

GAMES: If you're writing, you're doing it right. If you're staring...

Games: 4-step method

- 1) Write down the entities: # type characters.
 - * Consider building entities directly into the sketch.
- 2) Create a sketch based on the type of game.
 - * Almost always use slots (_ _ _) or a chart
- 3) Write down the rules.
- 4) Answer the questions appropriately.

Games step 2: types and sketches

- 1) **Sequencing**: standard, loose, and double (sketch = slots)
 - * Entities are placed in order.
 - * Given days of the week, months, or years, place these on top of a chart.
- 2) **Yes/no** (sketch = chart)
 - * Only two choices in placing a set of entities.
 - Ex: sale/no sale, Oakland/Berkeley, on/off
- 3) **Distribution** (sketch = chart)
 - * A larger set of entities is placed among a smaller set.
 - * Given two sets of entities, place the smaller set on top of the chart.
- 4) **Matching** (sketch = chart)
 - * Entities are matched to one another.
 - * The game might describe a specific sketch.
- 5) **Selection** of a specific number (sketch = labeled slots)
 - * A specific number of entities is to be selected from a group of multiple entities.
- 6) **Combination** (charts = slots in a chart, chart/grid)
 - * Game types are combined.

Games step 3: write down the rules

Do the following whenever possible:

- 1) Build rules into the sketch.
 - * Use slashes to build in “remaining” entities.
 - * Build in “no-consecutive” rule.
- 2) Rephrase rules as **blocks**.
 - * Combine rules to create blocks.
 - * “No-consecutive” rules.
 - * Build large blocks into the master sketch.
 - * Eliminate blocked entities from the “ends” of the game (use sparingly).
- 3) Rephrase rules into “**if/then**” form.
 - * Recall contra-positive.
 - * Build “special if/then” rules into the sketch.
- 4) Rephrase rules into specific **numbers**.
 - * Consider “0” as a possibility.
 - * Complete number possibilities.
 - * Math: begin by assigning “0” to the lower-numbered entity.
 - * If/then: recall contra-positive
 - * Look for other rules that limit the lowest or highest numbers.
- 5) Rephrase rules as **groups** or **split-groups**.
 - * Use slashes to place split-groups.
- 6) Note any floating entities.
 - * Use slashes to place interchangeable floaters.
- 7) Translate “exactly” into “only”.
- 8) Compare rules to make any additional deductions.

Games step 4: answer the questions appropriately

I. Define the question type by identifying the nature of the wrong answers.

* *Must be true*: wrong answers *could* or *cannot* be true but do not *have* to be true.

* *Must be false/CANNOT be true*: wrong answers *could* (or *must*) be true.

* *Could be true*: wrong answers *cannot* be true.

* *Could be false*: wrong answers *must* be true.

NOTE: **Could be** a complete and accurate: one specific example that works.

NOTE: **Is** a complete and accurate: all possibilities that work.

II. Answer the questions in the following order:

1) Line-up

2) "If" questions

3) "Non-if" questions

4) Rule change questions

* "Non-if" questions should be answered in order given any good deductions.

III. **Line-up** questions (when the answers provide a complete line-up of the game)

1) Select a single rule.

* Recall that rules are built into the sketch and in shorthand

2) Apply that rule to each answer, eliminating those answers that violate the rule.

3) Repeat process until only one answer remains.

* **Partial line-up** (when the answers provide only a partial line-up)

* Apply same method, but consider "completing" certain answers.

* **Line-up EXCEPT**:

* Apply the same method, but *select* the answer that violates a rule.

* Use the correct answer as "prior-work" when answering other questions.

IV. **"If"** questions

1) Build the "if" into the sketch.

* When the "if" is ambiguous, place entities anywhere allowed by rules.

2) Immediately check each rule one-by-one to build anything else into the sketch.

* Look for entities common to the rules and those built into the sketch.

* Build in blocks or numbers, eliminate ends, recall contra-positive.

* Complete the sketch when it is easy to do so.

* Consider creating 2 (or 3) options at once.

3) Evaluate the answers appropriately (see part VII).

V. **"Non-if"** questions

1) Use rules and deductions.

2) Use prior work.

3) Evaluate answers appropriately (see part VII)

VI. **Rule-change** questions

1) Apply the rule change only if you feel that you have the time.

VII. **Evaluating answers appropriately**

1) Compare each answer to the information in the sketch/prior work.

* If the correctness of an answer is not readily apparent, move on.

* Only eliminate those answers that are clearly wrong.

2) Build remaining answers into the sketch.

3) Be willing to make a quick guess and move on.

Formal logic (if/then statements)

The *contra-positive* is the only deduction that can be made from an if/then statement.

- I. Practical formal logic: “If it’s a rabbit then it is an animal.”
- 1) The “if” element acts as a trigger that forces a certain conclusion.
 - 2) The “then” element is the conclusion.
 - 3) Which of the following is a logical deduction from the if/then statement above?
 - a) If it is not rabbit, then it is not an animal. **NO**
 - b) If it is an animal, then it is a rabbit. **NO**
 - c) **If it not an animal, then it is not a rabbit. YES (Contra-positive)**
 - 4) **Contra-positive**: switch and negate the elements.
 - * Switch “and” to “or” and vice-versa.
- II. Abstract formal logic: If X then Y.
- 1) Contra-positive: If not Y then not X
 - 2) “If X *and* Y then Z”; contra-positive: If not Z then not X *or* not Y
 - 3) “If X *or* Y then Z”; contra-positive: If not Z then not X *and* not Y
 - * “not...and” = “neither...nor”

Formal logic exercise

- 1) If it’s a rabbit, then it is an animal.
Contra-positive:
- 2) If X then Y
Contra-positive:
- 3) If it’s a rabbit, then it is an animal and it breaths.
Contra positive:
- 4) If X then Y and Z
Contra-positive:
- 5) If it’s a rabbit, then it’s a boy or a girl.
Contra-positive:
- 6) If X then Y or Z
Contra-positive:
- 7) If it’s a rabbit, then its not a dog
Contra-positive:
- 8) If X then not Y
Contra-positive

Formal logic (continued)

III. Always recall the **contra-positive** when given an **if/then** rule

The contra-positive: every if/then rule has at least 2 “ifs” (triggers).

Ex. “If C then D”

2 “ifs” (triggers): 1) If C
2) If not D

Ex. “If F then G and H”

3 “ifs” (triggers): 1) If F
2) If not G
3) If not H

Ex. “If K then L or M”

2 “ifs” (triggers): 1) If K
2) If not L and not M (neither L nor M)

IV. The following can be rephrased as “if X then Y”

* Translating tricky rules into if/then form: if rain (X) then clouds (Y)

- a) Y if X
- b) X only if Y
- c) Only Y is X
- d) Y whenever X
- e) No X unless/without Y

* G if *but* only if H = If G then H **AND** If G then H

* No Fs are Ms = if F then not M

V. Two special if/then rules

1) If F then no G

* Since F and G can never be “**yes**” together, at least one must be “**no**”.

2) If no K then L

* Since K and L can never be “**no**” together, at least one must be “**yes**”.

YES	NO
(K/L) or both	(F/G) or both

3) NOTE: “If X then no Y **and** no Z” = “if X then no Y” and “if X then no Z”
“If no X **or** no Y then Z” = “if no X then Z” and “if no Y then Z”

