

FUNDAMENTALS OF AI-AUGMENTED ENGINEERING

CHAPTER 4 / HOMEWORK 4

Engineering Homework

The Channel That Must Not Overflow — A Freeboard Certification

Trap It · Hunt It · Trust It · A Guarantee Is a Contract

Suggested Assessment · 20 points

target effort ≈ 70–90 minutes · type all answers and paste the required screenshot directly into this document



AI First Engineering

Figure: AI First
Engineering
(AIFE) course
logo

Taner Onsay, Ph.D. & Sebnem Onsay · © 2026 AI First Engineering ·

AI use is allowed and encouraged — as an Engineering Analyst that may now propose a method. But every prediction must be committed in writing before you compute, every bracket must be certified by you, the screenshot must show YOUR OWN Python session running on your computer — not an AI transcript — and every AI claim you record must carry your own ACCEPT / MODIFY / REJECT verdict. Assessment rewards evidence + certification + verification, not polished prose.

THE ENGINEERING CONTEXT — a drainage channel with a freeboard requirement

A rectangular concrete drainage channel carries stormwater away from a new development: width $b = 2.0$ m, total depth $D = 1.20$ m, longitudinal slope $S = 0.002$ (a 2-m drop per kilometre), finished-concrete roughness $n = 0.015$. The design storm delivers $Q_{\text{design}} = 3.0$ m³/s. The city's drainage standard is blunt: at the design flow the water surface must leave at least 0.30 m of FREEBOARD below the top of the channel — the water depth may not exceed 0.90 m — and the certified freeboard must be stated to the nearest centimetre. The city owns those numbers, not the programmer.

An engineer must certify, by computation, the water depth at which this channel carries exactly the design flow — the NORMAL DEPTH — and decide whether the freeboard requirement is met. The flow equation has no closed-form solution for the depth: no algebra hands you the answer. This is the first homework in which a root must be HUNTED — bracketed, bisected, and certified with a bound.

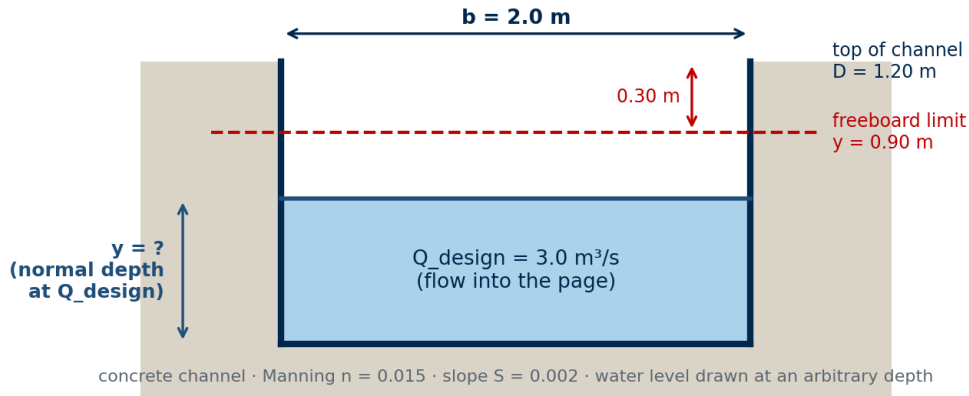


Figure: Rectangular channel cross-section for this homework — width b , water depth y (the unknown normal depth), the 0.90-m freeboard limit, and the 1.20-m top of channel. The model box and parameter table below remain the authoritative values for computation; the water level in the figure is drawn at an arbitrary depth.

THE UNIFORM-FLOW MODEL — provided; no hydraulics course required

For steady, uniform flow in an open channel, **Manning's equation** relates the flow rate to the depth. For a rectangular channel of width b carrying water at depth y , the wetted cross-section is $A = b \cdot y$, the wetted perimeter (bottom + two walls) is $P = b + 2 \cdot y$, and the hydraulic radius is $R = A / P$. Then:

Deeper water → larger A and larger R → more flow: $Q(y)$ rises steadily with y , from $Q(0) = 0$ upward without limit. The NORMAL DEPTH y_n is the depth at which $Q(y_n) = Q_{\text{design}}$. Written a new way:

$$Q(y) = (1/n) \cdot A \cdot R^{2/3} \cdot \sqrt{S} \quad f(y) = Q(y) - Q_{\text{design}} = 0$$

y = water depth (m) · b = channel width (m) · n = Manning roughness (SI) · S = slope (m/m) · Q = flow rate (m^3/s)

The leading factor 1 belongs to SI units (metres, m^3/s). Manning's equation is also used in US customary units (feet, ft^3/s), where the factor is 1.49 — same equation, different unit system. The values below are instructional design parameters for this assignment. Everything you compute is a MODEL PREDICTION for an idealized channel — not yet field evidence about a real one.

Quantity	Symbol	Value	Units
Channel width	b	2.0	m
Total channel depth	D	1.20	m
Longitudinal slope	S	0.002	m/m
Manning roughness (finished concrete, SI)	n	0.015	—
Design flow	Q_{design}	3.0	m^3/s

Quantity	Symbol	Value	Units
Required freeboard (city standard)	FB_min	0.30	m
Certified freeboard resolution (city standard)	—	nearest 0.01	m

Time guide: Part A ≈ 10 min · Part B ≈ 10 min · Part C ≈ 8 min · Part D ≈ 10 min · Part E ≈ 15 min · Part F ≈ 8 min · Part G ≈ 12 min · Part H ≈ 7 min. If you are far past 90 minutes, submit what you have with honest notes — do not expand the assignment.

PART A · Write It & Commit Predictions (≈ 10 min)

Study the figure together with the context, model, and parameter boxes above — observe the physical system before computing anything about it. Then write the question in the chapter's shape and commit — in writing, before any computation. Wrong is fine; uncommitted is not. Do not edit these later.

A1 — The Problem, in Your Own Words. In one or two sentences: what exactly are you asking the computer to find, and what engineering decision depends on the result?

ANSWER A1:

A2 — The Sign Has Meaning. $f(y) = Q(y) - Q_{\text{design}}$. Say what each sign means physically for THIS channel — in terms of the water and the design storm, not algebra.

$f(y) < 0$ means:

$f(y) > 0$ means:

$f(y) = 0$ means:

A3 — Prediction P1 · The Decision. Before computing: will the channel PASS (normal depth ≤ 0.90 m) or FAIL the freeboard requirement? Commit, then give one sentence of reasoning.

P1: PASSES ☐ **FAILS** ☐ **too close to call** ☐

REASONING P1:

A4 — Prediction P2 · The Price of a Guarantee. Bisection will start from the physical cage $[0, 1.20$ m] and must certify the depth to ± 0.005 m (half a centimetre — the city's resolution). How many halvings will that take? Commit a number: 10? 50? 200?

P2: _____ halvings

A5 — Prediction P3 · A Tighter Cage. If a coarse sweep first narrowed the cage to $[0.80, 1.00$ m] — six times narrower — would bisection need fewer halvings? How many fewer? Commit now — Part D will hold you to it.

P3: _____ fewer halvings

PART B · Own the Model — Two Evaluations by Hand (≈ 10 min)

The program in Part E is provided nearly complete. But a provided program is not a black box, and a bracket is not certified by hoping: before anything runs, evaluate the model yourself. These two hand values are your known cases for the rest of the homework.

B1 — Quantities and Roles. Complete the Role column: is each quantity an input parameter (set once, never changes), the unknown the search moves (the state), an intermediate computed from the state, or an output (reported at the end)?

Quantity	Symbol	Units	Role — your answer
Channel width	b	m	
Manning roughness	n	—	
Slope	S	m/m	
Water depth	y	m	
Hydraulic radius	$R = b \cdot y / (b + 2y)$	m	
$f(y) = Q(y) - Q_{\text{design}}$	f	m ³ /s	

B2 — f at the Freeboard Limit, by Hand. Evaluate $Q(0.90)$ step by step using the model box — A, P, R, $R^{(2/3)}$, $\sqrt{S}$, Q — then $f(0.90)$. Show the arithmetic. Then answer: what does the SIGN of $f(0.90)$ already tell you about the decision, before any search?

ANSWER B2: A = _____ m² P = _____ m R = _____ m $R^{(2/3)}$ = _____ $\sqrt{S}$ = _____ Q(0.90) = _____ m³/s f(0.90) = _____ m³/s (sign _____)

What the sign at the limit depth says about the decision:

B3 — A Second Evaluation. Now evaluate $f(0.80)$ the same way. Do the two hand values have opposite signs? Write the sentence that makes [0.80, 0.90] a certified bracket — and name the theorem's hinge that the sentence relies on.

ANSWER B3: Q(0.80) = _____ m³/s f(0.80) = _____ m³/s (sign _____)
Opposite signs? YES ☐ NO ☐

Certification sentence:

PART C · Trap It — the Physical Cage and the Contract (≈ 8 min)

Part B found a tight bracket by hand. The program will instead start from a cage you can defend without computing — Chapter 4's first source of brackets: physics. Certify it, then read the contract's fine print for THIS model.

C1 — The Cage from Physics. Explain, without a calculator, why $f(0)$ must be negative and why $f(1.20)$ must be positive (use your B2 value and the direction Q moves with depth). Then evaluate $f(1.20)$ in Python or by hand and record both endpoint values.

$f(0) = \underline{\hspace{2cm}} \text{ m}^3/\text{s}$ (why, physically: _____)

$f(1.20) = \underline{\hspace{2cm}} \text{ m}^3/\text{s}$ (why, physically: _____)

C2 — Continuity and the Right Root. The guarantee needs a continuous f and it promises only “at least one root.” (a) Why is $f(y)$ continuous for $y > 0$ — which pieces of the model could break continuity, and do they? (b) Why is there exactly ONE crossing — what does deeper water always do to Q ? (c) Which crossing does the decision need?

ANSWER C2:

C3 — The Contract in One Line. Write the single line of Python that certifies the sign change before any loop runs, and say what happens to the program if the cage is wrong.

ANSWER C3:

PART D · Hunt It by Hand — Three Halvings and the Schedule (≈ 10 min)

No AI, no Python: start from the hand-certified bracket $[0.80, 1.00]$ ($f(1.00) = +0.756 \text{ m}^3/\text{s}$ — you may take this value as given) and run three halvings. The arithmetic is Part B's; the only judgment per row is WHICH HALF SURVIVES.

n	midpoint m (m)	f(m) (m ³ /s)	bracket kept · half-width (m)
1	0.900	(from B2)	
2			
3			

D1 — Report After Three Halvings. State the depth as midpoint $\pm$ half-width. Is the freeboard decision already certain from this bracket alone? Why or why not?

D1: $y_n =$ _____ $\pm$ _____ m Decision certain? YES ☐ NO ☐
because: _____

D2 — The Schedule. Bisection's bound after n halvings is $W_0 / 2^{n+1}$. The city's ± 0.005 m resolution sets the tolerance. Compute the number of halvings needed from the physical cage $[0, 1.20]$ and from the tighter cage $[0.80, 1.00]$. Show the inequality you solved.

D2: from $[0, 1.20]$: $n =$ _____ from $[0.80, 1.00]$: $n =$ _____
inequality: _____

D3 — Reconcile P2 and P3. Compare with your committed predictions. A six-times wider cage cost how many extra halvings? Explain in one sentence why bisection “forgives a generous bracket.”

ANSWER D3:

PART E · Hunt It in Python — the Certified Run (≈ 15 min)

The program below is complete except for the two marked lines. Copy it into your Python environment (IDLE, Spyder, Jupyter, VS Code — any is fine), complete ① and ②, and run it from the physical cage with the city's tolerance. Your Part B hand values tell you exactly what the first evaluations must return.

```
# channel_depth.py – Chapter 4: normal depth by bisection
import math

b, n, S = 2.0, 0.015, 0.002          # width (m), Manning n (SI), slope
Q_design = 3.0                       # design flow (m³/s)

def manning_Q(y):                    # flow carried at depth y (SI)
    A = b * y
    P = b + 2 * y
    R = A / P
    return (1.0 / n) * A * R ** (2.0 / 3.0) * math.sqrt(S)

def f(y):                             # f(y) = 0 at the normal depth
    return manning_Q(y) - Q_design

a, b_ = 0.0, 1.20                    # physical cage (m)
fa, fb = f(a), f(b_)                 # 2 endpoint evaluations
assert fa * fb < 0                    # certify the sign change
tol = 0.005                           # from the city standard (m)
halvings = 0

while _____ > tol:               # COMPLETE ①: stop on the BOUND, not a count
    m = (a + b_) / 2
    fm = f(m)
    halvings = halvings + 1
    print(halvings, a, b_, m, fm, (b_ - a) / 2)
    if _____ <= 0:               # COMPLETE ②: sign change in [a, m]?
        b_, fb = m, fm
    else:
        a, fa = m, fm

print(halvings, "halvings:", (a + b) / 2, "+/-", (b - a) / 2, "m")
```

E1 — Evidence. Paste ONE screenshot showing your completed program together with its printed log and final line, with no unresolved error. This is the only screenshot this homework requires.

PASTE PYTHON CODE / OUTPUT SCREENSHOT HERE

(click here and paste your screenshot — Ctrl+V / Cmd+V)

E2 — Result & Known-Case Checks. Report the run and test it against your own evidence: the printed row where $m = 0.900$ must reproduce your B2 value; the halving count must match your D2 schedule.

Halvings run: _____ (D2 predicted _____ → agree? YES ☐ NO ☐)

Final: $y_n = \text{_____} \pm \text{_____} m$

Printed $f(0.900)$ agrees with B2? YES ☐ NO ☐ $Q(y_n)$ from the model = _____ $m^3/s \approx Q_{\text{design}}$? YES ☐ NO ☐

E3 — Read the Log. From your printed rows: does the half-width halve every row? Does the midpoint move toward the root monotonically? Which row first certified the decision (bracket entirely below 0.90 m)?

ANSWER E3:

E4 — The Tighter Cage. Change only the cage to $a, b_ = 0.80, 1.00$ and rerun. Record the halvings and the result. Do the two runs agree within their bounds? What did the tighter cage buy, and what did it cost you to obtain it?

E4: halvings = _____ $y_n = \text{_____} \pm \text{_____} m$ agree with the [0, 1.20] run within the bounds? YES ☐ NO ☐

E5 — Justified Digits. Python printed many digits. Using the final half-width as your evidence, write the depth and the freeboard ($D - y_n$) with the digits the bound justifies — and no more.

E5: $y_n = \text{_____} m$ freeboard = _____ m (bound: $\pm \text{_____} m$)

Optional — Excel. Build the first three rows of the same algorithm in Excel: $a, b, m = (a+b)/2, f(a), f(m)$, the sign test $f(a) \cdot f(m) \leq 0$, and IF-formulas that produce the new a and new b ; copy the row down. If you do, paste a small screenshot after E1. Same algorithm, new platform — the bisection logic is the method.

PART F · Trust It — Tolerance, Margin & the Decision (≈ 8 min)

F1 — Who Owns ± 0.005 m? Name the requirement that fixed the tolerance, and explain why “tol = 0.005” is not a Python setting you were free to change. What would a tolerance of 0.0001 m have cost, and what would it have bought the city?

ANSWER F1:

F2 — The Freeboard Verdict. Compute the certified freeboard as a range: $D - (y_n \pm \text{bound})$. Compare the worst case with the 0.30 m requirement and record the verdict.

F2: freeboard = _____ to _____ m worst case $\geq$ 0.30 m? YES ☐
NO ☐ PASSES ☐ FAILS ☐ margin over 0.30 m: _____ m

F3 — Bound versus Margin. Compare the bisection bound (± 0.005 m) with the margin between the certified freeboard and 0.30 m. Which is larger, and by roughly what factor? Now imagine the design flow were larger so that the margin shrank to 0.004 m: could the same run still certify a PASS? What would you have to do?

ANSWER F3:

PART G · AI Analyst & BREAK THE AI — Audit the Guarantee (≈ 12 min)

In Chapter 4 the analyst may propose the method — and precisely because it may, every recommendation must arrive with its contract attached. The rule has not changed: AI assists. Human verifies. Engineer decides.

G1 — Hand AI Your Certificate. Give any capable AI assistant your $f(y)$, your certified cage with both signs, and the ± 0.005 m tolerance, with a prompt close to this:

“Here is my root-finding problem: $f(y) = Q(y) - 3.0$ with Manning’s equation in SI units for a 2.0-m-wide rectangular channel, $n = 0.015$, $S = 0.002$ [paste your f]. My bracket $[0, 1.20]$ has $f(0) = -3.0$ and $f(1.20) = +1.78$. The required tolerance is ± 0.005 m. Before any code: check my sign-change evidence, tell me exactly what bisection’s guarantee will and will not cover here, and work out how many halvings my requirement implies. Do not run anything yet.”

Record ONE meaningful claim from the AI analysis, verbatim (one or two lines — not the whole conversation).

ANSWER G1:

G2 — Your Verdict on the Analysis. Judge that claim as the engineer. Do not manufacture disagreement: ACCEPT is a fully legitimate verdict when you have verified it — say how.

ACCEPT ☐ MODIFY ☐ REJECT ☐

WHY (1–2 sentences):

BREAK THE AI — The Certified Memo. A different AI assistant was asked to certify the channel and produced the memo and code below. Its bisection is flawless: the sign change is certified in code, the loop stops on the bound, the count matches the schedule. Every number in it is real — run the code yourself if you doubt it. And it is wrong. Something the theorem never covered is being sold.

AI CERTIFICATION MEMO — verbatim, with its code

“Using bisection with a verified sign change on $[0, 1.20\text{ m}]$ and a tolerance of 0.005 m , the normal depth converges in 7 iterations to $y = 0.633\text{ m} \pm 0.005\text{ m}$ — guaranteed by the intermediate value theorem. Freeboard $= 1.20 - 0.633 = 0.567\text{ m}$, nearly double the 0.30 m required. Conclusion: the channel passes the city standard with a large safety margin.” Its flow function: `def Q(y): A = b*y; P = b + 2*y; R = A/P; return 1.49 / n * A * R**(2/3) * S**0.5`

G3 — Attack the Contract, Not the Syntax. Using your own Part B, E and F evidence, give at least THREE specific reasons the memo’s engineering claim does not stand. Aim at what the guarantee covers and what it does not: is the bisection wrong, or is something upstream of it wrong? What does the memo’s own $f(0.90)$ equal, and how does that compare with your hand value? Which direction does its error push the decision — safe or unsafe?

ANSWER G3:

G4 — Verdict on the Memo. One check, one sentence.

ACCEPT ☐ MODIFY ☐ REJECT ☐

WHY (one sentence):

PART H · A Guarantee Is a Contract — Certify (≈ 7 min)

H1 — What the Guarantee Does NOT Establish. Your run satisfied the contract — certified sign change, continuous f , one crossing — and the bound is mathematically guaranteed. State TWO things this guarantee still does not establish about the real channel. (Think: the as-built width, the roughness of aged or silted concrete, the uniform-flow idealization, the units.)

ANSWER H1:

H2 — The Engineering Certification. Complete the statement. Every blank matters — report the depth to the digits your bound justifies, not every digit Python printed.

“Under the stated uniform-flow model and parameters ($b = 2.0 \text{ m}$, $n = 0.015 \text{ SI}$, $S = 0.002$, $Q_{\text{design}} = 3.0 \text{ m}^3/\text{s}$), I wrote $f(y) = Q(y) - Q_{\text{design}}$ and certified the bracket $[0, 1.20 \text{ m}]$ with $f(0) = \underline{\hspace{2cm}}$ and $f(1.20) = \underline{\hspace{2cm}} \text{ m}^3/\text{s}$. f is continuous for $y > 0$ and has exactly one crossing. Bisection with the city’s $\pm 0.005 \text{ m}$ tolerance needed halvings (predicted) and gives the normal depth $y_n = \underline{\hspace{2cm}} \pm \underline{\hspace{2cm}} \text{ m}$ (midpoint $\pm$ final half-width). Known case: $Q(y_n) = \underline{\hspace{2cm}} \text{ m}^3/\text{s}$.

The certified freeboard is to m against the 0.30 m requirement. The channel **PASSES** ☐ / **FAILS** ☐ the city standard, with a margin of m — times the numerical bound.

I recommend:

This is a model-based computational conclusion under the stated assumptions — the bisection bound covers the computation only; it does not validate the model, the roughness, the as-built geometry, or the units, and field verification is required before final acceptance.”

Evidence behind this certification (one line each): ① hand-evaluated known cases ② certified sign change + continuity ③ predicted $n = \text{run } n$; root $\pm$ half-width ④ bound vs margin; the memo’s failure named.

EVIDENCE ①:

EVIDENCE ②:

EVIDENCE ③:

EVIDENCE ④:

THE THEOREM WAS NEVER THE PROBLEM.

Two bisections, both flawless, both “guaranteed” — one certified the channel correctly, the other certified it with a factor that belongs to another unit system. The mathematics kept its promise both times; the promise covered only the computation. Carry the principle into every root you will ever sign: a guarantee holds only where its conditions hold — and the conditions that matter most (the model, the parameters, the units) are the ones no theorem checks for you.

SUBMIT · Evidence Checklist & Human Review (≈ 3 min)

Submit ONE completed Word document — this file — containing all of the following:

- ☐ A1 problem statement · A2 sign meanings · committed predictions P1, P2, P3 — unedited
- ☐ B1 roles table · B2 hand evaluation at the limit depth with arithmetic and the decision sentence · B3 second evaluation + certification sentence
- ☐ C1 physical cage with both endpoint values and reasons · C2 continuity / single-crossing / right-root answers · C3 the assert line

- ☐ Part D three-halving table · D1 report and decision-certainty · D2 schedule with the inequality · D3 reconciliation
- ☐ E1 code + log screenshot · E2 halvings, result and known-case checks · E3 log reading · E4 tighter-cage rerun · E5 justified digits
- ☐ F1 tolerance ownership · F2 freeboard range and verdict · F3 bound-versus-margin reasoning
- ☐ G1 recorded AI claim · G2 verdict · G3 three-point attack on the memo · G4 memo verdict
- ☐ H1 two not-established items · H2 completed certification with all four evidence lines
- ☐ Human review statement signed and dated (below)

SUBMIT ONE COMPLETED HOMEWORK DOCUMENT

Keep all required evidence in this document rather than as separate Python or image files. A separate .py file is welcome as an optional attachment, but the document must stand on its own. If this homework is being used as part of a course, follow your instructor's submission and file-naming requirements.

HUMAN REVIEW

"I reviewed the predictions, hand evaluations, bracket certification, code, screenshot, log, and written responses in this submission and confirm that they represent the work I performed, that I can explain every line of the program, and that I can defend every condition behind my certification."

Human reviewed and approved: _____

Student name: _____ Date: _____

Suggested Assessment Rubric — 20 points

Used by the instructor and, where applicable, by an AI grading agent. All required evidence must appear inside this document — the grader will not open separate Python or image files. A submission whose program runs, but whose certification does not check the contract behind the guarantee, cannot earn a high score.

Criterion	What full credit looks like	Pts
Write it & committed predictions (A1–A5)	A1 names both the computed quantity (the normal depth) and the decision (freeboard). A2 gives all three signs a physical meaning. P1–P3 committed with reasoning. Prediction correctness is NOT graded — commitment and reasoning are.	2
Model ownership & hand evaluations (B1–B3)	Roles column correct. B2 arithmetic shown and consistent with the key; the sign at the limit depth interpreted correctly for the decision. B3 second value correct; certification sentence names the sign change AND continuity as the hinge.	3

Criterion	What full credit looks like	Pts
Trap it — cage and contract (C1–C3)	Both endpoint values with physical reasons; continuity and the single crossing argued from the model (A and R grow with y); the right root identified; the assert written with its consequence explained.	2
Hunt it by hand — halvings and schedule (D1–D3)	Three rows with the surviving bracket each time; report as midpoint $\pm$ half-width; decision-certainty judged from the bracket; both halving counts from $W_0/2^{n+1} \leq \text{tol}$ with the inequality shown; P2/P3 reconciled with the “generous bracket” insight.	3
Python run — certified, inspected, verified (E1–E5)	Both marked lines completed correctly (stop on the half-width; sign test $f(a) \cdot f(m)$); screenshot shows the log and final line; halving count matches the schedule; B2 known case reproduced; log reading correct (bound halves, midpoint not monotone, first decisive row); tighter-cage rerun agrees within bounds; digits justified by the bound.	4
Trust it — tolerance, margin, decision (F1–F3)	Tolerance ownership named; freeboard stated as a certified range; verdict from the worst case; bound-versus-margin compared quantitatively, with the shrunk-margin case reasoned correctly (more halvings, or a decision that cannot be certified at this tolerance).	2
AI analyst & BREAK THE AI (G1–G4)	One verbatim AI claim judged with independent justification. The memo attacked with at least three specific, evidence-based reasons — the unit-system factor, the hand-value mismatch at 0.90 m, the unsafe direction of the error, and the scope of the guarantee. Verdicts graded on justification, never on which box is checked.	3
A guarantee is a contract — certification (H1–H2)	Two legitimate not-established items. Certification complete and internally consistent: endpoint values, halvings, root $\pm$ bound with justified digits, known case, freeboard range, verdict, margin-to-bound ratio, a real recommendation, the model-based caveat intact, all four evidence lines filled.	1
	TOTAL	20