# **Pre-Coding Essentials (Component: tests/vm\_tst\_ranked.rs, Version/FormulaID: VM-ENGINE v0)**

## **1) Goal & Success**

Verify **ranked methods**: (a) **IRV with exhaustion**, (b) **Condorcet with Schulze completion**. Must match Annex B expected winners and logs.

Success: pipeline returns the correct **winner**, **RoundLog / PairwiseMatrix** evidence, and **Decisive** label.

## **2) Scope**

In: unit tests that drive full pipeline on **Annex B — Part 3** fixtures VM-TST-010/011.

Out: gates/frontier/mmp; RNG ties (not triggered in these cases).

## **3) Inputs → Outputs (with schemas/IDs)**

Inputs:

**DivisionRegistry / Units / Options** with deterministic order.

**BallotTally** (IRV rounds / Condorcet ballots).

**ParameterSet** with ballot\_type, ranked\_exhaustion\_policy or condorcet\_completion.

Outputs:

**Result**: executive winner + IRV summary (exhausted, continuing, final round). **Label: Decisive**.

**RunRecord**: provenance (timestamp etc.)—implicitly validated by pipeline.

## **4) Entities/Tables (minimal)**

## **5) Variables (only ones used here)**

## **6) Functions (signatures only)**

fn run\_irv\_exhaustion\_case() -> () — loads VM-TST-010, runs pipeline, asserts winner **B**, exhausted **10**, continuing **90**, final round {B:55, A:35}.

fn run\_condorcet\_schulze\_cycle() -> () — loads VM-TST-011, runs pipeline, asserts **Schulze winner B** and PairwiseMatrix reflects A>B 55–45, B>C 60–40, C>A 60–40.

## **7) Algorithm Outline (bullet steps)**

Load fixture bundle (REG/Options/TLY/PS).

Run pipeline: VALIDATE→TABULATE (ranked rules)→ALLOCATE→AGGREGATE→APPLY\_DECISION\_RULES→LABEL→BUILD\_RESULT/RUN\_RECORD.

Extract winner + audit payloads (IRV RoundLog or Condorcet PairwiseMatrix) and assert.

## **8) State Flow (very short)**

Follow Doc 5 state machine; no RNG ties expected. If a blocking tie appeared, RESOLVE\_TIES would serialize with policy/seed; not used here.

## **9) Determinism & Numeric Rules**

**IRV:** majority of **continuing ballots**; exhaustion removes ballots from denominator.

**Condorcet:** if no Condorcet winner, apply **Schulze** per VM-VAR-005.

Percent/rounding unaffected here; tests assert integer tallies and exact winners.

## **10) Edge Cases & Failure Policy**

Validate presence/shape of ranked preferences; missing/malformed → MethodConfigError (out of scope for “happy path” tests).

## **11) Test Checklist (must pass)**

**VM-TST-010 (IRV)** Inputs: 40×B>A>C, 35×A>C, 15×C>B, 10×C. Expect R1 A=35 B=40 C=25 → eliminate C; transfer 15 to B; **10 exhaust**; continuing=**90**; final **B=55**, **A=35**; **winner B**; **Decisive**.

**VM-TST-011 (Condorcet/Schulze)** Pairwise margins: A>B **55–45**, B>C **60–40**, C>A **60–40**; **winner B**; **Decisive**. Also assert PairwiseMatrix presence.

**Ready to code assertions** against Result winner and IRV/Condorcet audit payloads.