st-disconnecting Set**

- Consider a graph $G = (V, E)$ and $s, t \in V$.
  - An **st-disconnecting set** is a subset $S \subseteq V \setminus \{s, t\}$ whose removal disconnects $G$, such that $s$ and $t$ are in different components.
Menger’s Theorem

- **Theorem (Menger 1927).** Consider a graph $G = (V, E)$ and vertices $s, t \in V$ such that $(s, t) \notin E$. Then the size of the smallest $st$-disconnecting set equals to the maximum number of vertex-disjoint paths between $s$ and $t$.

Proof

- $k_{\text{path}}$ – maximum number of vertex disjoint paths between $s$ and $t$.
- $k_{\text{disc}}$ – minimum size of an $st$-disconnecting set.

We have $k_{\text{disc}} \geq k_{\text{path}}$ since every $st$-disconnecting set must contain a vertex from every path.

We prove $k_{\text{disc}} \leq k_{\text{path}}$ by induction on $|V|$.
  - **Induction basis.** When $|V| = 2$, we have $k_{\text{disc}} = k_{\text{path}} = 0$
**Induction Step**

- \( N(s) \) – the set of neighbors of \( s \) in \( G \).
  - Notice that \( N(s) \) disconnects \( s \) from \( t \), and so does \( N(t) \).
- We partition the analysis of the induction step into two cases:
  - There exists a minimum-sized \( st \)-disconnecting set \( D \) such that \( D \neq N(s) \) and \( D \neq N(t) \).
  - Every minimum-sized \( st \)-disconnecting set is either \( N(s) \) or \( N(t) \) (one of these two sets might not be minimal).

**The First Case**

- Assume that there exists a minimum-sized \( st \)-disconnecting set \( D \) such that \( D \neq N(s) \) and \( D \neq N(t) \).
  - Removing \( D \) disconnects \( G \) into several components.
  - \( C_s \) - the component containing \( s \).
  - \( C_t \) - the component containing \( t \).
  - How can we use the induction hypothesis?
The First Case (cont.)

- $G_s$ - the induced graph on $C_s \cup D$.
  - We add a vertex $t'$ to $G_s$ and edges between $t'$ and every vertex of $D$.
  - Since $D$ is a min $st$-disconnecting set of $G$, it is also a min $st'$-disconnecting set of $G_s$.
  - By the **induction hypothesis**, there are $|D| = k_{\text{disc}}$ vertex-disjoint paths from $s$ to $t'$.

Completing the First Case

- We have a set of **vertex disjoint paths** from $s$ to each of the $k_{\text{disc}}$ vertices of $D$.
- Similarly, we have a set of **vertex disjoint paths** from each of the $k_{\text{disc}}$ vertices of $D$ to $t$.
  - Combining the two yields a set of $k_{\text{disc}}$ vertex disjoint paths from $s$ to $t$.
  - That is, $k_{\text{disc}} \leq k_{\text{path}}$, completing the proof in this case.
The Second Case

- Assume that every min st-disconnecting set is either $N(s)$ or $N(t)$.
  - That is, $v \in V \setminus (\{s, t\} \cup N(s) \cup N(t))$ is not in any minimum-sized st-disconnecting set.
  - By removing such a vertex $v$, we obtain a graph $G'$, also with a min st-disconnecting set of size $k_{\text{disc}}$.
  - By the hypothesis, $G'$ contains $k_{\text{disc}}$ vertex-disjoint paths between $s$ and $t$. These also exist in $G$.

A Missing Case

- What is still missing in case 2?
  - What if there is no vertex $v \in V \setminus (\{s, t\} \cup N(s) \cup N(t))$?
  - Let $C = N(s) \cap N(t)$, $N_s = N(s) \setminus C$, and $N_t = N(t) \setminus C$.
  - Any disconnecting set must contain $C$, which also corresponds to $|C|$ paths of the form $s \rightarrow v \rightarrow t$, where $v \in C$.
  - Each of the other $k_{\text{disc}} - |C|$ vertices of the minimum disconnecting set is either in $N_s$ or $N_t$. 
Completing the Missing Case

- Let $C = N(s) \cap N(t)$, $N_s = N(s) \setminus C$, and $N_t = N(t) \setminus C$. Consider the bipartite subgraph on $N_s \cup N_t$ (removing edges between vertices of the same side).
- The minimum disconnecting set contains $k_{\text{disc}} - |C|$ vertices in this subgraph. These vertices form a minimum vertex cover.

Recall: König’s Theorem

- **Theorem.** Let $G = (V_1 \cup V_2, E)$ be a bipartite graph. Then the size of a maximum matching of $G$ is equal to the size of a minimum vertex cover of $G$. 
Using Vertex Covers

- Since $G'$ has a minimum vertex cover of size $k_{\text{disc}} - |C|$, it has a matching $A$ of the same size.
  - Each matching edge corresponds to a path $s \rightarrow v \rightarrow u \rightarrow t$ ($v \in N_s$ and $u \in N_t$).
  - These paths are vertex disjoint, so we again have at least $k_{\text{disc}}$ vertex-disjoint paths.

Conclusion

- **Menger’s theorem** yields an alternative definition of $k$-connectedness.
  - **Original definition.** A graph $G = (V, E)$ is said to be $k$-connected if $|V| > k$ and we cannot obtain a non-connected graph by removing $k - 1$ vertices from $V$.
  - **Equivalent definition.** A graph $G = (V, E)$ is said to be $k$-connected if $|V| > k$ and between any two vertices $s, t \in V$ with $(s, t) \notin E$ there are at least $k$ vertex-disjoint paths.
Verifying $k$-Connectedness

- **Problem.** Given a graph $G = (V, E)$ and an integer $k > 0$, describe an algorithm for checking whether $G$ is $k$-connected.

Solution

- **For every pair of vertices** $s, t \in V$ with $(s, t) \notin E$, we check whether there are $k$ vertex-disjoint paths between $s$ and $t$.
  - $G$ is $k$-connected if and only if all of the $\binom{|V|}{2}$ checks pass.
- How can we check whether there are $k$ vertex-disjoint paths between $s$ and $t$?
  - We did this in 6a using flow networks.
Building a Flow Network

- A quick reminder from 6a:
  - The source is $s$. The sink is $t$.
  - The capacities are all 1.
  - We split every edge into a pair of anti parallel edges.
  - We split every $v \in V$ into $v_{in}$ and $v_{out}$.

More Efficient

- We showed how to check whether a graph is $k$-connected by finding maximum flow in $\binom{|V|}{2}$ flow networks.
- By more involved argument, it suffices to find $|V| - 1$ maximum flows.
The End

**ANSWER:** natural selection

---

**THAG MAKE FIRE.**

---

**THAG INVENT WHEEL.**

---

**NOW THAG WILL...**

---

**SABER TOOTH!**

---

**WAIT! THAG CALCULATE.**

---

**SHORTEST DISTANCE TO LOCATION MAXIMIZING PROBABILITY OF SURVIVAL.**

---

**L = \int dy \sqrt{1 + y'^2} dx**

\[ \frac{\partial L}{\partial y} - \frac{d}{dx} \frac{\partial L}{\partial y'} = 0 \]

---

**QUESTION:** Why are there so many more jocks than nerds in the world today?