

Graph theory

We started the discussion by reviewing the definitions in graph theory that they learned last time, on August 24, 2025. The list of definitions is as follows, and see the session report of August 24, 2025 for the details: graphs, edges, vertices, paths, directed graphs, undirected graphs, loops, multiple edges, distance between vertices, degree of a vertex, connected component, finite graph, and simple graph.

This time we went on to write down more rigorous proving of the **Handshaking Theorem**, i.e., the total number of vertices of odd degree in a finite simple graph is an even number. First, we wrote the proof that students came up with, which draws inspiration from the “cup game” that they played on August 10, 2025.

Handshaking Theorem: Let $G = (V, E)$ be a finite simple graph.

Proof: We proceed by induction on the number of edges $|E|$ in the graph.

Base Case ($|E| = 0$): If the graph has no edges, then every vertex has degree 0, which is even. Hence, the number of vertices with odd degree is 0, which is even.

Inductive Step: Assume that for every simple graph with k edges, the number of odd-degree vertices is even.

Now, consider a graph G' with $k + 1$ edges. Construct G' by adding one edge $e = \{u, v\}$ to a graph G with k edges.

Adding edge e increases the degree of each of the two vertices u and v by 1. Consider the change in parity of the degrees:

- If $\deg(u)$ was even before adding e , it becomes odd.
- If $\deg(u)$ was odd, it becomes even.
- Similarly for $\deg(v)$.

Thus, the parity of $\deg(u)$ and $\deg(v)$ flips. Therefore we have:

- If both u and v had even degrees, both become odd: the number of odd-degree vertices increases by 2.
- If both had odd degrees, both become even: the number of odd-degree vertices decreases by 2.
- If one was even and the other odd: one becomes odd, one becomes even: the number of odd-degree vertices remains unchanged.

In each case, the parity of the total number of odd-degree vertices is preserved. By the inductive hypothesis, the number of odd-degree vertices in G is even. Therefore, the number of odd-degree vertices in G' is also even. By mathematical induction, in any finite simple graph, the number of vertices with odd degree is even

Students also came up with another idea, which led to another proof of the Handshaking Theorem.

Another proof: Let's calculate the total number of edge ends in two different ways: vertex by vertex and edge by edge.

- First, just as in the problem about the computers, this total can be calculated vertex by vertex: it will be equal to the sum of the edge ends that start at each vertex. Using the new terminology, we can rephrase this approach as: "the total number of edge ends is equal to the sum of the degrees of the vertices."
- Next, let's note that this total number of edge ends can also be calculated edge by edge: it is equal to twice the total number of edges in the graph.

Since each edge has two ends, the total number of edge ends must be even. This means the total calculated using the first approach, vertex by vertex, must be even as well. Every even-degree vertex contributes an even addend; every odd-degree vertex contributes an odd addend. For the total to be even, the sum of these odd addends must be even. Therefore, the number of odd-degree vertices must be even.

Solutions to the previously assigned set of problems, see session report of August 24, 2025, on the Handshaking Theorem were discussed.

Combinatorics

In the latter half, students explored the following problems related to Catalan numbers. These problems are an addition to the problems from the last session on August 24, 2025. The combinatorics of Catalan numbers will stretch also for another session. Solutions were to be discussed much later, after giving sufficient time in the next session. Students also had time to spend on problems from August 24, 2025.

Problem 1. A bunny is hopping up and down a very long staircase. Each hop is either:

- an **up-hop** (going one step higher), or
- a **down-hop** (going one step lower).

The bunny starts on the ground (height 0), makes exactly $2n$ hops (up or down) and **must** return to the ground at the end of it.

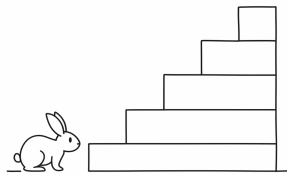


Figure 1:

- (Qn 1) Suppose the bunny makes 2 hops ($n = 1$). How many valid sequences of hops are there? List them.
- (Qn 2) Do the same for 4 hops ($n = 2$). How many valid sequences are there? What patterns do you see?

(Qn 3) Repeat this for 6 hops ($n = 3$).

(Qn 4) **Challenging Question:** How many such sequences are there for $2n$ hops?

Problem 2. Let us consider a **string of $2n$ brackets** having n opening brackets and n closing brackets. We say such a string is called **balanced** while reading from left to right,

- (a) you must be able to pair each open bracket with a closed bracket
- (b) You cannot close a bracket without opening it . In otherwords, you never have more closing brackets) than opening ones (.

Examples:

- $((()))$ is balanced.
- $()()()$ is not balanced.
- $((()))$ is balanced.
- $)((()$ is not balanced.

(Qn 1) List the number of balanced strings for $n = 1, 2, 3$.

(Qn 2) **Challenging Question:** How many such balanced string are there of length $2n$? i.e. with n opening brackets and n closing brackets?

(Qn 3) Is there any connection to this problem and the previous one? If so, what is the connection?

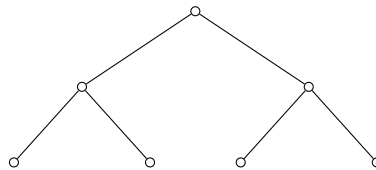
Problem 3. A **full binary tree** is a special kind of graph made using dots (called nodes) and lines (called edges), with these rules:

- Every node is either:
 - a **leaf node** — it has no children (no lines going down), or
 - an **internal node** — it has exactly **two** children.
- The tree starts from a top node called the **root**.
- There are no loops or cycles. It looks like a branching structure going downward.

Let n be the number of **internal nodes** — the ones that have children.

Example:

Here is a full binary tree with 3 internal nodes:

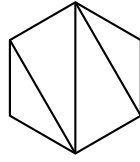


(Qn 1) Draw all full binary trees for $n = 1, n = 2, n = 3$ and $n = 4$ internal nodes. How many are there in each cases ?

(Qn 2) **Challenging Question:** How many such full binary trees are there with n internal nodes?

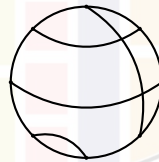
(Qn 3) Is there any connection to this problem and the previous two problems? If so, what is the connection?

Problem 4. Consider a convex polygon with $n + 2$ sides. A *triangulation* of this polygon is a way of drawing non-crossing diagonals inside the polygon so that the entire polygon is divided into triangles. For example, here is one triangulation of a hexagon ($n = 4$).



- (Qn 1) How many ways can a quadrilateral ($n = 2$) be triangulated? Can you draw all of them?
- (Qn 2) How many ways can a pentagon ($n = 3$) be triangulated? Draw all of them.
- (Qn 3) Draw all the triangulations of a hexagon ($n = 4$).
- (Qn 4) **Challenging Question:** How many triangulations are there of an $(n + 2)$ -gon? What is the formula or pattern?
- (Qn 5) **Challenging Question:** Label the vertices of the $(n + 2)$ -gon as $1, 2, 3, \dots, n + 2$. In a triangulation, consider the triangle T that includes vertices 1, 2, and some vertex k . What are the sizes of the two smaller polygons outside T ? Does this give a clue about how many triangulations contain triangle T ?
- (Qn 6) **Challenging Question:** If the left region has $k - 3$ vertices and the right has $n - k$ vertices in (Qn 5), Does this give any recurrence relation?
- (Qn 7) **Challenging Question:** Does it has any similarity with each of the previous problem? If it has some connection, Explain it in details.

Problem 5. $2n$ people sit around a round table. Any person may shake hands with another, but we only consider non-crossing handshakes. For example, the figure below shows a crossing scenario that is not allowed when $n = 4$.



Example: A crossing pairing [not valid] for $n = 4$ (among 8 people).

- (Qn 1) How many ways can 4 people ($n = 2$) shake hands without crossing? Draw all scenarios.
- (Qn 2) How many ways can 6 people ($n = 3$) shake hands without crossing? Draw all scenarios.
- (Qn 3) Draw all the ways in which 8 people ($n = 4$) can shake hands without crossing.
- (Qn 4) **Challenging Question:** How many ways can $2n$ people shake hands without crossing?
- (Qn 5) **Challenging Question:** Fix person 1 and suppose they shake hands with person k . How many people lie on each side of this handshake? Why must each side be an independent non-crossing handshake problem?
- (Qn 6) **Challenging Question:** Does it has any similarity with each of the previous problems? If it has some connection, Explain it in details.