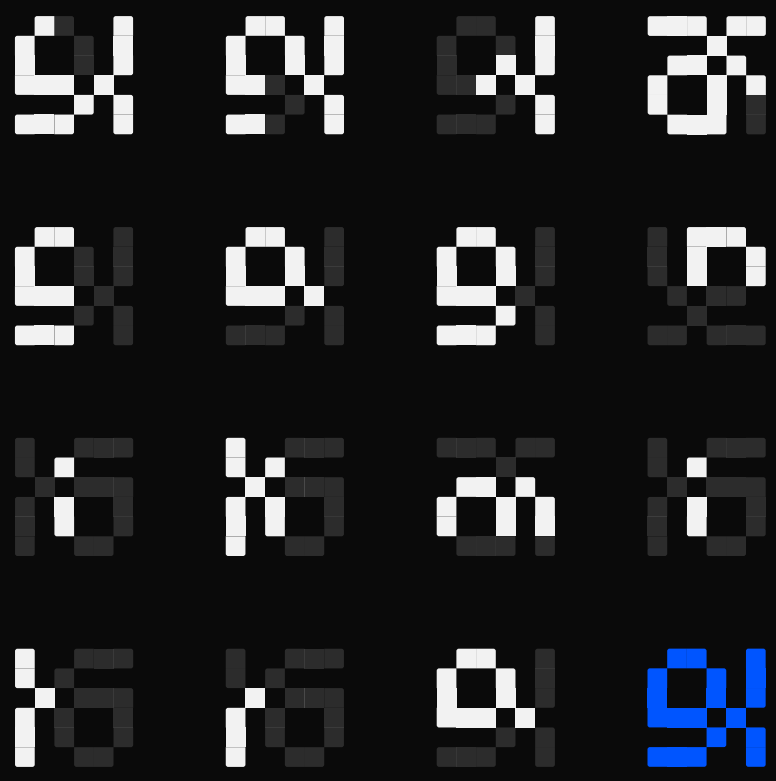


The Brilliant Employee · 05

Worse than a coin flip

The first exam my routing system ever took from a stranger. Mine scored 0.3108 on a 5,000 row replay. Random scores 0.5202 on the full run.

sgnk.ai/brilliant ↗

Sagnik Mitra

Why this exists

sgnk is a one-person studio. A language model writes a large share of the production code. You do not need your own system to get answers out of a model. You need one to find out when the answers were wrong, because wrong work reads exactly like right work and no prompt fixes that.

Three things from that week, none visible from inside the tool: a sizing step ignored on 796 of 816 tasks, two fixes I had written up as done that were never made, and a detector whose 1,112 catches were mostly one test string of my own.

THE SYSTEM, AS OF 5 AUGUST 2026, 00:39Z	
rules written	11 June, 55 days ago
ledger running	27 June, 39 days
tasks logged	2,959, across 3,397 rows
controls on tool calls	**6 registered, 3 of them armed**

You can rent the model. You cannot rent the part that tells you it was wrong.

Five parts, and where today sits

Five parts sit around the model. What changes from one piece of writing to the next is which of them is under the microscope.

the Gate	sizes a job before a model is picked
the Gambler	recommends which model gets the job
the Rails	refuse the irreversible, before it runs
the Judge	grades finished work pass or fail
the Ledger	at least one append-only line per task

Listed in the order they act on a task, not the order they were built.

Today is about the Gambler, marked above. The other four are the machinery around it, and this post does not assume you know any of them.

This page is the map. Nothing here needs another post to make sense.

A month of report cards, all written at home

My automation system has a component I call the Gambler, the part that recommends which model gets each job. Cheap model for routine work, expensive model when the job earns it.

For its first month, every grade the Gambler received came from inside my own workspace. The loudest was a 44 agent self-review campaign across the whole system. Verdict: world class on 2 of 10 dimensions, competitive on 8, behind on 0. Not one of those graders was built by someone else.

> **ASIDE.** That campaign was honest work. It measured coverage, whether a component exists for each concern. It could not measure ranking.

A grader that shares your assumptions inherits your blind spots.

The exam someone else wrote

> **Plain words: ROUTERBENCH.** A public benchmark of 401,467 precomputed inference outcomes, 36,497 rows scored against each of 11 models. You replay a routing policy over it and it scores the quality and the cost of the assignments the policy makes. Nobody who built it has ever seen my system.

Three numbers matter. Across the full 36,497 rows, a router that assigns models at random scores 0.5202 and one that always picks the cheapest model scores 0.3061. Replayed with exploration off over a 5,000 row subsample, my shipped configuration scored 0.3108, which on that subsample is exactly the always-cheap score, because it pulled the cheap arm on all 5,000 rows.

> **ASIDE.** Date of the exam: 2026-07-27 UTC, day 30 of the system's life. The first external measurement it ever had.

The first number from outside the building refuted the config.

Two verdicts, one system

Both columns are honest. They measure different things. The internal campaign measured coverage: does machinery exist for each concern. The exam measured behavior: do the assignments beat chance.

INTERNAL, SELF-RUN		EXTERNAL, ROUTERBENCH	
campaign	44 agents, mine	live config, 5,000 row replay	0.3108
world class	2 of 10	random baseline, full run	0.5202
competitive	8 of 10	always-cheap, full run	0.3061
behind	0 of 10	graders built by	strangers

Coverage was fine. Behavior lost to a coin. Nothing in the left column was positioned to notice.

Existence of machinery is not evidence the machinery works.

I misread the exam on the first pass

My first report said we passed: 0.7699, comfortably above random. The number was real and the verdict was wrong. I had added 5 percent forced exploration to the simulation to keep it from sticking, then forgot to account for it. The pass mark belonged to my test harness, not my router.

FIRST REPORT	CORRECTED
score **0.7699**	score **0.3108**
verdict **beats the coin**	verdict **loses to the coin**
hidden input **5% forced exploration**	hidden input **none**

> **ASIDE.** Both numbers sit in the same written report. The corrected one is in bold.

Even the outside exam got marked by my own test harness on the first pass.

Why it flatlined: the absorbing proof

1. Task zero: no model has the 5 observations the abstention gate requires
2. The policy falls back to the default arm, the cheap model
3. Only that arm accumulates observations, so only it can ever pass the gate
4. Every other model is scored minus infinity, forever
5. Result: 5,000 simulated tasks, one distinct model chosen 5,000 times

With exploration off, the entire tuning grid produced byte identical output. That is the signature of a policy whose parameters cannot matter. Weeks of arguing about those parameters, settled in one line.

> **ASIDE.** The 5 observation abstention gate is a good rule alone. Combined with zero exploration it is a trap that closes on the first arm touched.

A zero exploration policy does not route. It repeats.

What the exam bought

A refutation is only depressing until you spend it. Retuning the Gambler, the part that recommends the model, against the exam showed my pessimism setting was a red herring and the cost term was the real lever.

THE RETUNE, EXPLORATION ON, SCORED AGAINST ALWAYS-EXPENSIVE

cost weight 0, the shipped value, exploration on	90.0% quality at 76.1% cost
cost weight 100	**77.0% quality at 7.4% cost**
cost weight 1000	75.7% quality at 6.5% cost

One sweep of the cost term, every row with exploration on. The middle row is the recommended design point. The live rule still carries no cost term at all.

Exploration was armed 72 minutes later, in a separate commit under an unrelated subject line, carrying the minimum detectable effect calculation the arming rule had always required and nobody had done. The next verdict arrives with a sample size instead of a feeling.

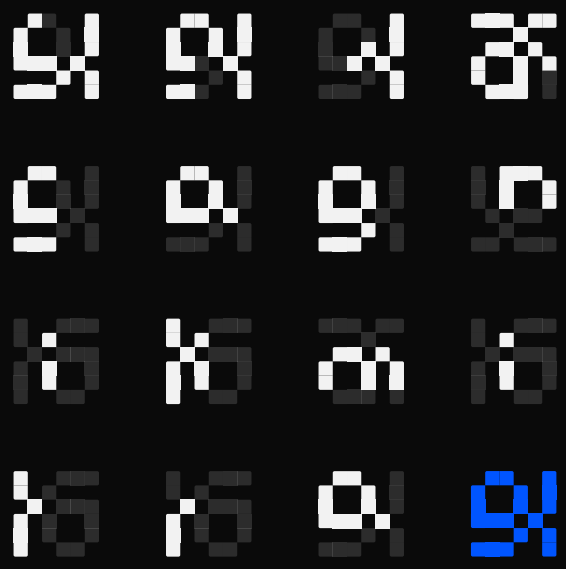
External exams do not just grade the system. They tune it.

The instrument you did not build

Every check I write shares my assumptions: my task mix, my definition of success, my simulation code, my blind spots. An instrument built by strangers disagrees with you in ways you cannot anticipate. That is not a flaw of the instrument. That is the product.

The audit written the day before the exam said flatly that no external benchmark had ever been run against this system. This was the first. No second one has run since. One exam in, the score stands at one refuted internal verdict, and every comparative claim I make about this system is self-assessed until a second instrument reports.

Self-assessment is rehearsal. The exam is the performance.



Homework graded by the family is practice, not a grade. The derivation, the misread, and the absorbing proof.

sgnk.ai/brilliant ↗

Sagnik Mitra

I build AI systems and the machinery that keeps them honest: gates, ledgers, judges, and rails. Product design for years, engineering the whole time. This series is what the building actually taught me.

`sgnk.ai · in/sagnikmitra · github.com/sagnikmitra`