Challenge 08: Swampland via Modularity Higher-Form Symmetries

Pure Thought AI Challenge 08

Pure Thought AI Challenges Project

January 18, 2026

Abstract

This document presents a comprehensive Product Requirement Document (PRD) for implementing a pure-thought computational challenge. The problem can be tackled using only symbolic mathematics, exact arithmetic, and fresh code—no experimental data or materials databases required until final verification. All results must be accompanied by machine-checkable certificates.

Contents

Domain: Quantum Gravity **Difficulty:** High **Timeline:** 9-12 months

0.1 Problem Statement

Use modularity, integrality, and higher-form symmetry constraints to produce theorem-level obstructions to QG-inconsistent CFT spectra.

0.2 Core Question

Which CFT partition functions are ruled out by modular invariance + discrete symmetries + anomaly constraints?

0.3 Mathematical Formulation

Constraints:

- **Modular invariance:** Z() = Z(-1/)
- **Integrality:** Fourier coefficients d(h) $_0$

- **Higher-form symmetries:** 1-form $Z_N, 2 - formsymmetries$
- **Anomaly matching:** 't Hooft anomalies for discrete symmetries
- **Cobordism:** Invertible phases from cobordism groups

Combined: Z() must satisfy modular + symmetry charges + anomaly equations
If infeasible → "No such CFT exists" theorem

0.4 Implementation

```
def modular_bootstrap_with_symmetry(c, symmetry_group, charges):
    # Setup partition function Z =    d(h)  _h (  )
    # Impose modular + symmetry constraints
    # Check feasibility via SDP
    pass

def anomaly_constraint(symmetry_data):
    # Compute 't Hooft anomaly
    # Match bulk/boundary anomaly
    pass

def cobordism_obstruction(spacetime_dim, symmetry):
    # Compute cobordism group   ^{st}(B G)
    # Check if theory can be gapped consistently
    pass
```

0.5 Example Prompt

```
Consider a 2D CFT with c=24 and Z_2    Z_2 higher-form symmetry.
Impose: (1) modular invariance, (2) symmetry charge assignments,
(3) anomaly matching with bulk SET. Determine if such CFT exists or
prove impossibility via infeasibility certificate.
```

0.6 Success Criteria

MVR: One new "no such CFT" result with dual certificate
Strong: Systematic scan over (c, symmetry group) → impossibility map
Publication: New swampland constraints from categorical consistency

0.7 Verification

```
def verify_no_cft_theorem(c, symmetry, certificate):
    # Verify certificate proves infeasibility of (modular + anomaly
       eqns)
    assert check_dual_certificate(certificate)
    assert certificate_gap < 0  # Proves impossibility
```