Last week:

- Spanning trees.
- Minimum spanning trees in weighted graphs:
  - Kruskal's algorithm
  - Prim's algorithm

This week:

- Counting spanning trees (no proofs).
- Measuring connectivity:
  - Cut-vertices
  - Blocks

Recall: • A spanning tree of a graph G is a subgraph of G which is spanning (uses all vertices) and is also a tree (connected, no cycles).

• If $V(G) = \{v_1, \ldots, v_n\}$, the **incidence matrix** of G is $A = [a_{ij}]$ where $a_{ij} = \begin{cases} 1 & \text{if } v_i v_j \in E(G), \\ 0 & \text{otherwise.} \end{cases}$

Definition. • *The **Laplace matrix** of* G *is* $Q = [q_{ij}]$ *where* $q_{ij} = \begin{cases} \deg_G(v_i) & \textit{if } i = j, \\ -1 & \textit{if } v_i v_j \in E(G), \\ 0 & \textit{otherwise.} \end{cases}$

• *We denote by* $Q(1,1)$ *the matrix obtained from* Q *by deleting the first row and column.*

Theorem. *(The Matrix Tree Theorem) If* G *is a connected graph with Laplace matrix* Q*, then the number of spanning trees of* G *equals* $\det Q(1,1)$*.*

Example. *Compute the number of spanning trees of the following graph.*

Recall: • An edge e in a graph G is a **bridge** if $G - e$ has more components than G.

• If v is a vertex in G, then $G - v$ deletes v and all incident edges from G.

Definition. *A vertex v in a graph G is a **cut-vertex** if $G - v$ has more components than G.*

Example. *Identify all cut-vertices in the following graph.*

Example. *(1) Many bridges and a unique cut-vertex.* *(2) Many cut-vertices but no bridges.*

Theorem (5.1)**.** *Let* v *be a vertex incident with a bridge in a connected graph* G*. Then* v *is a cut-vertex of* G *if and only if* $\deg v \geqslant 2$*.*

Exercise 1. *In a tree with at least* 3 *vertices, any non-leaf is a cut-vertex.*

Exercise 2. *Let* G *be a connected graph on at least* 3 *vertices. If* G *contains an edge then it contains a cut-vertex.*

Theorem (5.3)**.** *Let v be a cut-vertex in a connected graph G, and let u, w be vertices in distinct components of $G - v$. Then v lies on every u–w path of G.*

Theorem (5.5)**.** *Let* G *connected with at least* 2 *vertices and fix* u *in* $V(G)$*. If* v *is a vertex that is farthest from* u *in* G*, then* v *is not a cut-vertex of* G*.*

Exercise 3 (Corollary 5.6)**.** *If* G *is connected with at least* 2 *vertices, then* G *contains at least* 2 *vertices that are not cut-vertices.*

Definition. *A connected graph with no cut-vertices is called* ***non-separable****.*

Theorem (5.7)**.** *A graph* G *with* $|V(G)| \geqslant 3$ *is non-separable if and only if every two vertices lie on a common cycle.*

Recall: Components in a graph G are equivalence classes under the relation on $V(G)$ defined by $u\mathcal{R}v$ iff there is a u–v path.

Definition. *Let G be a graph. We define a relation $\mathcal{B}$ on $E(G)$ by $e\mathcal{B}f$ iff $e = f$ or there is a cycle on G that uses e and f.*

Theorem (5.8)**.** *The relation $\mathcal{B}$ defined above is an equivalence relation.*