FIT1058 Foundations of Computing

MONASH UNIVERSITY · SEMESTER 2, 2026 · UNDERGRADUATE

Full guide + ask Sia about FIT1058

SIDE 1/2 DEFINITIONS → FORMULAS → DECISION RULES → CHECKS **EXAM REVISION SHEET** Final assessment 40% · component threshold 45%

0 · Exam Map COURSE · BRIEF

Q 10	A1 10	MST 15	A2 25	Final 40%

Quizzes 10% · Assignment 1 10% · Mid-semester test 15% · Assignment 2 25% · Final assessment 40%

Permitted materials: not confirmed for this offering in the course materials we hold — check the exam instructions on the LMS.

THRESHOLD	WHERE	CONSEQUENCE
45%	Final assessment	Failed hurdle: NH; maximum unit mark 45
45%	In-semester total	
50%	Overall unit mark	Required to pass

The final is an **examination-period assessment**. Operational details remain controlled by the current institutional instructions.

1 · Universal Answer Path COURSE STRATEGY

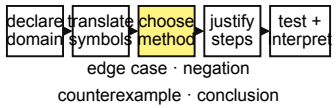


Figure 1. The recurring source-grounded solution path.

PROMPT SIGNAL	FIRST MOVE	LAST CHECK
Collection / overlap	Name sets and universe	Disjointness, finiteness
Mapping / relation	State domain and codomain	Definition for arbitrary elements
"For every" claim	Choose proof structure	Boundary or counterexample
Counting / chance	Declare outcomes	Order, repetition, conditioning
Network	Declare graph convention	Directedness, loops, weights

WHAT A COMPLETE LINE EARNS

Definition: identify the object and its declared universe or domain. **Translation:** separate quantities supplied by the prompt from quantities derived. **Operation:** write the rule before substituting. **Justification:** preserve direction, indices and conditions line by line. **Stress-test:** try the boundary, negation or a counterexample. **Interpretation:** answer in the language of the prompt, not merely with a symbol or number.

The final rewards the same transferable discipline across topics: declare → translate → choose → justify → test → conclude. If a later line fails, repair the first unsupported move rather than copying a complete solution.

EXAM CUE *Marks are lost when natural language, symbols and proof drift apart. Definitions do the checking.*

2 · Sets & Inclusion–Exclusion C1

Set: well-defined collection of distinct objects. *Subset*: every element of one set belongs to another. *Power set*: all subsets, including ∅ and the set itself.

TWO-SET INCLUSION–EXCLUSION

```
|A ∪ B| = |A| + |B| - |A ∩ B|
Subtract the intersection once: it
was counted in both cardinalities.
```

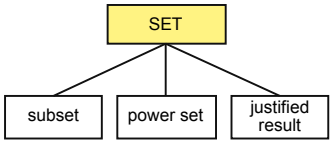


Figure 2. Source figure redrawn for one-column recall.

TASK	WRITE	CHECK
Translate	Universe; membership; named sets	Every element is eligible
Operate	∪, ∩, subset relation	Overlap counted once
Refute	One valid counterexample	State corrected condition

Control the universe, finiteness and disjointness assumptions. A numerical answer without those conditions can invoke the wrong counting identity.

3 · Functions & Relations C2

Function: each domain element is assigned exactly one codomain element. *Injective*: distinct domain inputs map to distinct outputs. The declared domain and codomain control injectivity and surjectivity.

COMPOSITION

```
(g ∘ f)(x) = g(f(x))
Apply f first, then g; the
intermediate codomain/domain must be
compatible.
```

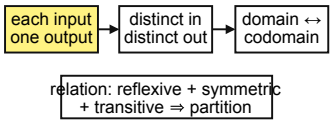


Figure 3. Function and equivalence-relation definition tests.

An *equivalence relation* is reflexive, symmetric and transitive; therefore it partitions a set. To earn the conclusion, check all three laws for arbitrary eligible elements—not only examples.

PROPERTY	DEFINITION-LED TEST
Function	Every domain input receives exactly one codomain output.
Injective	Distinct domain inputs must have distinct outputs.
Reflexive	Each eligible element is related to itself.
Symmetric	Reverse every related pair.
Transitive	Chain two related pairs through the middle element.

4 · Pick the Proof Method C3

METHOD	USE WHEN	NON-NEGOTIABLE
Direct	Definitions turn assumptions into conclusion	Name the final definition
Contrapositive	¬Q gives a cleaner route to ¬P	It is equivalent to P → Q
Contradiction	Negating the claim exposes an impossibility	Identify the contradiction
Induction	Universal statement indexed by successive integers	Base + hypothesis + step

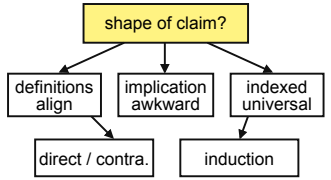


Figure 4. Let the claim's structure choose the proof.

INDUCTION CHECKPOINT

```
Base: verify P(first index).
Hypothesis: assume P(k) for an
arbitrary eligible k.
Step: use that assumption to prove
P(k+1).
Conclusion: the stated range
follows.
```

P(k) ⇒ P(k+1) is not enough alone. Checking several cases, asserting the hypothesis, or restating the target does not prove a universal statement.

INDUCTION LINE-BY-LINE

Write the first eligible index and verify it. State the induction hypothesis exactly, including its range. In the step, begin with the P(k+1) expression and show where P(k) is used; do not assume the line being proved. Finish by stating that the base case and implication cover the declared range.

FAILURE	WHY IT FAILS
Checks k=1,2,3 only	Finite evidence is not a universal proof.
Assumes P(k+1)	The target has been smuggled into the premise.
Algebra ends without definition	The conclusion is not connected back to the claim.

5 · Proof Quality Audit C3 FAQ

AUDIT	QUESTION TO ASK
Domain	Are all variables and eligible values declared?
Direction	Did I prove the requested implication, not its converse?
Generality	Are the elements arbitrary rather than hand-picked?
Dependency	Does each line follow from a definition, assumption or established result?

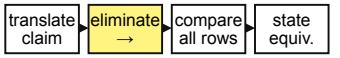
6 · Propositional Logic C4

Proposition: a declarative statement assigned a truth value. *Logical equivalence*: expressions match under every valuation. *Normal form*: a standardised form such as conjunctive or disjunctive normal form.

IMPLICATION ELIMINATION

```
p → q ≡ ¬p ∨ q
The implication is false only when p
is true and q is false.
```

P	Q	P → Q	¬P ∨ Q
T	T	T	T
T	F	F	F
F	T	T	T
F	F	T	T



implication ≠ converse ≠ biconditional

Figure 5. Equivalence demands every truth row.

Do not swap implication, converse and biconditional. Their truth conditions differ even if the same propositions appear.

7 · Predicates & Quantifiers C5

Predicate: an open statement whose truth depends on variables from a declared domain. ∀ claims every object; ∃ claims at least one object.

QUANTIFIER NEGATION

```
¬∀x P(x) ≡ ∃x ¬P(x)
A universal claim fails exactly when
a counterexample exists in the
declared domain.
```

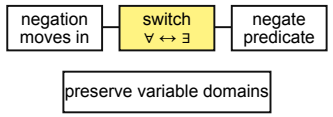


Figure 6. Negate scope, quantifier and predicate together.

Moving or swapping quantifiers can change meaning even when the same predicates appear. Write the domain beside the first translation.

READ	WRITE / TEST
"Every object has…"	Start with ∀ over the declared object domain.
"There exists…"	Start with ∃; one valid witness establishes it.
"Not every…"	Use ∃ and negate the predicate; seek a counterexample.
Nested claim	Keep the original order until negation switches each quantifier.

8 · Sequences & Series C6

Sequence: a function from an ordered integer index set to values. A *recurrence relation* defines terms from earlier terms plus initial conditions. A *geometric series* has a constant ratio.

FINITE GEOMETRIC SUM

```
Σ from i=0 to n-1 of arⁱ =
a(1-rⁿ)/(1-r)
```

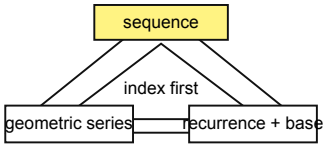


Figure 7. Index and base conditions define the object.

BEFORE SUMMING	STATE EXPLICITLY
Index range	First and last exponent; number of terms
Geometric structure	First term a; common ratio r
Recurrence	Initial value(s) and valid index
Audit	Generate early terms and compare

Count terms separately from reading the largest exponent. Off-by-one indexing or a missing initial value defines a different sequence.

Fast verification: expand the first two or three terms from the stated index and compare them with the prompt. For a recurrence, confirm every required initial condition before generating a later term. For a geometric sum, identify a, r and n independently; only then substitute.

9 · Combinatorial Counting C7

The *multiplication principle* multiplies options at sequential independent stages. A *permutation* is ordered; a *combination* is an unordered selection of fixed size.

COMBINATION

```
C(n,k) = n!/[k!(n-k)!]
```

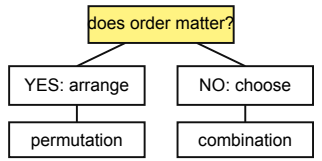


Figure 8. Name sameness before selecting a rule.

Then ask: repetition? exclusions? overlap? Never divide or multiply by factorials without identifying the symmetry; that is the source's stated over-/under-counting trap.

COUNTING AUDIT	QUESTION
Outcome	What exactly counts as one distinct result?

10 · Number Theory C8

Divisibility: a divides b when b = ak for some integer k. The *greatest common divisor* is the largest positive integer dividing both. *Congruence*: equal integer remainders modulo a positive modulus.

EUCLIDEAN RECURRENCE

```
gcd(a,b) = gcd(b, a mod b)
Repeat until remainder 0; take the
last non-zero remainder.
```

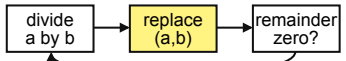


Figure 9. Each remainder preserves common divisors.

Modular cancellation is conditional. Division and cancellation rules over integers do not always transfer unchanged modulo composite numbers.

11 · Discrete Probability C9

Sample space: all outcomes in the model. *Conditional probability*: restrict attention to cases where another event occurs. *Independence*: learning one event does not change the probability of the other.

BAYES' RULE

```
P(A | B) = P(B | A)P(A) / P(B)
```

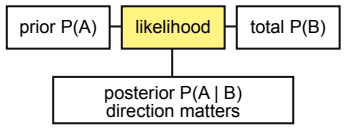
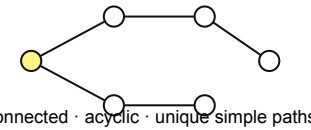


Figure 10. Prior × likelihood, then normalise.

DO NOT CONFUSE	DEFINITION CHECK
Disjoint	Cannot occur together
Independent	One does not change the other's probability
P(A|B) vs P(B|A)	The restricted population reverses

12 · Graphs & Trees C10

A *graph* is vertices plus edges under a stated convention. A *path* obeys its movement rules. A *tree* is connected and acyclic; equivalently, every vertex pair has a unique simple path.



connected · acyclic · unique simple paths

Figure 11. Tree tests need graph conventions first.

FIT1058 Worked Drills, Traps & Rapid Recall

FOUNDATIONS OF COMPUTING · MONASH UNIVERSITY · SEMESTER 2, 2026

Full guide + ask Sia about FIT1058

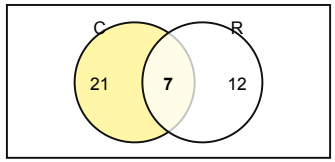
SIDE 2/2 FIVE WORKED ROUTES · IF YOU SEE X → DO Y · CLASSIC TRAPS · EXAM-MORNING PLAN

SELF-TEST SIDE

Answers stay hidden until your final line

W1 · Inclusion–Exclusion C1 WORKED

Prompt. 28 coding students, 19 robotics students, 7 in both. How many are in at least one?



28 + 19 − 7 = 40

Figure 12. Overlap is counted once, not twice.

1. Name sets C and R.
2. Use $|C \cup R| = |C| + |R| - |C \cap R|$.
3. Compute 28 + 19 − 7 = 40.
4. Audit: 21 only coding + 7 both + 12 only robotics = 40.

ANSWER

`40 students are in at least one club.`

TRAP *28 + 19 = 47 double-counts the seven students in the intersection.*

Rapid Set / Relation Checks C1–C2

IF YOU SEE...	DO THIS
"always" + set identity	Try overlap, ∅ and equal sets; one counterexample refutes.
Mapping diagram	First test: each input has exactly one output.
"Injective?"	Test whether distinct inputs can share an output.
"Equivalence?"	Prove reflexive, symmetric and transitive separately.

a ~ a reflexive | a ~ b ⇒ b ~ a symmetric | a~b,b~c ⇒ a~c

all three ⇒ equivalence classes

Figure 13. Three obligations, one partition conclusion.

RELATION AUDIT: DIVISIBILITY BY 3

On integers define a~b when 3 divides a−b. **Reflexive:** a−a=0 is divisible by 3. **Symmetric:** if 3 divides a−b, it also divides b−a. **Transitive:** if 3 divides a−b and b−c, it divides their sum a−c. Therefore the relation is reflexive, symmetric and transitive and partitions the integers into three residue classes.

Use arbitrary integers throughout. Checking 1~4 or 2~5 illustrates the rule but does not prove a relation law.

SET ERROR	DIAGNOSTIC REPAIR
Union answer exceeds |A|+|B|	Recheck signs and the declared universe.
Overlap not visible	Sketch two regions and label intersection first.

W2 · Direct Proof C3 WORKED

Prompt. Prove: if n is odd, then n^2 is odd.

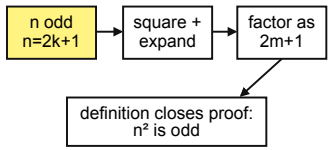


Figure 14. Algebra earns meaning only at the definition.

1. Let n = 2k + 1 for integer k.
2. $n^2 = (2k+1)^2 = 4k^2 + 4k + 1$.
3. $n^2 = 2(2k^2 + 2k) + 1$.
4. Since $2k^2 + 2k$ is an integer, n^2 has the form 2m + 1.

ANSWER

`Therefore n² is odd by the definition of odd.`

Logic & Quantifier Drills C4–C5 WORKED

PROMPT	ANSWER PATH
Rewrite p implies q using only ¬ and ∨.	It fails only at T,F → compare four rows → p → q ≡ ¬p ∨ q.
Negate: every user remembers some password.	Switch ∀ user to ∃ user; switch ∃ password to ∀ password; negate remembers.

∀ user ∃ password : remembers
original scope and domains

∃ user ∀ password : does not remember
negated statement

Figure 15. Switch each quantifier; negate the predicate.

PROOF AND LOGIC FAILURE MODES

FAULTY MOVE	REPAIR
Proves several examples	Return to arbitrary eligible elements or give a counterexample if refuting.
Uses converse of P→Q	Write P, Q and the requested direction before choosing a method.
Declares "equivalent" after one row	Compare all four valuations.
Negates predicate only	Move negation through scope, switching every crossed quantifier.
Forgets base case	State and verify the first eligible index.

Self-test: cover the answer column. For each faulty move, explain whether it is a translation, calculation or interpretation error, then write the first corrected line. Repair the earliest failure.

TRAP *Negating words but leaving ∀/∃ unchanged changes the intended statement incorrectly.*

W3 · Geometric Sum C6 WORKED

Prompt. Find $1 + 2 + 4 + \ldots + 2^9$.

2^0	2^1	2^2	2^3	2^4	2^5	2^6	2^7	2^8	2^9

10 terms · a=1 · r=2

$(1-2^{10})/(1-2)=1023$

Figure 16. Largest exponent 9 still means ten terms.

1. Recognise a = 1, r = 2.
2. Indices 0 through 9 give n = 10 terms.
3. Apply $a(1-r^n)/(1-r)$.
4. $(1-2^{10})/(1-2) = 2^{10}-1 = 1023$.

ANSWER

`1023`

Counting Judgement Cards C7

IF YOU SEE X	→ DO Y
Stages with independent choices	Multiply the number of options at each stage.
Ordered arrangement without replacement	Use a permutation structure.
Unordered k-person group	Use C(n,k).
Restrictions / forbidden cases	Define valid outcomes before counting.
Possible symmetry	Say what makes two outcomes identical before using factorials.

8 students choose 3 — order does not matter — 8!/(3!5!) = 56

56 distinct teams

Figure 17. Guide check: C(8,3)=56 teams.

TRAP *8×7×6 counts the same three-person team in multiple orders; the combination denominator removes that symmetry.*

CHANGED-SCENARIO CHECKS

For the series, change only the last exponent and recount the indices before touching the formula. For the team prompt, ask whether roles are introduced: once outcomes are distinguished by role, the "order does not matter" decision must be reconsidered. For either problem, state the object being counted or summed before choosing notation.

CHECK	SERIES	COUNTING
Object	Ordered indexed terms	Distinct outcomes
Structure	Constant ratio	Stages / arrangement / selection
Boundary	First and last index	Small n or k
Primary trap	Off-by-one	Unstated symmetry

W4 · Euclid's Algorithm C8 WORKED

Prompt. Find gcd(252,105).

252 = 2×105 + 42

105 = 2×42 + 21

42 = 2×21 + 0

last non-zero remainder = 21

Figure 18. Remainders descend to zero.

1. 252 = 2×105 + 42.
2. 105 = 2×42 + 21.
3. 42 = 2×21 + 0.
4. The last non-zero remainder is 21.

ANSWER

`gcd(252,105) = 21`

AUDIT *21 divides 252 and 105. Every remainder step preserves the common-divisor set.*

Modular & Relation Traps C2 · C8

TRAP	REPAIR
Cancel a common factor modulo a composite number automatically.	Return to the definition of congruence; ordinary integer cancellation rules do not always transfer.
Check a~b on selected examples only.	Use arbitrary eligible integers for each relation law.
Forget the modulus or change it mid-solution.	State the positive modulus with each congruence chain.
Stop Euclid before remainder 0.	Continue; report the last non-zero remainder.

…−3,0,3… class 0 | …−2,1,4… class 1 | …−1,2,5… class 2

a ~ b when 3 divides a−b

Figure 19. Modulo 3 partitions integers into three classes.

EUCLID SELF-CHECKS

At every line, the dividend equals quotient×divisor + remainder. The remainder must be smaller than the divisor. When the remainder reaches zero, the current divisor—not zero—is the gcd. Verify the reported gcd divides both original integers.

Congruence discipline: retain the positive modulus beside the statement, translate congruence into equality of remainders, and resist importing ordinary division or cancellation without checking that it remains valid for the modulus. Composite moduli are the stated warning case.

LINE	INVARIANT / DECISION
a = qb + r	Replace (a,b) by (b,r).
r ≠ 0	Continue the algorithm.
r = 0	Previous non-zero remainder is the gcd.

W5 · Bayes / Base Rate C9 WORKED

Prompt. 1% of files are malicious. A detector catches 90% of malicious files and falsely flags 5% of benign files. Find P(malicious | flag).

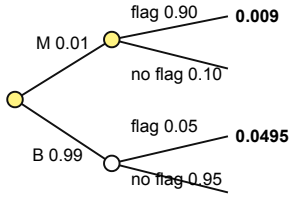


Figure 20. Both paths contribute to the flag total.

1. Malicious and flagged: 0.01×0.90 = 0.009.
2. Benign and flagged: 0.99×0.05 = 0.0495.
3. Total flagged: 0.009 + 0.0495 = 0.0585.
4. Posterior: 0.009 / 0.0585 ≈ 0.154.

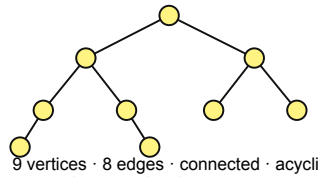
ANSWER

`P(malicious | flag) ≈ 15.4%`

TRAP *P(flag | malicious)=90% is not P(malicious | flag). The base rate changes the denominator.*

Graph Rapid Check C10 WORKED

Claim. A connected undirected graph has 9 vertices and 8 edges. Is it necessarily a tree?



9 vertices · 8 edges · connected · acyclic

Figure 21. Edge count works with connectivity.

Answer: yes, under the stated undirected graph convention: n=9 and m=8=n−1, while connectivity excludes separate cyclic and isolated structure. Never use the edge count without the stated condition.

GRAPH WORD	WHAT MUST BE VISIBLE
Path	A sequence obeying the graph's movement rules.
Connected	Every relevant vertex pair is reachable.
Acyclic	No cycle occurs under the declared convention.
Tree	Connected + acyclic; unique simple paths.
Weighted / directed	State what edge direction or weight changes.

Graph audit order: declare directed or undirected, then loops/parallel edges/weights, then test the requested property. The theorem comes after the convention.

Method Chooser C1–C10

QUESTION STEM	METHOD	MANDATORY CHECK
"How many in either set?"	Inclusion–exclusion	Intersection subtracted once
"Is this a function / injective?"	Apply definitions to domain and codomain	All eligible inputs
"Prove P→Q"	Direct or contrapositive	Direction preserved
"For all n…"	Induction if successor structure fits	Base and step
"Equivalent?"	Compare every truth valuation	Not just one row
"Negate ∀/∃"	Switch quantifiers; negate predicate	Domains unchanged
"Sum a, ar, …"	Finite geometric sum	Index count
"Select k, order irrelevant"	Combination	Outcome sameness
"Find gcd"	Euclidean algorithm	Last non-zero remainder
"Probability after evidence"	Bayes' rule	Condition direction + total evidence
"Tree?"	Connectivity / acyclicity / edge test	Graph convention

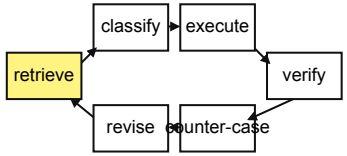


Figure 22. Retrieval improves through changed counter-cases.

Classic Traps ALL FAQ

- **Domain omitted:** truth, function properties and quantifiers become ambiguous.
- **Evidence mistaken for proof:** checked examples do not establish a universal.
- **Converse drift:** proving Q→P does not prove P→Q.
- **Index drift:** exponent 9 can coexist with ten terms when indexing begins at 0.
- **Counting by reflex:** factorials appear before order and symmetry are defined.
- **Conditional reversal:** likelihood is reported as posterior.
- **Graph convention omitted:** a theorem is applied despite directedness, loops, parallel edges or weights.

Exam-Morning Plan COURSE STRATEGY

PASS	ACTION
1 · Recite	Ten definitions and ten displayed formulas without notes.
2 · Classify	For each prompt, name method before calculating.