Science Discovery Artificial Intelligence
2026, Vol. 1, No. 3, pp. 117–129
https://doi.org/10.11648/j.sdai.20260103.12

Review Article

Teaching Students to Verify Generative AI Using the TRACE-AI Framework

Khama Valentine Arinze[1], Umakalu Chioma Precious[1], Ifeanyichukwu Martins Ordi[1], Adesegun Nurudeen Osijirin[1, 2, *]

[1]Department of Mathematics and Computer Science Education, Enugu State University of Science and Technology, Enugu, Nigeria

[2]Department of Healthcare Management, Federal University of Allied Health Sciences, Enugu, Nigeria

Abstract

Generative artificial intelligence (GenAI) can produce polished mathematical solutions, executable code, persuasive media, and professional recommendations, yet fluency does not establish correctness, provenance, security, or contextual fitness. Existing AI-literacy frameworks appropriately describe broad competencies, while emerging studies in mathematics, programming, and health professions education show the value of domain-specific validation. What remains underdeveloped is a portable instructional process that makes verification, uncertainty, and reliance observable across disciplines. This conceptual article introduces TRACE-AI: Translate the task and stakes; Represent the problem, assumptions, and constraints; Audit the AI's reasoning and output; Corroborate pivotal claims through independent evidence, tests, or tools; Estimate uncertainty and warranted reliance; Act proportionately by accepting, revising, rejecting, or escalating; and Internalize learning through reflection and unaided transfer. The framework was developed through a theory-informed critical synthesis of AI literacy, mathematical problem solving, computational thinking and debugging, epistemic cognition, self-regulated learning, and trust calibration. Each stage produces an assessable artefact, and a risk gate increases verification and human oversight as uncertainty, irreversibility, or potential harm rises. Worked applications in mathematics, programming/cybersecurity, and health-professional reasoning demonstrate invariant functions with discipline-specific standards of proof. An assessment architecture distinguishes final-answer quality from process quality, appropriate reliance, confidence calibration, verification quality, and delayed unaided transfer. Six propositions define a research programme for comparative and randomized evaluation. TRACE-AI is not presented as an effectiveness-tested intervention; it is a transdisciplinary curriculum and research architecture requiring contextual adaptation and empirical validation. Its contribution is to move GenAI education beyond prompt production toward disciplined human judgment about when AI output deserves trust.

Keywords

Generative Artificial Intelligence, AI Literacy, Mathematical Problem Solving, Computational Thinking, Verification, Trust Calibration, Cybersecurity Education, Self-regulated Learning

*Correspondence: Adesegun Nurudeen Osijirin (adesegunosijirin@fuahse.edu.ng)

Received: 3 August 2026; **Accepted:** 17 August 2026; **Published:** 9 November 2026

Copyright: © The Author(s), 2026. Published by Science Publishing Group. This is an **Open Access** article, distributed under the terms of the Creative Commons Attribution 4.0 License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution and reproduction in any medium, provided the original work is properly cited.

1. Introduction

Generative artificial intelligence (GenAI), a form of artificial intelligence (AI), has made sophisticated-looking solutions available before many students have developed the knowledge needed to judge them. A prompt can return a proof, a program, a statistical explanation, a patient-management suggestion, or a cybersecurity recommendation in seconds. The educational problem is not simply that these outputs can be false. It is that they can be partly correct, elegantly presented, and consequentially wrong in ways that invite premature trust. Large language models (LLMs) may fabricate references, omit assumptions, conceal uncertainty, reproduce bias, or offer technically valid advice that is unsuitable for the learner's context [11, 12]. Fluency lowers the subjective cost of acceptance; it does not increase the epistemic warrant of the answer.

AI literacy scholarship has established that students need more than operational skill. Influential frameworks encompass understanding AI, using and evaluating it, creating with it, and engaging with its ethical and societal implications [15, 17, 18, 20]. A recent systematic review found GenAI literacy to be multidimensional across educational and organisational settings, with competencies spanning technical, cognitive, communicative, and ethical domains [25]. Critical AI-literacy scholarship further foregrounds power, agency, data, fairness, and sociotechnical context [32]. These are essential curricular destinations. However, a competency taxonomy does not by itself specify what a student should do, in what order, when facing a plausible AI-generated answer under time, evidence, and risk constraints.

At the same time, discipline-specific research is beginning to reveal what productive verification can look like. Mathematics educators have used rational questioning to move students from result matching toward criterion-guided validation of AI-generated solutions [36]. Programming students value GenAI but can find AI-generated code more difficult to correct than instructor-designed tasks [13], while structured GenAI-assisted debugging appears to reorganise patterns of critical thinking without uniformly improving all performance components [1]. In high-school mathematics, access to an unguarded GenAI interface improved assisted performance but harmed subsequent unaided performance; safeguards reduced that harm [3]. These studies converge on an important distinction: a high-quality AI-assisted product is not necessarily evidence of learning, verification competence, or appropriate reliance.

The conceptual gap lies between broad literacy frameworks and domain-specific teaching techniques. Higher education needs a transdisciplinary mechanism that preserves the evidentiary standards of each subject while giving students a stable routine for deciding whether AI output warrants action. This article proposes TRACE-AI: Translate, Represent, Audit, Corroborate, Estimate, Act, and Internalize. The framework integrates mathematical problem solving, computational thinking and debugging, epistemic cognition, self-regulation, and trust calibration. It is intended for university and teacher-education contexts, with worked illustrations in mathematics, programming/cybersecurity, and health-professional reasoning.

The article makes four contributions. First, it identifies verification as the missing procedural layer between AI-literacy competence and disciplinary performance. Second, it defines seven learner actions that each yield assessable evidence. Third, it introduces a risk gate that scales corroboration and human oversight to uncertainty and potential harm. Fourth, it specifies outcomes and propositions for empirical evaluation, including appropriate reliance and delayed unaided transfer. TRACE-AI is a theory-informed conceptual framework, not an effectiveness-tested intervention. No primary data are reported.

2. Approach to Framework Development

TRACE-AI was developed through a theory-informed critical synthesis. The purpose was conceptual integration and design translation, not exhaustive evidence retrieval, meta-analysis, or estimation of intervention effects. Literature was purposively selected from six intersecting bodies of work: AI and GenAI literacy; mathematical problem solving and validation; computational thinking, programming, and debugging; epistemic cognition and vigilance; self-regulated learning and cognitive offloading; and human-automation trust and risk. Seminal theory was combined with recent reviews and empirical studies published through July 2026. Backward and forward citation tracing was used to test whether the proposed functions were already represented in adjacent frameworks.

Source selection was iterative rather than systematic. Candidate sources were retained when they contributed a distinct construct, instructional mechanism, empirical boundary condition, or disciplinary verification practice; duplicative or product-specific items without transferable instructional implications were not prioritised. No formal risk-of-bias score was applied, and the synthesis makes no claim of exhaustive coverage.

Framework construction followed three design questions. What learner actions are required before, during, and after AI consultation? Which elements should remain invariant across disciplines, and which must be governed by disciplinary standards? What evidence would distinguish successful task completion from warranted reliance and durable learning? Candidate stages were retained only if they satisfied four criteria: each had a distinct instructional function; each generated an observable artefact; omission created an identifiable failure mode; and the function could transfer across at least mathematics, computing, and a professional decision domain.

The resulting framework is deliberately mid-range. It does not replace comprehensive AI-literacy frameworks, a mathematics curriculum, secure software-development practices, clinical guidelines, or institutional AI policy. Instead, it connects them at the point of learner action. It also avoids product-specific instructions: model labels, interfaces, and capabilities change, whereas the need to define the task, represent the problem, test claims, estimate uncertainty, and justify reliance persists.

Because this conceptual article synthesises published literature and reports no primary data or research involving human participants, animals, or identifiable records, institutional ethics approval and informed consent were not required. A generative AI system was used for structural development, drafting, and language editing; it did not make accountable decisions. The authors verified all scholarly claims and references and remain responsible for the conceptual contribution, disciplinary accuracy, interpretation, and final text.

3. The Conceptual Problem: From Competent Use to Warranted Reliance

3.1. Why Prompting Is an Incomplete Educational Target

Prompt construction can improve the relevance, structure, or transparency of an output, but it cannot guarantee truth, valid inference, secure code, authentic provenance, or contextual appropriateness. A student may write an excellent prompt and still receive a flawed solution. More subtly, a student may receive a correct answer but bypass the retrieval, representation, and error-monitoring processes through which expertise develops. Teaching prompt templates without a verification architecture risks equating productive interaction with justified belief.

This distinction matters because assistance can raise performance while it is available and reduce performance after it is removed. Cognitive offloading is useful when a tool extends human capability, but educationally hazardous when it substitutes for the cognitive process being taught [26]. The design challenge is not to eliminate offloading. It is to decide which work may be delegated, which must remain human, and how the learner will know whether the delegated result is dependable.

3.2. Verification Is Disciplinary and Transdisciplinary

Verification has a common structure but different standards. A mathematical claim may be checked against definitions, theorem conditions, symbolic transformations, counterexamples, or independent computation. Code may be evaluated through tracing, compilation, unit tests, boundary cases, static analysis, dependency inspection, and security review. A professional recommendation may require current guidelines, patient or client context, local policy, and accountable human oversight. The invariant educational question is whether the evidence is strong enough for the intended use; the answer depends on the domain.

This is why generic fact-checking is insufficient. Verification requires claim prioritisation, source or test selection, interpretation, and a decision about residual uncertainty. It is both epistemic and practical. TRACE-AI therefore separates Audit, an internal examination of the AI output, from Corroborate, an independent test using evidence or tools not controlled by the same output. Asking the model to confirm itself may reveal inconsistency, but it is not independent corroboration.

3.3. Awareness Does Not Guarantee Detection

Students may know that AI can hallucinate yet remain unable to locate a subtle error in a proof, identify an insecure dependency, or recognise a fabricated audiovisual claim. Prior work in Nigerian tertiary institutions similarly found a gap between reported digital and cybersecurity awareness and practical detection of manipulated media [23]. Awareness is therefore a necessary but weak proxy for verification performance. Instruction must place learners in situations where credibility, correctness, and consequence are judged through observable action.

4. Theoretical Foundations

Table 1. Disciplinary and theoretical contributions to TRACE-AI.

Knowledge base	Core contribution	Most visible stages	Failure made visible
Mathematics education	Problem representation, heuristic selection, logical justification, modelling, checking	Translate; Represent; Audit; Corroborate	A correct result with an invalid transformation or an unstated domain restriction
Computer science education	Decomposition, abstraction, algorithmic thinking, testing, debugging	Represent; Audit; Corroborate; Act	Executable code that fails at boundaries, exposes data, or violates requirements

Knowledge base	Core contribution	Most visible stages	Failure made visible
Epistemic cognition and vigilance	Epistemic aims, standards, reliable processes, source and testimony evaluation	Audit; Corroborate; Estimate	Believing a plausible explanation because it is fluent or cited-looking
Trust calibration and risk	Appropriate reliance, automation bias, consequence-sensitive oversight	Estimate; Act	Following advice despite weak evidence in a high-stakes situation
Self-regulated learning	Forethought, monitoring, reflection, strategy adaptation, transfer	Translate; Represent; Internalize	Completing an assisted task without being able to solve a related task independently

4.1. Mathematical Problem Solving: Represent Before Calculating

Pólya's familiar sequence—understand the problem, devise a plan, carry out the plan, and look back—established problem solving as more than answer production [24]. Schoenfeld [28] added the importance of resources, heuristics, control, and beliefs, while mathematical competency perspectives emphasise modelling, representation, reasoning, symbols, communication, and tools [21]. Empirical work in the Nigerian context has also shown the instructional value of Pólya-oriented problem solving for algebra achievement and retention [22]. TRACE-AI retains this discipline's insistence that the solver construct a mathematical object before accepting a procedure.

Represent is therefore not a paraphrasing exercise. Learners specify quantities, relationships, units, assumptions, constraints, and the form a legitimate answer must take. Audit then examines whether the AI solution preserves these elements, applies rules under valid conditions, and supplies warrants rather than only steps. Corroboration can use substitution, alternative derivation, graphing, dimensional checks, or counterexamples. Recent mathematics-education research supports moving students from intuitive agreement and result matching toward explicit criteria for validating AI-generated solutions [36].

4.2. Computational Thinking and Debugging: Treat Output as a Testable Artefact

Computational thinking frames problem solving through practices such as decomposition, abstraction, algorithmic design, representation, iteration, and evaluation [8, 29, 34]. In mathematics and science classrooms, these practices include data work, modelling and simulation, computational problem solving, and systems thinking [33]. TRACE-AI imports a particularly important computing disposition: an output is not trusted because it looks finished; it is subjected to tests designed to make it fail.

Debugging is epistemically valuable because it requires hypotheses about failure, evidence from program behaviour, and iterative correction. This is directly relevant to GenAI, which may produce code that compiles yet is inefficient, brittle, insecure, poorly documented, or mismatched to requirements. Students report favourable perceptions of GenAI while struggling disproportionately to correct LLM-generated code [13]. A systematic review of GenAI in programming education likewise identifies the need to preserve foundational logic and higher-order thinking through intentional integration and assessment [19]. TRACE-AI uses test cases, trace tables, linters, static analysis, peer review, and security checks as forms of independent corroboration rather than optional finishing touches.

4.3. Epistemic Cognition and Vigilance: Evaluate How Knowledge Claims Earn Acceptance

Epistemic cognition concerns aims such as truth or understanding, standards for judging whether those aims have been met, and processes considered reliable for doing so [2, 5]. Epistemic vigilance concerns the cognitive and social mechanisms through which people evaluate communicated information and its source [30]. GenAI complicates both: the apparent speaker is a system that can synthesise language without stable belief, lived responsibility, or direct access to every fact it states.

TRACE-AI makes the epistemic basis of acceptance explicit. Students identify the claim, the standard that should govern it, the evidence used, and the uncertainty that remains. Source evaluation includes authority, currency, independence, relevance, and contextual fit. A citation's existence is only the first check; it must actually support the claim. The aim is not universal scepticism but warranted belief proportionate to evidence.

4.4. Trust Calibration and Risk: Reliance Should Track Capability and Consequence

Trust in automation is appropriately calibrated when reliance corresponds to what the system can do in the relevant context [9, 14]. Automation bias can produce commission errors, in which incorrect advice is followed, and omission errors, in which unflagged information is overlooked [16]. Cognitive

forcing functions—such as requiring an initial judgment or justification before viewing AI advice—can reduce overreliance, although they add effort [4]. These insights motivate the ordering of TRACE-AI: learners define and represent before exposure to an answer, then make reliance explicit after corroboration.

Reliance is also risk-sensitive. The evidentiary threshold for accepting a wording suggestion is not the threshold for accepting a medication dose, a security configuration, or a statistical conclusion that will drive policy. TRACE-AI's risk gate increases corroboration and human oversight when potential harm, irreversibility, uncertainty, or local variation rises. The framework therefore resists both blind trust and reflexive distrust. Correct, low-risk assistance may be accepted efficiently; consequential claims require stronger evidence and, where appropriate, an accountable expert.

4.5. Self-regulated Learning: Preserve the Learner's Agency and Transfer

Self-regulated learning involves forethought, performance monitoring, strategic control, and reflection [35]. TRACE-AI distributes these functions across the cycle. Translate establishes the goal; Represent externalises the learner's initial understanding; Audit and Corroborate monitor progress; Estimate and Act regulate reliance; Internalize supports reflection and transfer. The final stage is indispensable because assisted success can conceal fragile knowledge. A related, delayed task completed without AI tests whether the learner has acquired a strategy rather than merely borrowed an answer.

5. The TRACE-AI Framework

TRACE-AI is a seven-stage cycle (Figure 1). The sequence is initially explicit and instructor-scaffolded. As learners become more capable, prompts and approved-source lists can fade, but the functions remain. A cycle may return from Corroborate to Represent when evidence reveals that the problem was incorrectly framed, or from Act to Audit when a proposed revision introduces a new claim.

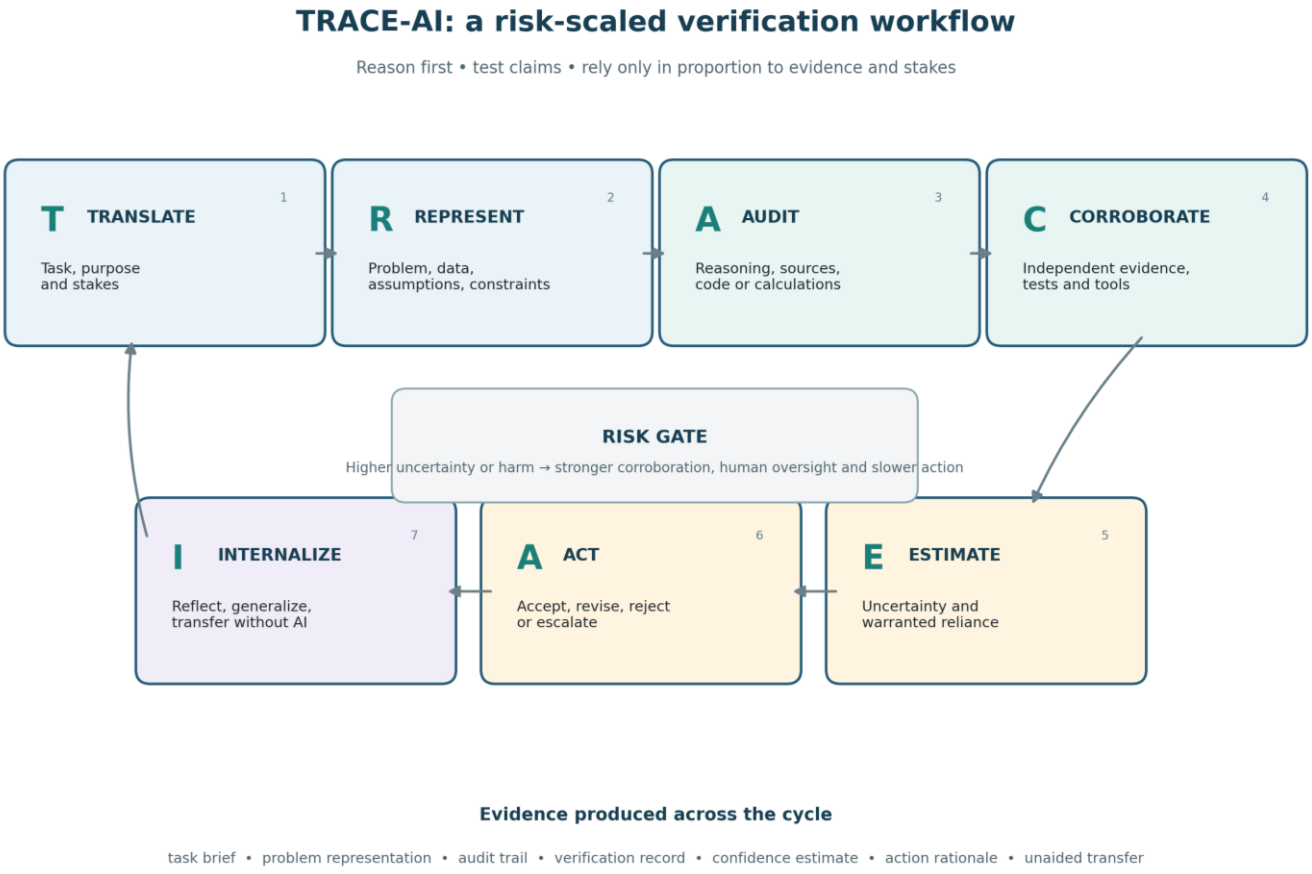


Figure 1. *The TRACE-AI verification cycle. Each stage produces an observable artefact. The central risk gate scales verification and human oversight to uncertainty, irreversibility, and potential harm. Original framework proposed in this article.*

Table 2. Operational specification of the TRACE-AI stages.

Stage	Function	Guiding questions	Observable artefact	Failure prevented
1. Translate	Define the task, audience, intended use, and stakes.	What exactly must be decided or produced? Who may be affected? What counts as success?	Task-and-stakes brief	Solving a different, easier, or lower-stakes problem
2. Represent	Externalise the problem before accepting AI framing.	What are the data, relationships, assumptions, constraints, unknowns, and initial expectations?	Problem representation or unaided baseline	Anchoring on the model's formulation; missing units or requirements
3. Audit	Interrogate the internal quality of the output.	Which claims, steps, sources, calculations, code paths, and omissions are pivotal?	Annotated output and issue log	Equating fluency, length, citations, or execution with correctness
4. Corroborate	Test pivotal elements independently.	Which theorem, source, alternative method, test case, tool, or expert can independently challenge this claim?	Claim-test-evidence-verdict record	Asking the same model to verify itself; checking only the final answer
5. Estimate	Judge residual uncertainty and warranted reliance.	How strong and independent is the evidence? What remains unresolved? How costly is error?	Confidence estimate with risk classification	False precision; confidence driven by presentation rather than evidence
6. Act	Choose a proportionate response.	Should the output be accepted, revised, rejected, or escalated? What safeguards are required?	Action decision and rationale	Using partially verified output beyond its safe scope
7. Internalize	Convert assistance into transferable learning.	What rule, misconception, test, or strategy should be retained? Can I solve a related task without AI?	Reflection plus unaided transfer task	Submission without learning; repeated dependence

5.1. T — Translate the Task and Stakes

Students first convert an assignment or problem into a decision statement. They identify the intended audience, use, time horizon, and consequences of error. A mathematics homework explanation, a program deployed with personal data, and advice affecting patient care cannot share the same verification threshold. The learner assigns a provisional risk level—low, moderate, or high—and explains it. Faculty should grade the justification, not only the label.

Operational risk-gate criteria are as follows. Low risk applies when the output is reversible, affects only a draft or practice task, has no sensitive data or third-party consequence, and can be checked readily; use at least one targeted check. Moderate risk applies when the output may influence assessment, shared data or code, or a consequential interpretation but remains reversible and reviewable; use two independent checks, or one authoritative source plus a direct test. High risk applies when error could cause material harm, expose sensitive data, compromise security, violate law or professional duty, or trigger an irreversible action; do not act without authoritative corroboration and accountable human review. When criteria conflict, assign the highest applicable level.

5.2. R — Represent the Problem, Assumptions, and Constraints

Before viewing or accepting an AI solution, students construct an unaided representation appropriate to the discipline. In mathematics this may include a diagram, variables, units, domain restrictions, and an expected range. In programming it may include inputs, outputs, invariants, data types, threat assumptions, and test cases. In professional reasoning it may include the person's context, priorities, missing information, and applicable policy. This stage creates a baseline against which AI influence can later be observed.

5.3. A — Audit the AI's Reasoning and Output

Audit converts a polished answer into inspectable components. Students mark factual claims, inferential steps, citations, assumptions, code branches, calculations, and omissions. They identify which components are pivotal—meaning that an error could change the conclusion or action. The audit is adversarial but not performatively sceptical: the purpose is to locate vulnerability, not to prove the system wrong. A useful rule is to challenge the strongest-looking claim as well as the most doubtful one, because fabricated confidence often clusters around technical detail.

5.4. C — Corroborate with Independent Evidence, Tests, or Tools

Students test pivotal claims using a method that does not merely repeat the original generation process. Mathematical corroboration may involve an alternative proof, substitution, graph, dimensional analysis, or a counterexample. Programming corroboration may involve hand tracing, compilation, unit and property-based tests, static analysis, dependency checks, or peer review. Professional corroboration may involve authoritative guidance, primary literature, local policy, or an accountable specialist. Evidence is recorded as supporting, contradicting, partly supporting, or unresolved. Multiple weak sources do not necessarily equal one strong, directly relevant source.

5.5. E — Estimate Uncertainty and Warranted Reliance

The learner estimates confidence after corroboration and separates three questions: How likely is the claim to be correct? How complete is it for this context? How safe is it to act on? A response may be likely correct but incomplete, or correct for a generic case but unsafe locally. Confidence should be expressed with a brief evidence-based rationale rather than as a bare percentage. Instructors can compare pre-AI and post-verification confidence with objective performance to identify overconfidence and unproductive deference.

5.6. A — Act Proportionately to Risk

Four actions are available: accept, revise, reject, or escalate. Acceptance means the evidence is adequate for the stated use, not that the system is generally trustworthy. Revision means useful components are retained while errors, omissions, or contextual mismatches are corrected. Rejection is appropriate when pivotal claims fail. Escalation is appropriate when uncertainty remains consequential, required evidence is inaccessible, or the learner lacks authority. The action record states what may be used, what may not, and under what conditions.

5.7. I — Internalize Through Reflection and Unaided Transfer

Students articulate what the verification process changed: a misconception corrected, a test that exposed failure, a source criterion learned, or a risk cue that altered action. They then complete a structurally related task without GenAI, ideally after a delay. Internalize prevents the framework from becoming a compliance checklist. Its success is demonstrated when the learner independently deploys a better representation, verification test, or escalation decision in a new context.

6. Cross-disciplinary Applications

Table 3. TRACE-AI applications across disciplinary contexts.

Context	Illustrative AI vulnerability	Independent corroboration	Risk-proportionate action	What assessment should detect
Mathematics: conditional probability	AI applies Bayes' theorem but confuses sensitivity with the posterior probability and omits the base rate.	Represent events and prevalence; recompute using a probability tree or natural frequencies; test limiting cases.	Revise the solution and explain why the base rate changes interpretation.	Correct number with invalid reasoning, missing domain conditions, or unsupported generalisation
Programming: authentication function	AI-generated code runs but stores passwords insecurely and fails on malformed input.	Trace data flow; create boundary/adversarial tests; run linter and dependency/security checks; compare secure-coding guidance.	Reject unsafe components, revise with approved library, and request security review before deployment.	Compilation treated as correctness; security and privacy omitted
Cybersecurity/media: urgent voice message	A persuasive audio clip appears to request a confidential transfer from a known authority.	Inspect provenance and channel; verify through a separate trusted contact; check contextual inconsistencies; preserve evidence.	Do not comply; escalate through institutional incident procedure.	Familiarity and urgency substituted for authentication
Health-professional reasoning: dosage suggestion	AI provides a plausible dose but ignores renal impairment and a locally used formulary.	Reconstruct patient factors; calculate independently; check current formulary/guideline; consult accountable clinician/pharmacist.	Reject or revise; escalate before any patient-facing action.	Correct generic advice applied to the wrong patient or jurisdiction

6.1. Mathematics Education

A teacher can provide a calculus, algebra, statistics, or modelling problem together with an AI-generated solution that is correct, partly correct, or subtly flawed. Students must first represent the problem and predict features of the answer before examining the output. The audit should target definitions, theorem conditions, logical transitions, notation, efficiency, and communicative adequacy. Corroboration should include at least one method not copied from the AI. This design extends Pólya-style looking back into an environment where the apparent solver is external and may be unreliable.

Variation is important. If every AI solution contains a planted error, students learn ritual distrust. A balanced sequence includes correct solutions, incorrect solutions, correct results reached invalidly, and solutions that are mathematically sound but poorly communicated. Students should sometimes accept AI output after adequate testing. Calibration, not rejection, is the target.

6.2. Computer Science and Cybersecurity Education

In programming, the unit of verification should be broader than whether code executes. Requirements, logic, complexity, readability, security, privacy, licensing, dependencies, and maintainability may all be consequential. Learners can receive AI-generated code with one or more seeded vulnerabilities and build a test plan before running it. Audit artefacts may include an annotated code review, threat model, test suite, and explanation of each correction. The framework is compatible with pair programming and peer review: one learner acts as implementer and another as verifier, then roles rotate.

Cybersecurity cases make the risk gate concrete. An AI-generated incident response, phishing classification, or access-control configuration can be evaluated against a threat model and institutional policy. Social-engineering and deepfake cases also demonstrate that polished language and familiar identity cues are not authentication. Independent-channel verification, provenance checks, and escalation procedures are forms of corroboration. This connects technical education with digital and media literacy rather than treating them as separate awareness topics.

6.3. Health-professional and Other Professional Education

Professional programmes add accountability, contextual variation, and potential harm. A plausible recommendation may be wrong because of missing client information, local resources, law, scope of practice, or current guidance. Health-professions studies show susceptibility to incorrect clinical decision support, the importance of calibrated trust, and the possibility that access to a capable model may not improve users' reasoning without effective integration [6, 7, 27]. TRACE-AI requires patient or client data to be represented before consultation, prohibits entry of identifiable information into unapproved systems, and routes high-risk unresolved claims to accountable human experts. The same logic can support law, engineering, management, and teacher education.

7. Curriculum Architecture

7.1. A Progression From Worked Validation to Independent Judgment

TRACE-AI should be taught as a progression rather than a one-off checklist. In an introductory phase, the instructor models each stage using a short case, supplies approved sources or tests, and makes the risk gate explicit. In guided practice, students work with partially completed templates and compare verification decisions. In independent practice, prompts fade, cases become ambiguous, and learners choose evidence. In transfer, students encounter a new domain or format and complete the task without AI assistance. Scaffolding should fade only when process evidence, not self-reported confidence, shows readiness.

7.2. A Feasible Learning Sequence

A 100-minute seminar or laboratory can introduce the full cycle: orientation and risk rules (10 minutes), Translate and unaided Represent (15 minutes), Audit (15 minutes), Corroborate (25 minutes), Estimate and Act (10 minutes), peer comparison and instructor debrief (20 minutes), and Internalize reflection (5 minutes). A related unaided transfer task can follow 7-14 days later. In resource-constrained settings, the AI output may be generated and printed by the instructor; students can still audit, corroborate, and calibrate without individual paid accounts or continuous connectivity.

7.3. Designing a Balanced Case Bank

Cases should vary along five dimensions: output accuracy, error subtlety, task stakes, information completeness, and learner expertise. Error types may include false facts, invalid inference, omitted constraints, outdated evidence, fabricated citations, insecure code, biased framing, or locally inapplicable advice. Correct outputs must also appear so that acceptance can be learned and underreliance measured. Model outputs should be captured close to teaching time because systems change. Where comparability matters, faculty can use frozen outputs rather than live generation.

8. Assessment Architecture

A final answer score cannot reveal how the student arrived

there. TRACE-AI assessment therefore separates product, process, reliance, calibration, and transfer. A learner can reach the correct answer by unsafe copying, or reach an imperfect answer after a defensible response to genuinely ambiguous evidence. Both require feedback, but not the same feedback.

Table 4. TRACE-AI assessment blueprint.

Construct	Observable indicator	Interpretive value
Task translation	Accuracy of purpose, audience, stakes, and success criteria	Underestimating consequence or solving the wrong problem
Problem representation	Completeness of data, relationships, assumptions, constraints, and unaided baseline	Anchoring and missing-context errors
Audit quality	Identification and prioritisation of claims, reasoning steps, code paths, sources, and omissions	Superficial error spotting
Corroboration quality	Independence, authority, relevance, currency, test coverage, and interpretation	Ceremonial checking or circular verification
Appropriate reliance	Acceptance of supported output plus rejection/revision/escalation of unsupported output	Overreliance and underreliance
Confidence calibration	Alignment between confidence and accuracy; evidence-based explanation of change	High-confidence error or unnecessary deference
Risk-proportionate action	Fit between stakes, residual uncertainty, safeguards, and chosen action	Using an answer beyond its verified scope
Unaided transfer	Performance on a related delayed task without GenAI	Assisted performance mistaken for learning
Implementation	Time, workload, fidelity, access, accessibility, privacy, and student experience	An effective design that cannot be equitably sustained

A concise analytic rubric can score each core process from 0 to 3: absent or unsafe; emerging but incomplete; competent and justified; or advanced, independent, and proportionate. Weighting should reflect the curriculum. In an introductory mathematics course, representation and mathematical corroboration may dominate. In secure programming, test coverage and risk action may carry greater weight. In a high-stakes professional case, escalation and contextual fit are non-negotiable.

Table 5. Completed TRACE-AI verification record and 0-3 process-rubric example.

Process stage	Completed evidence from a conditional-probability case	Score (0-3)
Translate	Instructional answer used for learning; an error is reversible but can reinforce a misconception, so the case is moderate risk.	3
Represent	Assume 10,000 people, prevalence 1% (100 people), sensitivity 90%, and specificity 95%; estimate P (condition | positive).	3
Audit	The AI reports 0.80 by treating sensitivity as the posterior probability and omits both the base rate and false positives.	3
Corroborate	Natural-frequency check: 90 true positives and 495 false positives; posterior = 90/(90 + 495) = 0.154.	3
Estimate	The 0.80 claim is contradicted; the corrected value is about 0.154. Confidence is high for the stated rates, but applicability is limited to them.	3
Act	Reject 0.80, revise the explanation, and avoid any inference beyond the given rates.	3
Internalize	The student explains the base-rate effect and solves an immediate related item unaided; delayed transfer is not yet demonstrated.	2

Process stage	Completed evidence from a conditional-probability case	Score (0-3)
Illustrative total	20/21: competent-to-advanced process evidence, with delayed transfer requiring follow-up.	20/21

This worked record is moderate risk because it is a reversible classroom exercise. The same numerical claim used for patient care would be high risk and could not be acted on without current clinical guidance and accountable professional review. Scores are illustrative and should be moderated against local descriptors.

Appropriate reliance requires exposure to both reliable and unreliable advice. Overreliance can be operationalised as accepting or acting on a consequentially incorrect AI claim, particularly when the learner's initial answer was correct or undecided. Underreliance is rejecting useful, adequately supported advice without a defensible reason. Confidence calibration can be assessed by comparing repeated confidence judgments with correctness, using calibration curves or Brier scores for suitable binary decisions. Transfer should be delayed and unaided; otherwise the design measures supported performance rather than retained capability.

9. Research Agenda, Implementation, and Discussion

TRACE-AI is intended to generate testable research rather than close debate. Six propositions follow from its mechanisms:

Proposition 1 — Process effect: Compared with unguided GenAI use, TRACE-AI will improve appropriate reliance and verification quality even when immediate final-answer scores are similar.

Proposition 2 — Representation effect: Requiring an unaided problem representation before AI exposure will reduce anchoring and harmful answer switching, particularly among learners with adequate prior knowledge.

Proposition 3 — Mediation: The effect of TRACE-AI on appropriate reliance will be mediated by the quality and independence of corroboration rather than by prompt sophistication alone.

Proposition 4 — Risk sensitivity: TRACE-AI will improve the alignment between verification effort and task stakes; its advantage over unguided use will be largest for consequential, ambiguous tasks.

Proposition 5 — Transfer: Guided verification followed by fading and unaided practice will produce stronger delayed transfer than unrestricted AI assistance, even when unrestricted assistance yields higher immediate performance.

Proposition 6 — Moderation and equity: Prior domain knowledge, epistemic beliefs, access, language, and instructor support will moderate outcomes; simplified templates and offline cases will reduce, but not eliminate, inequities.

9.1. Comparative and Randomized Designs

A rigorous initial test could compare three conditions: conventional problem- or case-based learning without GenAI; unguided GenAI-assisted learning; and TRACE-AI-guided learning. Randomization may occur at student, tutorial-group, or class level, with contamination considered. Outputs should be frozen or logged, assessors blinded where feasible, and tasks counterbalanced for AI correctness and error subtlety. Primary outcomes should be prespecified. Appropriate reliance, verification quality, and delayed unaided transfer are stronger primary candidates than satisfaction or immediate completion time.

Process data can reveal mechanism. Student representations, prompts, audit annotations, verification records, confidence changes, action decisions, and transfer responses can be analysed with validated rubrics. Think-aloud or interview studies may refine stage definitions before large trials. Multigroup measurement work is needed to establish whether assessment rubrics function similarly across disciplines, languages, and levels of expertise. Longitudinal studies should examine whether the explicit cycle becomes an internalised habit or remains a burdensome classroom ritual.

9.2. Minimum Reporting Standards

Studies should report the AI product, displayed model or version, access dates, retrieval capability, prompt and system constraints where known, source availability, case construction, planted error types, risk classification, instructor role, and permitted verification tools. They should distinguish live from frozen outputs and disclose whether students could consult additional models. Because commercial systems change, the intervention should be defined by educational functions and artefacts, not only by a product name.

9.3. Faculty Capability and Workload

Faculty must be able to identify pivotal claims, pretest cases, specify evidence standards, facilitate uncertainty, and give process feedback. Development is best shared across mathematics educators, computer science educators, subject specialists, librarians, and educational technologists. A reusable case bank, common rubric, and moderation examples can reduce

workload. Risk-management research in computing education similarly indicates that institutions must actively manage overreliance, bias, and skill loss while using GenAI's strengths [10]. Safe integration requires assessment redesign and staff development, not only access to a tool.

9.4. Access and Linguistic Equity

Requiring individual subscriptions or continuous broadband can reproduce existing inequalities. TRACE-AI can operate with instructor-provided outputs, shared devices, open-source or institutionally approved systems, and offline evidence packs. Cases should include multilingual and culturally situated contexts without assuming that the model's dominant-language response is universally appropriate. Accessibility also requires screen-reader-compatible templates, adequate time, alternative response formats, and explicit teaching of source navigation.

9.5. Privacy, Security, Intellectual Property, and Disclosure

Students must not enter identifiable patient, client, student, proprietary, or security-sensitive data into unapproved systems. Case materials should use fictional or appropriately de-identified data. Course policy should address data retention, intellectual property, authorship, disclosure, and prohibited uses in line with human-centred guidance for GenAI in education and research [31]. Verification records should document AI involvement without implying that disclosure alone makes use safe. In programming and cybersecurity tasks, pasting confidential code, credentials, or exploit details into public systems may itself constitute the principal risk.

9.6. Discussion

TRACE-AI reframes GenAI literacy as a problem of warranted reliance. Its central claim is that educational value does not come from prompting alone, nor from rejecting AI, but from making the learner's reasoning, tests, uncertainty, and action visible. The framework's transdisciplinary design is possible because mathematics and computer science share important epistemic practices: representation, decomposition, logical coherence, boundary testing, error diagnosis, and iteration. Professional education adds the necessary discipline of consequence, context, and accountable escalation.

The framework differs from broad AI-literacy models in granularity and purpose. It is not a complete account of what citizens should know about AI systems, data, ethics, power, or social impact. It is a workflow for a specific recurring event: a learner must judge whether an AI-generated claim or artefact is fit for a stated use. It differs from a fact-checking checklist by requiring an unaided representation, distinguishing internal audit from independent corroboration, scaling action to risk, and testing subsequent performance without assistance.

TRACE-AI also clarifies why awareness surveys are insufficient. Students can endorse the statement that AI is fallible while still accepting an invalid proof, insecure code, fabricated source, or unsafe recommendation. The framework makes these failures observable and teachable. It also allows appropriate acceptance to count as success. If an output survives proportionate testing and is suitable for the context, rejecting it merely because it came from AI is not critical thinking; it is miscalibrated distrust.

9.7. Limitations

Several limitations are important. First, TRACE-AI is conceptually derived and has not yet demonstrated effectiveness, feasibility, reliability, or cross-cultural validity. Second, seven explicit stages may impose cognitive and time costs, especially for novices or low-stakes tasks. The framework therefore requires scaffolding and risk-based compression rather than mechanical completion. Third, no generic workflow can substitute for disciplinary knowledge. Learners may not recognise what to test or how to interpret evidence without sufficient subject expertise. Fourth, the literature informing the synthesis is rapidly changing and uneven across disciplines; publication growth may reveal overlapping models or boundary conditions. Fifth, the mnemonic may imply a linearity that authentic problem solving does not possess. Iteration is expected, and empirical work should examine actual pathways.

Finally, TRACE-AI does not solve structural questions about commercial power, environmental cost, labour, surveillance, bias, or institutional governance. Those remain essential parts of critical AI literacy. The framework contributes a procedural layer that should be nested within, not substituted for, a broader human-centred curriculum.

10. Conclusion

Students need to learn more than how to obtain a useful answer from GenAI. They need to decide what problem is being solved, whether the answer's reasoning and evidence are adequate, how uncertainty interacts with consequence, and when a human must remain in control. TRACE-AI translates these requirements into seven teachable actions: Translate, Represent, Audit, Corroborate, Estimate, Act, and Internalize. Its value lies in combining mathematical validation, computational testing, epistemic vigilance, self-regulation, and risk-aware judgment without erasing disciplinary standards.

The framework is intentionally testable. Its next step is not rhetorical adoption but careful co-design and comparative evaluation across mathematics, computer science, cybersecurity, teacher education, and professional programmes. Until such evidence exists, TRACE-AI should be presented as a promising curriculum architecture rather than a proven intervention. The enduring educational goal is calibrated human judgment: trust AI output only to the extent that independent evidence, contextual fitness, and the stakes of action warrant it.

Abbreviations

AI	Artificial Intelligence
GenAI	Generative Artificial Intelligence
LLM	Large Language Model
TRACE-AI	Translate, Represent, Audit, Corroborate, Estimate, Act, and Internalize

Author Contributions

Khama Valentine Arinze: Conceptualization, Supervision, Validation, Writing – review & editing

Umakalu Chioma Precious: Investigation, Validation, Writing – review & editing

Ifeanyichukwu Martins Ordi: Methodology, Supervision, Validation, Writing – review & editing

Adesegun Nurudeen Osijirin: Conceptualization, Investigation, Methodology, Project administration, Validation, Visualization, Writing – original draft, Writing – review & editing

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] Albadarin, Y., Saqr, M., Pope, N., & Tukiainen, M. (2026). Evaluating the effect of generative AI-assisted debugging on students' critical thinking. Technology, Knowledge and Learning. https://doi.org/10.1007/s10758-026-09998-z

[2] Barzilai, S., & Chinn, C. A. (2018). On the goals of epistemic education: Promoting apt epistemic performance. The Journal of the Learning Sciences, 27(3), 353-389. https://doi.org/10.1080/10508406.2017.1392968

[3] Bastani, H., Bastani, O., Sungu, A., Ge, H., Kabakcı, Ö., & Mariman, R. (2025). Generative AI without guardrails can harm learning: Evidence from high school mathematics. Proceedings of the National Academy of Sciences, 122(26), e2422633122. https://doi.org/10.1073/pnas.2422633122

[4] Buçinca, Z., Malaya, M. B., & Gajos, K. Z. (2021). To trust or to think: Cognitive forcing functions can reduce overreliance on AI in AI-assisted decision-making. Proceedings of the ACM on Human-Computer Interaction, 5(CSCW1), Article 188. https://doi.org/10.1145/3449287

[5] Chinn, C. A., Buckland, L. A., & Samarapungavan, A. (2011). Expanding the dimensions of epistemic cognition: Arguments from philosophy and psychology. Educational Psychologist, 46(3), 141-167. https://doi.org/10.1080/00461520.2011.587722

[6] Gaube, S., Suresh, H., Raue, M., Merritt, A., Berkowitz, S. J., Lermer, E., Coughlin, J. F., Guttag, J. V., Colak, E., & Ghassemi, M. (2021). Do as AI say: Susceptibility in deployment of clinical decision-aids. npj Digital Medicine, 4, 31. https://doi.org/10.1038/s41746-021-00385-9

[7] Goh, E., Gallo, R., Hom, J., Strong, E., Weng, Y., Kerman, H., Cool, J. A., Kanjee, Z., Lane, K. P., Li, S., Ahuja, N., Horvitz, E., Yang, D., Chen, J. H., & Milstein, A. (2024). Large language model influence on diagnostic reasoning: A randomized clinical trial. JAMA Network Open, 7(10), e2440969. https://doi.org/10.1001/jamanetworkopen.2024.40969

[8] Grover, S., & Pea, R. (2013). Computational thinking in K-12: A review of the state of the field. Educational Researcher, 42(1), 38-43. https://doi.org/10.3102/0013189X12463051

[9] Hoff, K. A., & Bashir, M. (2015). Trust in automation: Integrating empirical evidence on factors that influence trust. Human Factors, 57(3), 407-434. https://doi.org/10.1177/0018720814547570

[10] Humble, N. (2024). Risk management strategy for generative AI in computing education: How to handle the strengths, weaknesses, opportunities, and threats? International Journal of Educational Technology in Higher Education, 21, Article 61. https://doi.org/10.1186/s41239-024-00494-x

[11] Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., & Fung, P. (2023). Survey of hallucination in natural language generation. ACM Computing Surveys, 55(12), Article 248. https://doi.org/10.1145/3571730

[12] Kasneci, E., Sessler, K., Küchemann, S., Bannert, M., Dementieva, D., Fischer, F., Gasser, U., Groh, G., Günnemann, S., Hüllermeier, E., Krusche, S., Kutyniok, G., Michaeli, T., Nerdel, C., Pfeffer, J., Poquet, O., Sailer, M., Schmidt, A., Seidel, T., Stadler, M., Weller, J., Kuhn, J., & Kasneci, G. (2023). ChatGPT for good? On opportunities and challenges of large language models for education. Learning and Individual Differences, 103, 102274. https://doi.org/10.1016/j.lindif.2023.102274

[13] Kohen-Vacs, D., Usher, M., & Jansen, M. (2025). Integrating generative AI into programming education: Student perceptions and the challenge of correcting AI errors. International Journal of Artificial Intelligence in Education, 35, 3166-3184. https://doi.org/10.1007/s40593-025-00496-4

[14] Lee, J. D., & See, K. A. (2004). Trust in automation: Designing for appropriate reliance. Human Factors, 46(1), 50-80. https://doi.org/10.1518/hfes.46.1.50_30392

[15] Long, D., & Magerko, B. (2020). What is AI literacy? Competencies and design considerations. In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (pp. 1-16). Association for Computing Machinery. https://doi.org/10.1145/3313831.3376727

[16] Lyell, D., & Coiera, E. (2017). Automation bias and verification complexity: A systematic review. Journal of the American Medical Informatics Association, 24(2), 423-431. https://doi.org/10.1093/jamia/ocw105

[17] Miao, F., & Cukurova, M. (2024). AI competency framework for teachers. UNESCO. https://unesdoc.unesco.org/ark:/48223/pf0000391104

[18] Miao, F., Shiohira, K., & Lao, N. (2024). AI competency framework for students. UNESCO. https://unesdoc.unesco.org/ark:/48223/pf0000391105

[19] Nathaniel, J., Oyelere, S. S., Suhonen, J., & Tedre, M. (2025). Literature review on the integration of generative AI in programming education. International Journal of Artificial Intelligence in Education, 35, 2724-2755. https://doi.org/10.1007/s40593-025-00524-3

[20] Ng, D. T. K., Leung, J. K. L., Chu, S. K. W., & Qiao, M. S. (2021). Conceptualizing AI literacy: An exploratory review. Computers and Education: Artificial Intelligence, 2, 100041. https://doi.org/10.1016/j.caeai.2021.100041

[21] Niss, M., & Højgaard, T. (2019). Mathematical competencies revisited. Educational Studies in Mathematics, 102, 9-28. https://doi.org/10.1007/s10649-019-09903-9

[22] Nneji, S. O. (2013). Effect of Polya George's problem solving model on students' achievement and retention in algebra. Journal of Educational and Social Research, 3(6), 41-48. https://doi.org/10.5901/jesr.2013.v3n6p41

[23] Osijirin, A. N., Sada, S. M., Edmond, V. U., Anigbo, L. C., & Okechukwu, O. (2026). AI-powered scams and deepfakes in tertiary institutions in Enugu State, Nigeria: The roles of cybersecurity awareness, digital literacy, and media literacy in students' fraud detection preparedness. Saudi Journal of Engineering and Technology, 11(4), 355-361. https://doi.org/10.36348/sjet.2026.v11i04.021

[24] Pólya, G. (1957). How to solve it: A new aspect of mathematical method (2nd ed.). Princeton University Press. (Original work published 1945).

[25] Reicho, M., Otrel-Cass, K., Ebner, M., Brünner, B., Freinhofer, D., & Irfan, J. (2026). Generative AI literacy across education and business: Competencies, obstacles, and benefits—a systematic literature review. International Journal of Educational Technology in Higher Education, 23, Article 23. https://doi.org/10.1186/s41239-026-00596-8

[26] Risko, E. F., & Gilbert, S. J. (2016). Cognitive offloading. Trends in Cognitive Sciences, 20(9), 676-688. https://doi.org/10.1016/j.tics.2016.07.002

[27] Sakamoto, T., Harada, Y., & Shimizu, T. (2024). Facilitating trust calibration in artificial intelligence-driven diagnostic decision support systems for medical students: Quasi-experimental study. JMIR Formative Research, 8, e58666. https://doi.org/10.2196/58666

[28] Schoenfeld, A. H. (1985). Mathematical problem solving. Academic Press.

[29] Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. Educational Research Review, 22, 142-158. https://doi.org/10.1016/j.edurev.2017.09.003

[30] Sperber, D., Clément, F., Heintz, C., Mascaro, O., Mercier, H., Origgi, G., & Wilson, D. (2010). Epistemic vigilance. Mind & Language, 25(4), 359-393. https://doi.org/10.1111/j.1468-0017.2010.01394.x

[31] UNESCO. (2023). Guidance for generative AI in education and research. https://unesdoc.unesco.org/ark:/48223/pf0000386693

[32] Veldhuis, A., Lo, P. Y., Kenny, S., & Antle, A. N. (2025). Critical artificial intelligence literacy: A scoping review and framework synthesis. International Journal of Child-Computer Interaction, 43, 100708. https://doi.org/10.1016/j.ijcci.2024.100708

[33] Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. Journal of Science Education and Technology, 25, 127-147. https://doi.org/10.1007/s10956-015-9581-5

[34] Wing, J. M. (2006). Computational thinking. Communications of the ACM, 49(3), 33-35. https://doi.org/10.1145/1118178.1118215

[35] Zimmerman, B. J. (2002). Becoming a self-regulated learner: An overview. Theory Into Practice, 41(2), 64-70. https://doi.org/10.1207/s15430421tip4102_2

[36] Zhuang, Y. (2026). Empowering students to critically validate AI-generated mathematical solutions through the rational questioning approach. Educational Studies in Mathematics. https://doi.org/10.1007/s10649-026-10528-y