B Connectivity, Spanning Trees, and Directed Graphs

B1 Connectivity

A cut-vertex is a vertex whose removal disconnects its component. A bridge is an edge whose removal disconnects its component: an edge is a bridge if and only if it belongs to no cycle. A block is a maximal connected subgraph that has no cut-vertex.

A cut-set is a set of vertices \( S \) such that \( G - S \) has more than one component. The connectivity \( \kappa(G) \) of graph \( G \) is the size of the smallest cut-set, except that we define \( \kappa(K_n) = n-1 \). The edge-connectivity \( \kappa'(G') \) is the smallest cardinality of a disconnecting set of edges, except that we define \( \kappa'(K_1) = 0 \).

For a graph \( G \), the notation \( \delta(G) \) means the minimum degree and \( \Delta(G) \) means the maximum degree. Fundamental inequality is:

\[
\text{Inequality chain. For any graph } G \text{ we have } \kappa(G) \leq \kappa'(G) \leq \delta(G).
\]

Proof. To show that \( \kappa' \leq \delta \), note that removing all edges incident with a vertex of minimum degree disconnects the graph. To show that \( \kappa \leq \kappa' \), we observe that both parameters are 0 if the graph is disconnected. So assume the graph is connected, and let \( X \) be a minimum disconnecting set of edges, say containing edge \( e \). If one were to remove all edges of \( X \) except \( e \), what remains is connected, by the minimality of \( X \). So, for each such edge, consider one of the ends that is not an end of \( e \), and remove that from the graph. If the result is disconnected, or there are only two vertices remaining, then the bound is proved. So assume otherwise. Then \( e \) is a bridge, and removing one end of \( e \) disconnects the graph. \( \square \)

It can be shown that if graph is 3-regular, then \( \kappa = \kappa' \).

Example. The hypercube \( Q_k \) has connectivity \( k \). Think of \( Q_k \) as two copies of \( Q_{k-1} \) where each pair of corresponding vertices are joined by an edge. We prove by induction, that removing \( k - 1 \) vertices does not disconnect \( Q_k \). If we remove no vertex from one of the copies of \( Q_{k-1} \), then that copy retains intact and every remaining vertex is connected to that copy. If we remove
some vertices from each copy of $Q_{k-1}$ totaling $k-1$, then by the induction hypothesis both copies remain connected, and since $k - 1 < 2^{k-1}$, some edge joining the two copies remains.

B2 Spanning Trees

A **spanning subgraph** is one with all the vertices and only some of the edges. One can transform any spanning tree into any other spanning tree by changing one edge at a time (that is, a **greedy algorithm**.) Taking an end-vertex of a spanning tree shows that every nontrivial graph has a vertex that is not a cut-vertex.

A famous result is the following:

**Cayley’s Formula.** The labeled $K_n$ has $n^{n-2}$ different spanning trees.

**Proof Sketch.** The proof idea is to prove a bijection between the set of spanning trees on the one hand, and the set of sequences of length $n - 2$ whose entries are drawn from $\{1, \ldots, n\}$ on the other hand. There are clearly $n^{n-2}$ such sequences.

We label the vertices of $K_n$ with 1 up to $n$, and given a spanning tree create what is called the Prüfer code. At each stage, delete the smallest-numbered remaining leaf and write down its neighbor. Stop when two vertices are left. The surprising fact is that this process is reversible. It can be argued that different trees have different codes and that given such a sequence one can build a spanning tree with that Prüfer code. □

The notation $G - e$ means the graph obtained by deleting edge $e$. The notation $G \cdot e$ means the graph obtained by contracting edge $e$, meaning merge its two ends into one vertex; note that the result might not be a simple graph.

**Contraction–Deletion.** If $\tau$ is the number of spanning trees then $\tau(G) = \tau(G - e) + \tau(G \cdot e)$ for any edge $e$.

**Proof.** Consider a spanning tree $T$ of graph $G$. If $T$ does not contain edge $e$, then $T$ is a spanning tree of $G - e$ and conversely. If $T$ does contain edge $e$, then
$T \cdot e$ is a spanning tree of $G \cdot e$, and conversely, every spanning tree of $G \cdot e$ becomes a spanning tree of $G$ when $e$ is “uncontracted”. $\Box$

The **adjacency matrix** is the square matrix where the entry in row $i$ column $j$ is 1 if there is an edge between vertices $i$ and $j$, and 0 otherwise. The **Laplacian matrix** is $D - A$ where $D$ is the diagonal matrix with the degrees and $A$ is the adjacency matrix. We omit the proof of:

**Matrix Tree Theorem.** The number of spanning trees is given by the absolute value of any cofactor (determinant of matrix with one row and column deleted) of the Laplacian.

**Graceful labeling of tree:** A labeling of the vertices with 0 up to $n - 1$ so that vertices have distinct labels and the edge-differences are distinct. A famous unsolved conjecture is that every tree has a graceful labeling.

In graphs with nonnegative edge-weights, Kruskal’s algorithm finds a **minimum-weight spanning tree**. The algorithm says repeatedly take the cheapest edge that does not create a cycle. (Proof of validity of Kruskal is by switches: comparing the alleged minimum spanning tree with one from Kruskal.)

A **breadth-first-search tree** is a spanning tree created by rooting at a vertex, taking its neighbors, then their neighbors, and so on. This can be generalized to calculate the distance from a vertex to all other vertices, in Dijkstra’s algorithm.

## B3 Some Extremal Results

Many questions in graph theory ask about the minimum or maximum of a quantity (such as edges) for graphs with or without some property. An **extremal** graph is an example with the minimum or maximum.

**Manvel.** The maximum number of edges in a triangle-free simple graph of order $n$ is $\lfloor n^2/4 \rfloor$.

**Proof.** Start with a vertex $v$ of maximum degree $\Delta$. Let $N(v)$ be its neighborhood. Because the graph has no triangle, the set $N(v)$ is independent. This means that every edge of the graph has one or two ends outside $N(v)$. So when
we sum up the degrees of the vertices outside \( N(v) \), this counts all edges (and some of them twice). There are \( n - \Delta \) vertices outside \( N(v) \) and each has degree at most \( \Delta \). So we get that the total number of edges is at most \( \Delta (n - \Delta) \). Then by calculus this is maximized at \( \Delta = n/2 \). The extremal graph is \( K_{\lceil n/2 \rceil, \lfloor n/2 \rfloor} \).

**BIPARTITE SUBGRAPHS.** The complete graph \( K_n \) can be:

(a) decomposed into \( \lceil \log_2 n \rceil \) bipartite subgraphs.

(b) decomposed into \( n - 1 \) complete bipartite subgraphs.

And these numbers are best possible.

**PROOF SKETCH.** We show only the existence here. (a) Assume \( n \) is a power of 2. For the first subgraph, partition the vertex set of \( K_n \) into two equal sets and take the complete bipartite subgraph with those as partite sets. What remains is two disjoint copies of \( K_{n/2} \) and one can use the same color scheme on both copies.

(b) Color all the edges incident with the first vertex with the same color. What remains is \( K_{n-1} \) and one can repeat.

Decomposition results can also be expressed as union results. The union of two graphs is constructed by unioning their vertex sets and edge sets (throwing away duplicates). The disjoint union of two graphs and is the union assuming their vertex sets are disjoint (and result is guaranteed to be disconnected).

**TREE SUBGRAPHS.** If a graph has minimum degree at least \( k \), then it contains as subgraphs all trees with at most \( k \) edges.

**PROOF.** Build each desired tree one vertex at a time.

### B4 Directed Graphs

A **directed graph** or digraph has directed edges (sometimes called arcs). Each arc has a head and a tail.

A path in a digraph must respect the arrows. A digraph is strongly connected if there is path from every vertex to every other vertex. The strong components
are its maximal strongly connected subgraphs. Subgraph, isomorphism, decomposition, union is similar to before. Examples include: Markov chains, finite automata, and game graphs.

The **in-degree** and **out-degree** of a vertex are the numbers of edges going in or going out of the vertex. If the out-degree is positive for all vertices, then the digraph has a cycle; similarly with in-degree always positive. It can be shown that a connected digraph is Eulerian if and only if the in-degree equals out-degree at every vertex.

An **orientation** of a graph means choosing one orientation of each edge. A **tournament** is an oriented $K_n$. We define a **king** as a vertex that can reach all other vertices with a path of length at most 2.

**Kings.** Every tournament has a king.

**Proof.** We claim that every vertex $v$ of maximum out-degree is a king. For, consider any other vertex $w$. If there is an arc from $v$ to $w$, then $v$ reaches $w$ in one. So assume not; that is, there is an arc from $w$ to $v$. Since the out-degree of $w$ is at most that of $v$, there must be some vertex $x$ such that $x$ is an out-neighbor of $v$ but not of $w$. This means $v$ reaches $w$ in two going $v \to x \to w$. \qed