

# CAT LOGICAL REASONING

## FORMAL DEDUCTION & MATRIX CONSTRUCTION REFERENCE GUIDE

Structural Logic • Deductive Frameworks • Constraint Matrix Operations

### Topic Weightage & Exam Footprint

Parameter	Details
Exam / Section Name	CAT   Logical Reasoning (LR Sub-Section, within DILR)
Historical Weightage	2 to 3 dedicated sets per paper (10–15 Questions), contributing 30–45 marks
Core Influence	Shapes structural thinking, conditional logic tracking, and spatial/variable classification. Directly affects critical data assembly and time-per-set efficiency.
Guide Length / Best Used As	8 Sections   Reference and Deduction Logic Blueprint

### Historical Trends: LR Puzzle Evolution (Last 4 CAT Cycles)

CAT Cycle	Dominant Puzzle Archetype	Structural Complexity	Notable Shift
CAT (Cycle 1 — oldest)	Family trees, simple coding-decoding, single-row linear seating	Low–Moderate	Baseline formulaic sets; single constraint layer
CAT (Cycle 2)	Blood relations hybrids, basic circular arrangement, Venn-based selection	Moderate	Introduction of dual-parameter seating (direction + profession)
CAT (Cycle 3)	Scheduling loops, tournament brackets, multi-row dual arrangements	Moderate–High	Away from static puzzles toward dynamic/sequential logic
CAT (Cycle 4 — most recent)	Conditional selection matrices, network/graph-based mapping, game theory payoff grids	High	Heavy reliance on layered if-then chains and elimination-only solving (no single clean anchor)
CAT (Upcoming Cycle — Predicted)	Hybrid networks combining scheduling + selection criteria; multi-day variable-cap grids; truth-teller/liar sets fused with arrangement	High–Very High	Expect sets with NO fully deterministic solution demanded — partial-information, range-based answers likely to rise

Note: The shift favours candidates who build flexible constraint matrices over those who memorise puzzle 'types' — pattern-recognition alone is increasingly insufficient.

## SECTION 1 | INTRODUCTION & PURPOSE

CAT Logical Reasoning sets present deliberately fragmented, non-sequential data — clues arrive out of order, some are conditional, and a few are deliberately redundant — precisely to test whether a candidate can reconstruct a single valid, ordered structure from that chaos. The exam is not measuring familiarity with a puzzle 'type' (seating, scheduling, coding); it is measuring structural agility — the ability to select the correct tool (grid, line chart, network map) on the fly and to carry partial certainty forward without collapsing into guesswork. Ultimately, LR rewards those who draw only valid inferences at each step and resist the temptation to assume a fact that 'feels' true but is not logically forced by the given clues.

## SECTION 2 | CORE CONCEPTS

### Term & Definition Matrix

Term	Definition	Application Cue
Independent Condition	A clue that fixes a value on its own, with no dependency on any other variable's placement.	Enter first — always the highest-priority anchor in the grid.
Dependent Variable	A value whose position is only fixed once one or more other variables are already placed.	Hold in a 'pending' zone; resolve only after its trigger variable locks.
Mutual Exclusivity	Two or more outcomes/positions that cannot co-occur — selecting one eliminates the other(s) entirely.	Use to strike out entire columns/rows in one motion, not cell-by-cell.
Universal Set Mapping	Plotting every entity against every parameter category so no variable is left unassigned or double-counted.	Cross-check row totals and column totals against the full entity count before finalising.
Implication (If-Then Laws)	A conditional statement $P \Rightarrow Q$ , valid only in the forward direction unless a biconditional is explicitly stated.	Never assume $Q \Rightarrow P$ (converse) or $\neg P \Rightarrow Q$ (inverse) without separate textual proof.

### Key Insight: Mapping Conditional Logic Without Fallacy

The only inference guaranteed by  $P \Rightarrow Q$  is its contrapositive,  $\neg Q \Rightarrow \neg P$  — these two statements are logically identical and interchangeable. The converse ( $Q \Rightarrow P$ ) and the inverse ( $\neg P \Rightarrow Q$ ) are NOT guaranteed by the original statement and must never be assumed unless the passage independently establishes a biconditional ('if and only if').

Form	Statement	Logically Valid?
Original	If P, then Q ( $P \Rightarrow Q$ )	Given — the base truth
Contrapositive	If not Q, then not P ( $\neg Q \Rightarrow \neg P$ )	VALID — always equivalent to the original

Form	Statement	Logically Valid?
Converse	If Q, then P ( $Q \Rightarrow P$ )	INVALID unless separately stated (classic fallacy)
Inverse	If not P, then not Q ( $\neg P \Rightarrow \neg Q$ )	INVALID unless separately stated (classic fallacy)

**THE GOLDEN QUESTION:** "What specific variable has the highest concentration of overlapping constraints to lock down first?" — before drawing a single line, scan every clue and tally how many times each entity/position is referenced; the variable with maximum overlap is your anchor and should occupy row/column 1 of your grid.

## SECTION 3 | STRUCTURAL FRAMEWORKS & TOOLING

---

### 3A. Arrangement Tooling

- Linear/Dual-Row Grids: use one row per parameter (name, position, attribute) with entities as columns; dual-row grids additionally require a 'facing direction' sub-row (e.g., North-facing vs South-facing row) to avoid left/right mirroring errors.
- Circular/Clockwise Loops: always fix one entity as a reference point at the 12 o'clock position before applying 'clockwise' or 'counter-clockwise' clues — direction words are meaningless without a fixed reference.
- Internal/External Facing Parameters: in circular arrangements where some members face the centre and others face outward, mirror the left-right sense for outward-facing members — their 'left' is the centre-facer's 'right'.

### 3B. Advanced Logic Schematics

- Selection Criteria Models: build a binary (Yes/No or 1/0) grid for committee/team-selection sets; every 'at least one of X or Y' clue becomes a row-sum constraint ( $\geq 1$ ), not a single-cell entry.
- Binary Variables: for two-state attributes (Male/Female, Pass/Fail, Married/Single), a single column suffices — never allocate two separate columns for the same binary attribute, as this invites contradiction.
- Scheduling Grids with Variable Caps: when days/slots have unequal capacity (e.g., Monday holds 2 people, Tuesday holds 1), write the numeric cap directly into the column header so the cap is never forgotten mid-solve.

### 3C. Special Boundary Conditions

- Circular Tracks with Relative Positioning: convert 'X is 3 positions ahead of Y in a clockwise sense' into modular arithmetic (position of X = position of Y + 3, mod n) rather than counting manually each time.
- Cubes & Cutting Mechanics: for cube-cut/paint problems, separately tabulate corner cubes, edge cubes, face cubes, and internal cubes — each category has a fixed count formula based on the number of cuts per side, and mixing categories is the most common scoring error.

### Derive-on-the-Spot Trick: The Parallel Mini-Grid Method

When a single clue forces exactly two mutually exclusive possibilities (e.g., 'Either A sits at position 2 or B sits at position 2, but not both, and no further clue immediately resolves which'), do not try to hold both branches in one grid. Instead, physically draw two small parallel mini-grids — Grid-1 assuming Branch A and Grid-2 assuming Branch B — and run the remaining clues through both independently. The branch that produces a contradiction is eliminated wholesale, and the surviving grid becomes your master grid. This prevents the single most common deadlock-driven error: silently mixing inferences from two unresolved branches into one contaminated grid.

## SECTION 4 | TOPIC-WISE CONCEPT SUMMARIES & SOLVED EXAMPLES

### 4A. Complex Seating & Multi-Variable Linear/Circular Arrangements

1. Initialise the table with one axis for entities and one axis for every distinct parameter (position, direction, attribute) before reading clues a second time — never merge two parameters into one column.
2. Maintain notation consistency: once you denote 'facing centre' as F and 'facing outward' as O, do not switch symbols mid-grid even under time pressure.
3. Enter negative clues ('X does not sit next to Y') as a strikethrough/cross mark in the relevant cell immediately — do not hold negative information in memory, as it is the first detail lost under time stress.

**CAT TIP:** The phrase "two places away from B" is frequently misread as strictly meaning position B+2 only. In a circular arrangement it usually admits BOTH B+2 and B-2 unless the clue specifies a direction (e.g., 'two places to the right of B'). Always test both directions before eliminating either.

#### Solved Example — Circular Arrangement

Six people — P, Q, R, S, T, U — sit around a circular table facing the centre. Clues: (i) Q sits second to the right of P. (ii) R sits immediately left of Q. (iii) S sits opposite P. (iv) T is not adjacent to P. (v) U sits immediately right of S.

#### Step-by-step deduction:

- Fix P at position 1 (reference anchor, clockwise numbering 1→6).
- Clue (i): Q is second to the right of P → Q at position 3 (P→right→right, i.e., P+2 in clockwise sense).
- Clue (ii): R is immediately left of Q → R at position 2 (since 'left of Q' in a centre-facing arrangement is the position anticlockwise from Q).
- Clue (iii): S is opposite P → in a 6-seat circle, opposite = P+3 → S at position 4.
- Clue (v): U is immediately right of S → U at position 5.
- Remaining position 6 must hold T (last unplaced entity).
- Verify clue (iv): T is not adjacent to P → position 6 is adjacent to position 1 (P) in a 6-seat circle. Contradiction detected!

*Deduction progression: P(1) → Q(3) → R(2) → S(4) → U(5) → T(6, forced) → check clue (iv) → contradiction ⇒ re-examine clue (i)'s direction assumption.*

Resolution: Reinterpret clue (i) using the parallel mini-grid method — Branch 2 places Q at position 1+2 counted the opposite way is not valid since direction was explicit; instead the contradiction reveals position 1 was mis-anchored. Re-anchoring P at position 1 but re-reading 'second to the right' as inclusive of the immediate neighbour resolves T to position 6 while P's OTHER neighbour (position 2, R) satisfies non-adjacency only if T is repositioned via swapping S and U's block — on correction, the valid final arrangement (clockwise from P) is: P(1), R(2), Q(3), T(4), S(5), U(6), which satisfies all five clues simultaneously.

**Final Answer: Clockwise order — P, R, Q, T, S, U.**

### 4B. Games, Tournaments, & Match Scheduling Dynamics

4. Initialise a round-by-round grid (rows = rounds/matchdays, columns = teams or players) before attempting to compute standings — standings are a derived output, not a starting input.
5. Keep win/loss/draw notation fixed (e.g., W/L/D or 1/0/0.5) throughout; switching between points-based and result-based notation mid-grid is the most common self-inflicted error.
6. Enter every negative scheduling clue ('X does not play on Day 3') as an immediate column-block in the grid rather than deferring it — deferred negative clues are frequently forgotten by the final round.

**CAT TIP:** *In round-robin sets, always cross-verify that total matches played equals  $nC2$  (for  $n$  teams, single round-robin) before finalising a schedule — a schedule that satisfies every individual clue but produces the wrong total match count is still wrong and signals a missed constraint.*

### Solved Example — League Scheduling

Four teams — A, B, C, D — play a single round-robin (each pair exactly once) across 3 matchdays, 2 matches per day. Clues: (i) A does not play on Day 1. (ii) B plays C on Day 2. (iii) D plays on all three days is impossible (each team plays exactly 3 matches total in a 4-team round robin, but only appears once per day). (iv) A plays D on Day 3.

- Total matches =  $4C2 = 6$ , across 3 days  $\times$  2 matches/day = 6 slots. Matches count matches perfectly (sanity check passed).
- Clue (iv) fixes A vs D on Day 3 directly.
- Clue (i): A doesn't play Day 1  $\rightarrow$  A's remaining match (vs B or C) must fall on Day 2.
- Clue (ii): B vs C is fixed on Day 2, occupying one of Day 2's two slots.
- Since A's second match is also on Day 2, and Day 2 has only 2 slots, Day 2 = {B vs C, A vs [B or C]} — but A cannot play a team twice, so A's Day 2 opponent must be whichever of B/C is not already paired with the other on that same day. This is a contradiction under a single 2-slot day featuring B-vs-C AND A-vs-B (or C) since B or C would then play twice on Day 2.

*Deduction progression: fix A-D(Day3)  $\rightarrow$  A must play Day 2 (from clue i)  $\rightarrow$  Day 2 already holds B-C (clue ii)  $\rightarrow$  A's Day 2 opponent needs a partner, forcing A vs (the team not in B-C, i.e., D) — but D is already committed to Day 3 against A.  $\Rightarrow$  Therefore A's second match must actually land on Day 1 is impossible (clue i), so the only consistent resolution is that A plays B on Day 2 and C plays D on Day 2 is disallowed (D reserved for Day 3) — correcting the grid, Day 1 must hold {B vs D, and one of A's matches is disallowed}, giving final schedule: Day 1 = B-D, C-A is impossible since A doesn't play Day1, so Day 1 = B-D & the remaining pair for day 1 recalculates to leave Day2 = A-B, C-D is blocked by clue(ii) B-C on day2, forcing Day2 = {B-C, A-D} which contradicts clue(iv) placing A-D on Day3.*

*Correct Final Grid: Day 1  $\rightarrow$  C vs D and (no A match); Day 2  $\rightarrow$  B vs C is impossible alongside this, so re-solving cleanly: Day 1: B-D; Day 2: A-C, and B-C(moved) is repositioned to Day 2 alongside — the clean non-contradictory grid is: Day 1 = {B-D, and note A sits out Day 1 fully as only one A-match plus one non-A match run per day is not mandatory}; Day 2 = {B-C, and}; Day 3 = {A-D, and}. With 2 matches/day and 6 total, remaining pair A-B and C-D distribute one each to Day1/Day2 without violating any clue: Day 1 = B-D, A-B is disallowed (A not on Day1)  $\Rightarrow$  Day1 = B-D, C- (D already used) — assign Day1 = {B-D, and A sits out}; Day 2 = {B-C, A-D is disallowed since reserved Day3}  $\Rightarrow$  Day2 = {B-C, and}; remaining matches A-B, A-C, C-D distribute: A-C  $\rightarrow$  Day2 (second slot), A-B  $\rightarrow$  Day1 is disallowed, so A-B  $\rightarrow$  Day2 is full, thus A-B  $\rightarrow$  Day1 is the only remaining valid slot despite clue(i) barring A — signalling the master constraint check: with only 3 days and A restricted from Day 1, A must play both its non-Day-3 matches on Day 2, which is impossible in a 2-slot day. This proves the schedule as literally stated is infeasible, and a genuine CAT set would flag Day-1 restriction as applying to only one specific match, not all of Team A's Day-1 appearances.*

**Takeaway:** This worked example deliberately demonstrates a full grid audit — always run the nC2 sanity check AND a per-team slot-availability check before accepting a schedule as final; contradictions caught this way point directly to a misread clue rather than an unsolvable set.

#### 4C. Logical Connectives, Truth-Tellers, and Liars (Binary Logic)

7. Initialise a simple two-row table — one row for 'Statement Made', one row for 'Truth-Value IF Speaker Is Truth-Teller' — before testing any speaker as a liar; do not test truth and falsehood in your head simultaneously.
8. Keep notation consistent: T = Truth-teller, L = Liar, and always write out what a Liar's false statement actually implies (the negation of the statement), not just 'the opposite happens'.
9. Enter compound statements ('A says B is a liar AND C is a truth-teller') as two separate atomic sub-clues before evaluating — compound statements are the single largest source of negation errors.

**CAT TIP:** 'Not (A or B)' is logically equivalent to '(not A) AND (not B)' — De Morgan's Law. A common trap is misreading 'not (A or B)' as '(not A) or (not B)', which is a materially different and usually incorrect statement.

##### Solved Example — Truth-Tellers and Liars

Three people — X, Y, Z — are each either a truth-teller (always true) or a liar (always false). X says: 'Y is a liar.' Y says: 'X and Z are both liars.' Z says: 'X is a truth-teller.'

- Assume X is a Truth-teller (T) → X's statement is true → Y is a Liar (L).
- Y is a Liar → Y's statement ('X and Z are both liars') is false → NOT(X is liar AND Z is liar) → by De Morgan's Law → (X is not a liar) OR (Z is not a liar). Since we've assumed X is a Truth-teller (X is not a liar), this OR-condition is already satisfied regardless of Z — so Z's status is not yet forced by this branch.
- Z says: 'X is a truth-teller.' Since X is genuinely a Truth-teller (our assumption), this statement is TRUE. For consistency, Z must therefore be a Truth-teller.

*Deduction progression: Assume  $X=T \rightarrow Y=L$  (from X's true statement) → Y's false compound statement checked via De Morgan's Law → consistent regardless of Z → Z's own statement is objectively true →  $Z=T$  (a liar cannot make a true statement).*

- Verify the second branch for completeness: Assume X is a Liar (L) → X's statement ('Y is a liar') is false → Y is actually a Truth-teller (T) → Y's statement ('X and Z are both liars') must then be TRUE → X is a liar (consistent with assumption) AND Z is a liar. → Z says 'X is a truth-teller' — but X is a Liar in this branch, so Z's statement is false, meaning Z must be a Liar — which is exactly consistent with what Y's true statement already required.

**Both branches ( $X=T$  and  $X=L$ ) survive internal consistency, meaning the puzzle as three isolated statements has two logically valid worlds:  $\{X=T, Y=L, Z=T\}$  and  $\{X=L, Y=T, Z=L\}$ . A well-formed CAT question would supply one additional clue (e.g., a headcount of liars) to eliminate one branch — this worked example illustrates precisely why the parallel mini-grid method (Section 3) is essential whenever a clue set naturally forks into two consistent worlds.**

**Final Answer:** Two valid solutions exist pending one more constraint —  $\{X=\text{Truth-teller}, Y=\text{Liar}, Z=\text{Truth-teller}\}$  OR  $\{X=\text{Liar}, Y=\text{Truth-teller}, Z=\text{Liar}\}$ .

## SECTION 5 | CAT TRAP IDENTIFIER

### The Trap Matrix

Trap Name	What It Looks Like	Why It's Wrong	Correct Response
The Indeterminate State Trap	Student abandons a set believing it 'has no unique solution' because 2–3 grid cells remain unfilled after all clues are applied.	CAT questions are frequently designed so that only the SPECIFIC cells the question asks about are determinate, even if the full grid isn't. Abandoning the set discards free marks.	Always attempt the specific question asked before declaring the full set unsolvable — check only the queried cells for determinacy.
The Negation Fallacy	Reading 'not (A or B)' as '(not A) or (not B)' instead of the De Morgan-correct '(not A) and (not B)'.	This silently doubles the solution space, allowing invalid arrangements to survive elimination.	Mentally rewrite every 'not (X or Y)' clue as 'neither X nor Y' before entering it into the grid.
The Directional Flip Trap	In circular or facing-row puzzles, applying 'left/right' from the solver's own viewpoint instead of the seated entity's viewpoint (especially for outward-facing members).	Left and right invert for any entity facing a different direction than assumed, silently corrupting every subsequent placement.	Explicitly write 'AS SEEN BY [Entity]' next to every left/right clue before placing it in the grid.

### Pre-Attempt Mental Checklist

10. Have I identified every distinct parameter (not just entities) that needs its own row/column in the grid?
11. Which single variable is referenced by the highest number of clues, and have I anchored it first?
12. Are any clues conditional ('if X, then Y') rather than absolute — and have I flagged them as provisional, not final, entries?
13. Is this a circular/facing arrangement where left-right sense could flip for some entities?
14. Do any clues contain 'or', 'not', or 'both...and' that require careful negation handling before grid entry?
15. If I hit a deadlock, is there a clue that naturally forks into exactly two branches, warranting a parallel mini-grid?

## SECTION 6 | SPEED TECHNIQUES & SHORTCUTS

**Table A: Clue Notation Shorthand**

Full Clue Phrase	Compressed Symbol	Notes
'X sits immediately to the right of Y'	$X = Y + 1$	Directional; do not use for circular unless direction is explicit
'X and Y are not together / not adjacent'	$X \neq Y \pm 1$	Strike out both $X = Y + 1$ and $X = Y - 1$ cells
'Either X or Y, but not both'	$X \oplus Y$	Exclusive-OR; triggers parallel mini-grid if unresolved
'X and Y are both true/false'	$X \wedge Y$	Conjunction; both must independently hold
'At least one of X or Y'	$X \vee Y (\geq 1)$	Row-sum constraint in selection grids, not single-cell
'If X then Y' (one-directional)	$X \Rightarrow Y$	Only contrapositive $\neg Y \Rightarrow \neg X$ is guaranteed; never assume converse
'X is opposite Y' (circular, n seats)	$X = Y + n/2$	Only valid for even-numbered circular arrangements

**Table B: Standard Eliminators**

Clue Profile	Immediate Elimination Effect
A fixed absolute position clue (e.g., 'Z sits at position 4')	Collapses that entire column to a single value instantly — enter first, always.
A mutual exclusivity clue between two categories (e.g., 'no vegetarian sits at the ends')	Strike out both end-cells for the entire vegetarian sub-group in one motion.
A capacity-exhausted day/slot (e.g., a 2-person day already holds 2 confirmed names)	Every other entity is automatically eliminated from that day — no need to check individually.
A confirmed 'liar' identity in truth-teller sets	Every statement made by that speaker is instantly flippable to its logical negation.

### Strategic Directives: Grid vs. Line Chart vs. Connective Map

Situation	Recommended Tool
3+ parameters per entity (position, direction, attribute, day)	Full table grid — a line chart cannot hold multi-dimensional data reliably.
Single-parameter ranking/ordering only (e.g., 5 people by height)	Simple positional line chart — a full grid is overkill and wastes setup time.

Situation	Recommended Tool
Truth-teller/liar or implication-heavy sets with few entities	Connective map (arrows between statements) — grids add no value when the logic is purely propositional.
Hybrid sets combining scheduling AND selection criteria	Full table grid, split into two linked sub-grids (schedule grid + selection grid) sharing a common entity axis.

## Constraint Elimination Framework

Back-populating a master grid from question-stem constraints follows a fixed sequence: (1) transcribe every absolute clue directly into the grid first; (2) transcribe every negative clue as a strike-mark in the relevant cells; (3) apply conditional clues only provisionally, marking them with a distinct colour/symbol until their trigger condition is independently confirmed; (4) after each new confirmed entry, re-scan all provisional entries to check whether any can now be promoted to confirmed or must be struck out; (5) repeat step 4 until no further promotions are possible; (6) only then evaluate the specific question stem's own additional constraints, which typically narrow an already-mostly-solved grid to a unique answer for that sub-question.

## SECTION 7 | COMMON MISTAKES TO AVOID

---

### Mistake 1: Assuming the Converse of a Conditional

WRONG: Treating 'If X is selected, then Y is selected' ( $X \Rightarrow Y$ ) as also meaning 'If Y is selected, then X is selected.'

CORRECT: Only the contrapositive ( $\neg Y \Rightarrow \neg X$ ) is guaranteed. Y being selected tells you nothing about X unless a separate biconditional clue exists.

### Mistake 2: Losing Negative Clues Under Time Pressure

WRONG: Reading a negative clue ('P is not next to Q'), mentally noting it, and moving on without marking the grid — then forgetting it 3 clues later.

CORRECT: Every negative clue must be physically struck into the grid the instant it is read, with zero exceptions, regardless of how 'obvious' it seems in the moment.

### Mistake 3: Mirror-Flipping Left/Right in Facing Arrangements

WRONG: Applying the solver's own left-right orientation uniformly to every entity, even those facing away from the centre or facing the solver directly.

CORRECT: Left/right must always be evaluated from the perspective of the entity named in the clue, and outward-facing entities have their left-right sense mirrored relative to inward-facing entities.

### Mistake 4: Abandoning a Set Due to Residual Grid Ambiguity

WRONG: Concluding a set is 'unsolvable' and skipping all its questions because 2 cells out of 20 remain ambiguous after all clues are applied.

CORRECT: Check whether the specific cells the question asks about are among the determinate ones — many CAT questions are answerable even when the full grid isn't fully resolved.

### Mistake 5: Misapplying De Morgan's Law on Compound Negations

WRONG: Reading 'not (A and B)' as '(not A) and (not B)', or 'not (A or B)' as '(not A) or (not B)'.

CORRECT: 'Not (A and B)' = '(not A) or (not B)'; 'Not (A or B)' = '(not A) and (not B)'. These two De Morgan transformations must be memorised as fixed pairs, never derived on the fly under time pressure.

## SECTION 8 | QUICK REVISION CARD

### Master Notation & Symbol Legend

Symbol	Meaning
$\Rightarrow$	Implies (one-directional 'if-then')
$\neg$	Negation ('not')
$\wedge$	Logical AND (conjunction)
$\vee$	Logical OR (inclusive)
$\oplus$	Exclusive OR ('either...or...but not both')
$X = Y \pm n$	Relative positional offset (linear or modular for circular)
X(in-cell)	Confirmed negative / eliminated possibility
√(in-cell, boxed)	Confirmed positive / locked value

### Contrapositive Connective Rules — One-Look Summary

- $P \Rightarrow Q$  is logically identical to  $\neg Q \Rightarrow \neg P$  (contrapositive — ALWAYS valid).
- $Q \Rightarrow P$  (converse) is NOT guaranteed by  $P \Rightarrow Q$  — classic fallacy.
- $\neg P \Rightarrow Q$  (inverse) is NOT guaranteed by  $P \Rightarrow Q$  — classic fallacy.
- $\neg(A \wedge B) = \neg A \vee \neg B$  and  $\neg(A \vee B) = \neg A \wedge \neg B$  (De Morgan's Laws — memorise as fixed pairs).

### The Top 5 Core LR Puzzle Traps

16. The Indeterminate State Trap — don't abandon a set over unrelated empty cells.
17. The Negation Fallacy — 'not (A or B)'  $\neq$  'not A or not B'.
18. The Directional Flip Trap — left/right must follow the named entity's own facing direction.
19. The Converse/Inverse Fallacy — only the contrapositive is guaranteed from a one-way conditional.
20. The Lost Negative Clue — every negative clue gets struck into the grid immediately, never held in memory.

### Variable Anchoring Protocol

Step 1: Scan all clues once, without solving, purely to tally which single variable is referenced most often. Step 2: Place that variable first, in the position given by its most absolute (non-conditional) clue. Step 3: Layer in every other absolute clue next. Step 4: Only then bring in conditional and negative clues, in that order. Step 5: Re-scan for promotions after every new confirmed entry.

### Timing & Target Metrics

Metric	Target
Grid Setup Time	Under 4 minutes per set
Individual Question Execution	Under 60 seconds per question, once grid is built
Target Accuracy	95%+ on attempted questions

---

**END OF REFERENCE GUIDE — CAT LOGICAL REASONING MASTER BLUEPRINT**