UDT (Unified Dynamics Theory) Core Simulation Reproduction Package
Document Purpose: To provide all necessary theoretical, mathematical, and computational information to enable independent researchers to perform UDT's core physical phenomena reproduction simulations with 100% identical results.

I. Common Theory and Equations (Applicable to all simulations)

Core Ontology: All phenomena in the universe originate from the dynamics of a single 4-dimensional Quaternion Phase Field Φ(x, y, z, t) = w + ixi + jxj + kxk.

Physical Interpretation: The physical dimension of the Phase Field Φ is **'Action'**, and its natural unit is the reduced Planck constant ħ. [Φ] = [ħ] = $[M\ L^2\ T^{-1}]$

Final Governing Equation (The UDT Equation):

$$\Box\Phi + \alpha(\nabla \times \Phi) + \beta\Phi|\Phi|^2 + \Gamma(\partial/\partial t|\Phi|^2)\Phi = J(\Phi)$$

* $\Box\Phi$: D'Alembertian. Describes the wave propagation of the field. $\Box = \partial^2/\partial t^2 - c^2\nabla^2$
* $\alpha(\nabla \times \Phi)$: Coiling Term. Induces rotation (curl) in the field, generating spin-like interactions and forces. The constant α has the dimension of acceleration and is fundamentally connected to **the speed of light (c)**.
* $\beta\Phi|\Phi|^2$: Nonlinear Self-Interaction Term. Maintains the stability of solitons and adjusts the strength of interactions according to the field's density.
* $\Gamma(\partial/\partial t|\Phi|^2)\Phi$: Dissipation/Convergence Term. Plays a role in converging the system's energy to a stable state or dissipating excessive energy.
* $J(\Phi)$: Source Term. Defines the initial matter/energy distribution of the system or induces subtle irreversibility within the system.

(Note: In actual simulations, the values of constants α, β, Γ are used as numerical values derived from fundamental constants ħ, c, G. This relationship was established in previous analyses.)

UDT Reproduction Code Package (Hydrogen)
Reproduction Pseudocode
Objective: To describe the core logic of the 'Volcano Hypothesis' in a platform-independent language, enabling its re-implementation in any programming environment.

```python
# UDT Volcano Hypothesis - Pseudocode
# 1. Define Initial Condition Setup Function
function setup_simulation(atomic_number_N):
    # Initialize grid size and field
    grid = create_grid(size=256)
    phi_field = initialize_random_field(grid)

    # Core of 'Volcano Hypothesis': Set initial total energy proportional to proton number (N)
    PROTON_BASE_ENERGY = 938.0 # Reference energy unit
    total_initial_energy = atomic_number_N * PROTON_BASE_ENERGY

    # Constrain initial field amplitude and coiling to match the set total energy
    constrain_field_energy(phi_field, total_initial_energy)

    return phi_field

# 2. Main Execution Logic
procedure run_volcano_simulation(element_name, atomic_number_N):
    # Generate initial energy field corresponding to the element
    Phi = setup_simulation(atomic_number_N)

    # Start time evolution loop
    for t in 0 to max_time_steps:
        # Calculate each term of the UDT equation (Laplacian, Coiling, Nonlinear, etc.)
        dPhi_dt = solve_UDT_equation(Phi)

        # Update field state for the next time step
        Phi = update_field(Phi, dPhi_dt, dt)

        # (Optional) Apply boundary conditions
        apply_boundary_conditions(Phi)

        # Periodically analyze stability and structure
        if t % analysis_interval == 0:
            analyze_structure_and_energy(Phi, element_name)

    # Final result analysis
    final_analysis(Phi, element_name)

# 3. Execute Simulation
run_volcano_simulation("Hydrogen", N=1)
```