AI Certification Program Lecture Notes

Application of Graph Theory and Computational Proof AI

Aligned with the AI Certification Program and the AMK Research Lab Program

Core theme	Why it matters	What learners build
Relational AI + formal reasoning	Graphs capture dependencies; proof AI checks correctness	Knowledge graph tools, attack graphs, proof search demos, capstone ideas

Prepared for instructional use. Visuals include original teaching diagrams and an AMK Research Lab example figure.

1. Lecture overview

Purpose. These notes introduce graph theory as a foundation for modern AI systems and computational proof AI as a pathway to verifiable reasoning, theorem proving, and safety-critical decision support.

Learning outcomes.

- Explain how graphs model entities, relations, dependencies, and constraints.
- Distinguish graph algorithms, graph machine learning, knowledge graphs, and computational proof systems.
- Identify practical applications in cybersecurity, fraud analytics, healthcare, logistics, and formal verification.
- Map technical work to governance expectations from NIST AI RMF, OECD AI Principles, UNESCO ethics guidance, and ISO/IEC 42001.

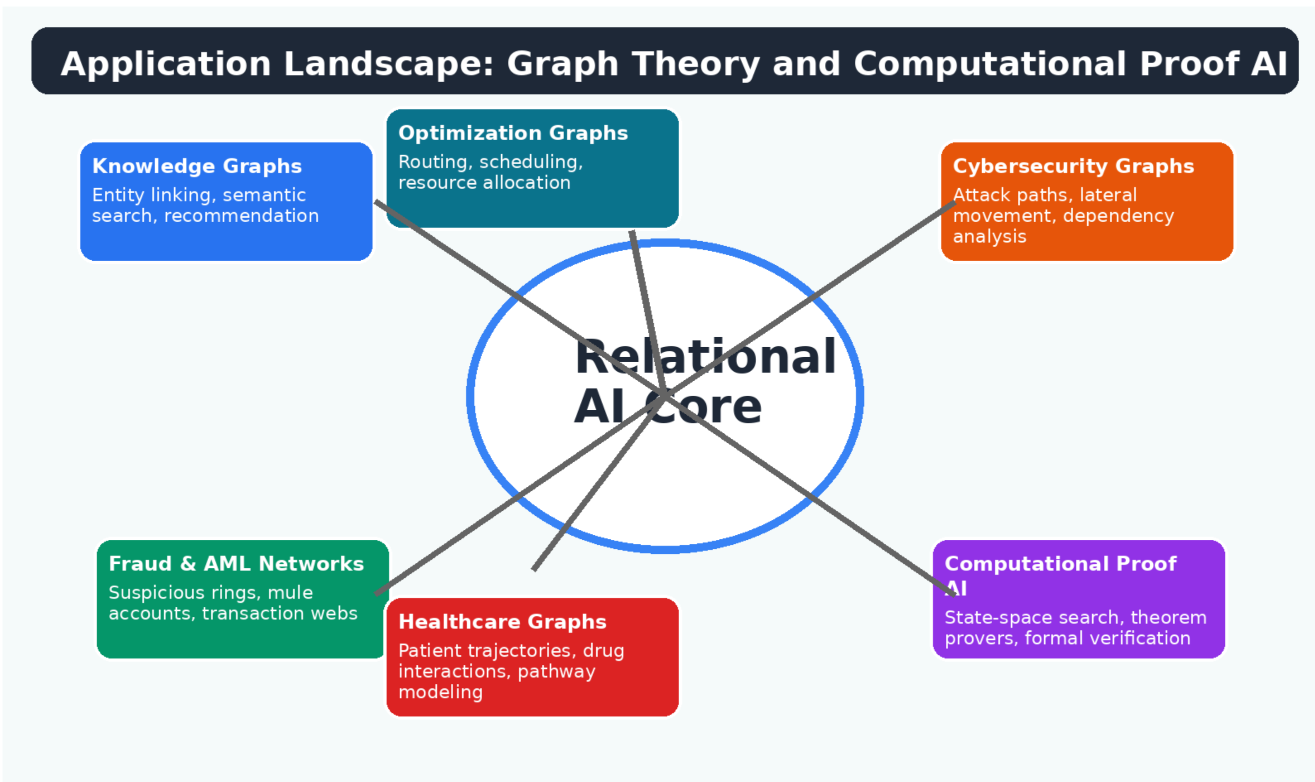

Figure 1. Teaching landscape linking graph theory to knowledge graphs, cybersecurity, fraud analytics, healthcare, optimization, and computational proof AI.

2. Why graph theory matters in AI

Graph theory represents systems as nodes and edges. This is valuable in AI because many real problems are relational rather than purely tabular: users connect to items, devices connect to services, transactions connect to accounts, and proof steps connect to subgoals.

- Shortest path and flow methods support routing, planning, and optimization.
- Centrality and community detection help identify influential entities, attack hubs, and fraud rings.
- Knowledge graphs encode semantics and improve retrieval, explainability, and reasoning.
- Graph neural networks extend machine learning to non-Euclidean data through message passing.

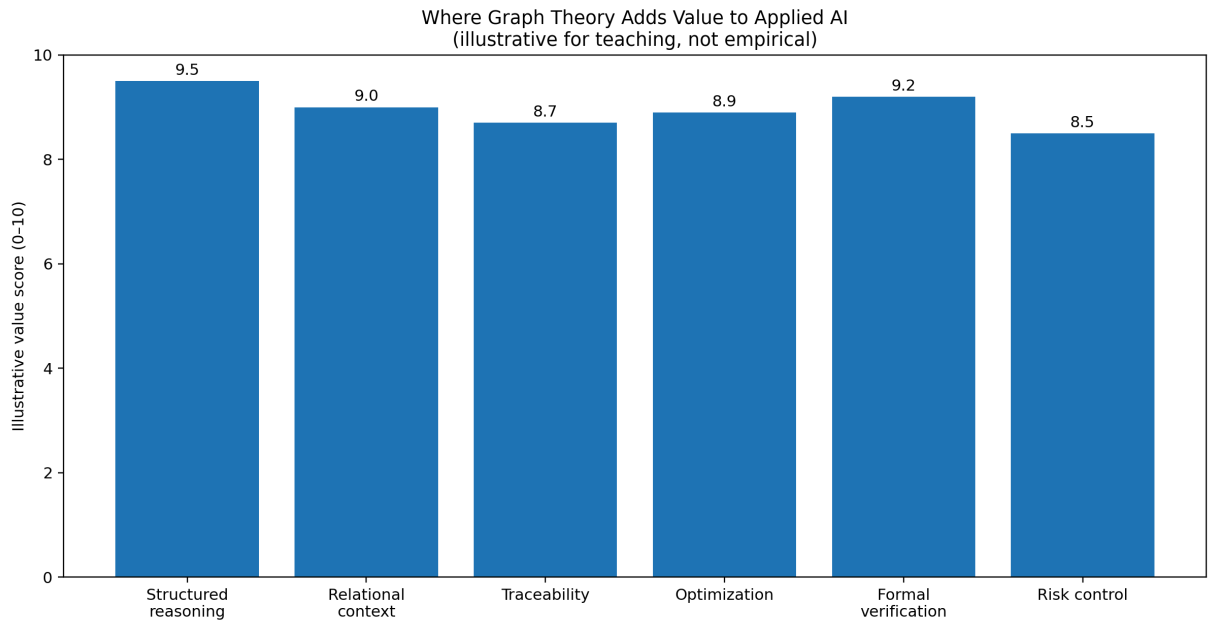

Figure 2. Illustrative classroom chart showing where graph-based methods add value in applied AI.

3. Computational proof AI in practice

Computational proof AI focuses on proving that a conclusion follows from axioms, rules, or prior lemmas. Unlike ordinary next-token prediction, proof systems must maintain formal consistency and produce checkable derivations.

- Theorem provers search over proof states and candidate tactics.
- Proof assistants verify each inference step inside a formal system.
- Learned heuristics can prioritize promising branches, while symbolic rules preserve correctness.
- This makes proof AI important for mathematics, software verification, hardware assurance, and safety-critical workflows.

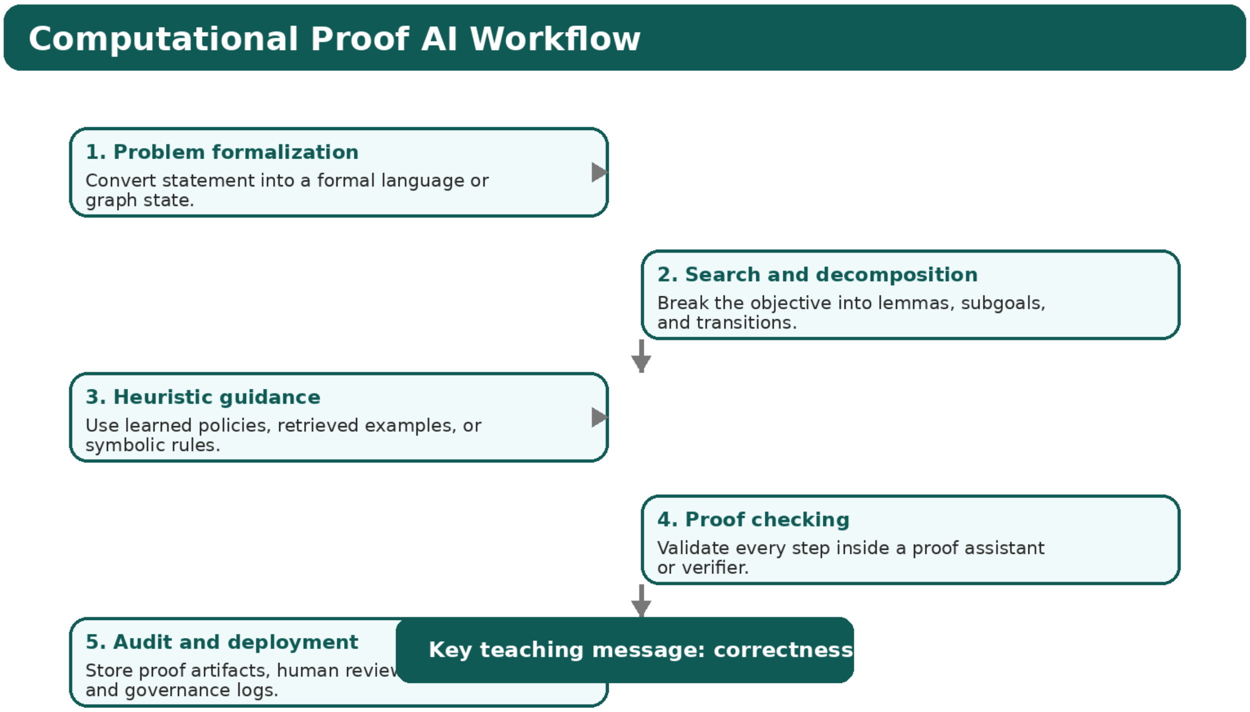

Figure 3. A practical workflow for computational proof AI, from formalization to proof checking and governance logging.

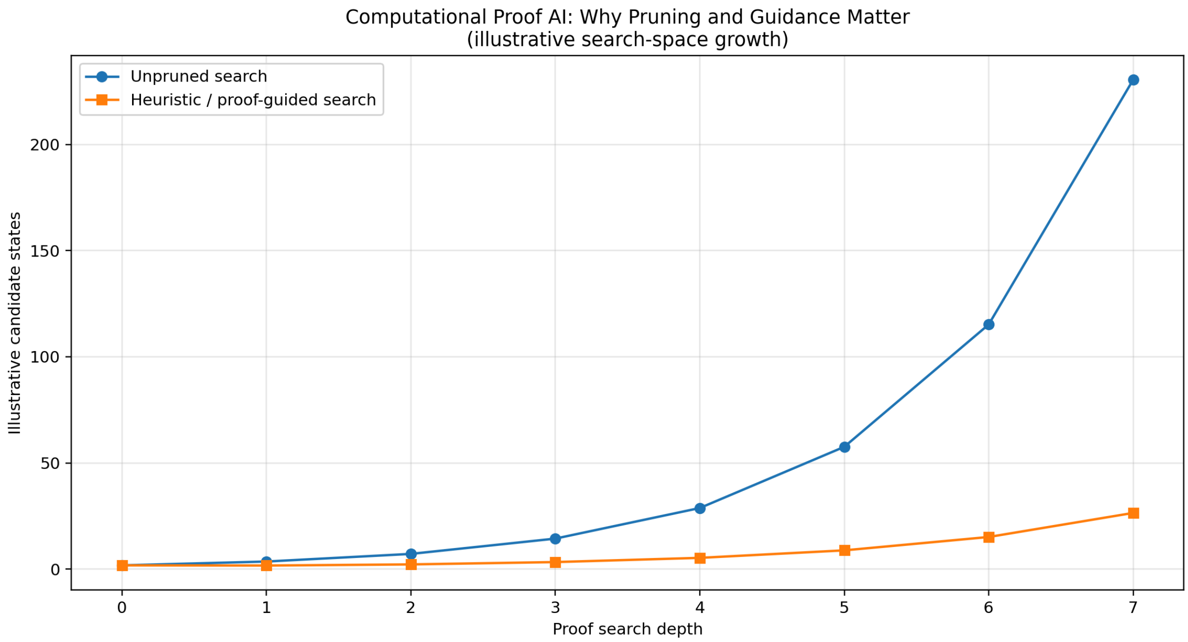

Figure 4. Illustrative proof-search chart showing how pruning and guidance can reduce search-space growth.

4. Practical applications for the certification program

Area	Graph / proof method	Typical AI task	Sample learner artifact	Risk or control note

Cybersecurity	Attack graphs, dependency graphs, GNNs	Lateral movement analysis, anomaly scoring	Intrusion graph dashboard	Protect logs, validate alerts, keep human override
Fraud / AML	Transaction graphs, community detection	Suspicious ring detection	Fraud network explorer	Monitor false positives and fairness
Healthcare	Patient trajectory graphs, pathway graphs	Risk prediction, care sequencing	Clinical graph prototype	Use strong privacy, audit access, clinician oversight
Education	Knowledge graphs, proof assistants	Concept mapping, automated hinting	Proof tutor or curriculum graph	Avoid over-automation in assessment
Software assurance	State graphs, theorem proving	Formal verification and invariant checking	Proof trace report	Retain evidence and change control

5. Alignment with the AMK Research Lab Program

Research-ready pathway. The topic fits the AMK Research Lab because it naturally supports a full research lifecycle: problem formulation, mathematical modeling, prototype implementation, evaluation, and governance analysis.

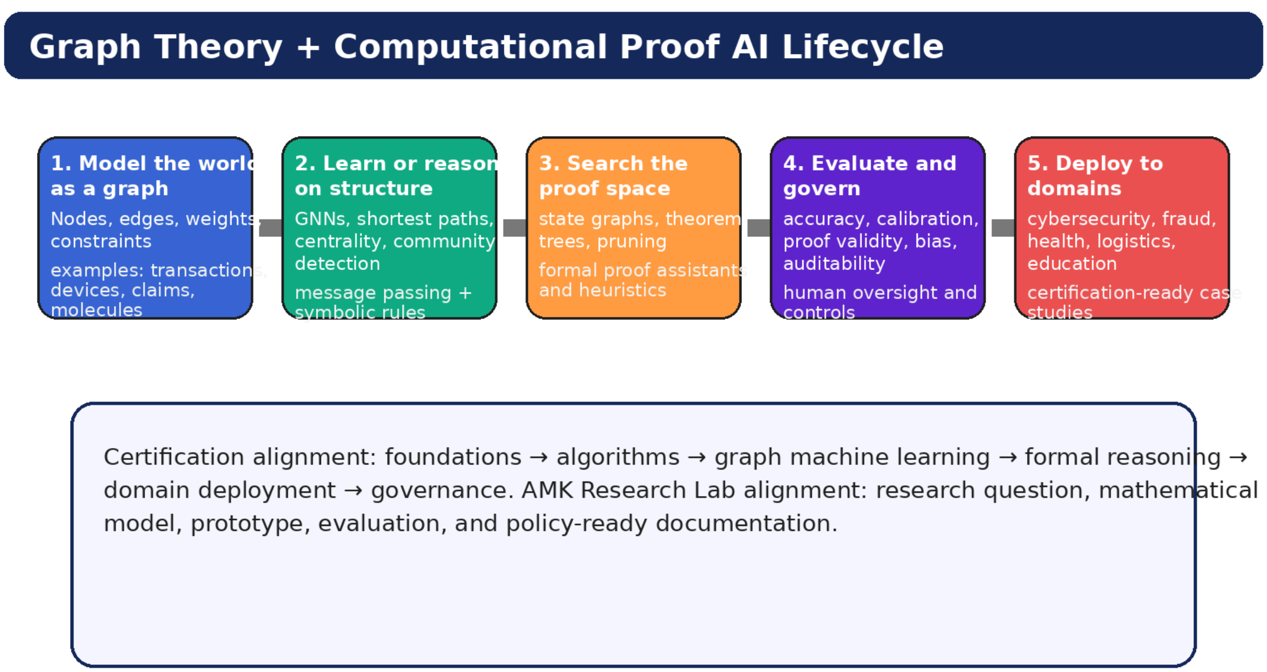

Figure 5. A research and certification lifecycle for graph-theoretic AI and computational proof AI.

Suggested capstone directions:

- Graph-based cybersecurity intrusion detection with attack-path explanation.
- Financial fraud ring detection using transaction graphs and community analysis.
- Healthcare prediction with patient journey graphs and interpretable risk pathways.
- Computational proof tutor that converts geometry or logic problems into proof states and validates solution steps.

6. Example AMK-style capstone concept

Sample concept. A fiduciary advisory system can use graph modeling for dependency analysis and computational reasoning for explainable, auditable decision support. The figure below illustrates how a research project can synthesize governance, stability, sentiment, cybersecurity, and auditability.

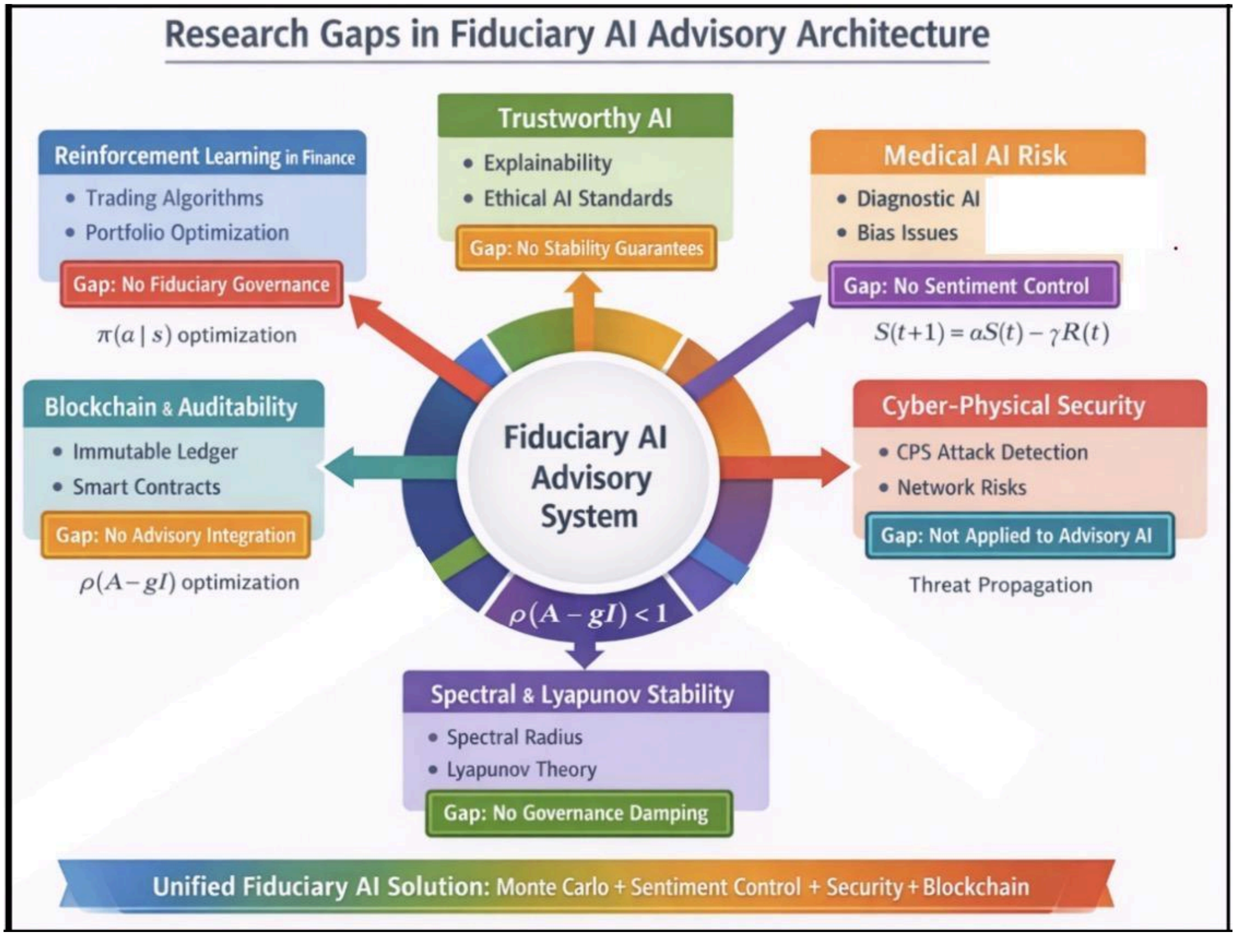

Figure 6. Example research-gap graphic supplied by the user, suitable as an AMK Research Lab capstone inspiration.

7. Governance, ethics, and controls

Why controls matter. Graph-based and proof-based AI can improve reasoning and traceability, but they also introduce data, privacy, safety, and governance risks. NIST AI RMF emphasizes governance, mapping context, measuring risk, and managing controls across the lifecycle. OECD and UNESCO emphasize human rights, transparency, robustness, accountability, and human oversight. ISO/IEC 42001 turns these expectations into a management-system discipline.

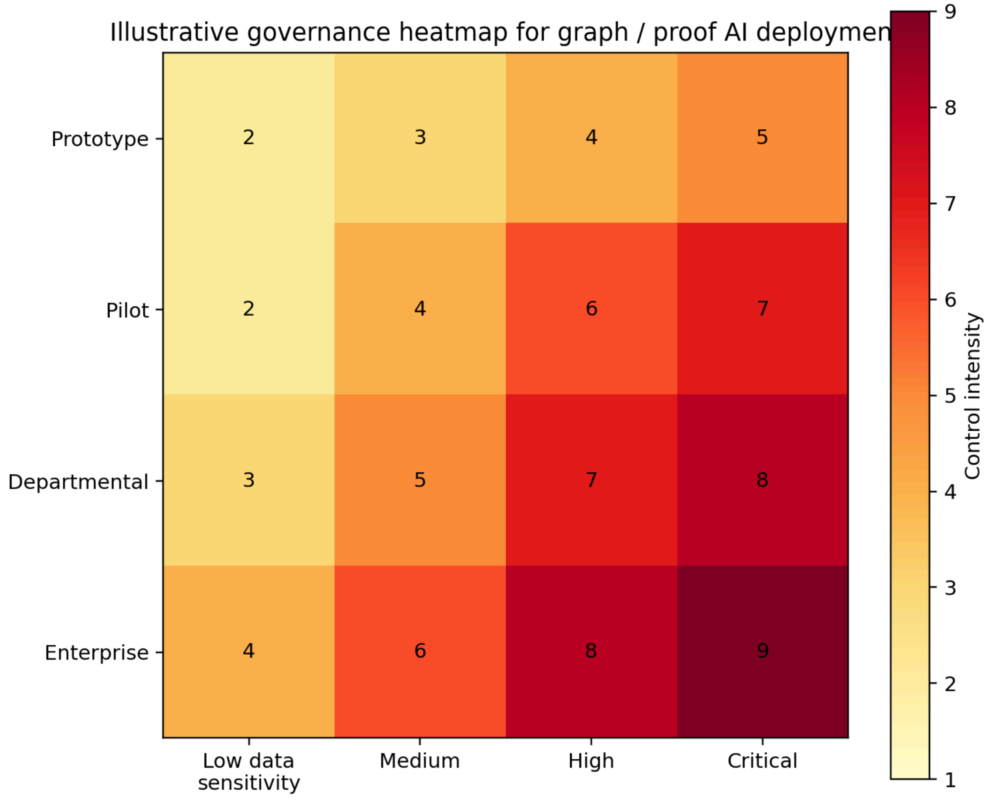

Figure 7. Illustrative governance heatmap for scaling graph and proof AI from prototype to enterprise use.

Control area	Good practice	Why it matters	Framework anchor
Data governance	Minimize sensitive links, document provenance, apply access controls	Graphs can reveal hidden personal or strategic relationships	NIST AI RMF; ISO/IEC 42001
Model governance	Benchmark GNNs and proof agents for calibration, robustness, and failure modes	Graph and proof systems may fail under distribution shift or sparse evidence	NIST AI RMF; OECD
Human oversight	Require review for high-impact outputs and ambiguous proofs	Formal correctness does not remove contextual judgment needs	UNESCO; OECD
Auditability	Preserve proof traces, graph lineage, and decision logs	Traceability supports assurance, teaching, and incident review	ISO/IEC 42001; NIST AI RMF

8. Teaching sequence for delivery

Week / module	Topic	Hands-on activity	Output
1	Graph fundamentals	Build a node-edge model for a simple domain	Adjacency list and visualization
2	Core algorithms	Run shortest path, centrality, and community detection	Algorithm comparison notes
3	Knowledge graphs + GNNs	Train a small graph classifier or link predictor	Notebook and model card
4	Computational proof AI	Represent a logic or geometry problem as proof states	Proof trace demo

5	Governance and evaluation	Apply risk checklist and documentation template	Mini evaluation report
6	Capstone integration	Combine method, domain, and controls	Proposal + prototype plan

9. Discussion prompts

- When should a graph-based AI system use statistical learning alone, and when should it add symbolic or proof-based constraints?
- What trade-offs arise between expressive relational modeling and privacy protection?
- How can proof traces improve trust without creating false confidence in real-world decisions?
- Which AMK Research Lab capstone topic would best demonstrate both technical depth and governance maturity?

10. References

NIST. (2023). Artificial Intelligence Risk Management Framework (AI RMF 1.0). U.S. Department of Commerce.

OECD. (2024). OECD AI Principles.

UNESCO. (2021/2024). Recommendation on the Ethics of Artificial Intelligence.

ISO/IEC. (2023). ISO/IEC 42001: Artificial intelligence management system.

Liang, F., et al. (2022). Survey of graph neural networks and applications. NIST.

Khemani, B., et al. (2024). A review of graph neural networks: concepts, architectures, techniques, challenges, datasets, applications and future directions. Journal of Big Data.

Trinh, T. H., et al. (2024). Solving olympiad geometry without human demonstrations. Nature.

Hubert, T., et al. (2025). Olympiad-level formal mathematical reasoning with AlphaProof and AlphaGeometry 2. Nature.