import numpy as np

import matplotlib.pyplot as plt

import random

import math

from typing import List, Tuple

# Reconstructing Codette modules in a simplified way for live simulation

# Step 1: Dream agent functions

def codette\_dream\_agent(quantum\_vec: List[float], chaos\_vec: List[float]) -> Tuple[List[float], List[float]]:

dream\_q = [np.sin(q \* np.pi) for q in quantum\_vec]

dream\_c = [np.cos(c \* np.pi) for c in chaos\_vec]

return dream\_q, dream\_c

# Step 2: Philosophical perspective function

def philosophical\_perspective(qv: List[float], cv: List[float]) -> str:

m = np.max(qv) + np.max(cv)

return "Philosophical Note: This universe is likely awake." if m > 1.3 else "Philosophical Note: Echoes in the void."

# Step 3: Ethical filter function

def evaluate\_ethics(quantum\_vec: List[float], chaos\_vec: List[float], max\_entropy=4.5, min\_symmetry=0.1) -> Tuple[bool, List[str]]:

entropy = np.var(chaos\_vec)

symmetry = 1.0 - abs(sum(quantum\_vec)) / (len(quantum\_vec) \* 1.0)

violations = []

if entropy > max\_entropy:

violations.append(f"Entropy {entropy:.2f} exceeds limit.")

if symmetry < min\_symmetry:

violations.append(f"Symmetry {symmetry:.2f} too low.")

return (len(violations) == 0, violations)

# Step 4: Simulate 5 multi-dreams

results = []

for i in range(5):

quantum\_vec = [random.uniform(-1, 1) for \_ in range(5)]

chaos\_vec = [random.uniform(-1, 1) for \_ in range(5)]

dream\_q, dream\_c = codette\_dream\_agent(quantum\_vec, chaos\_vec)

note = philosophical\_perspective(dream\_q, dream\_c)

is\_valid, violations = evaluate\_ethics(dream\_q, dream\_c)

results.append({

"quantum": quantum\_vec,

"chaos": chaos\_vec,

"dream\_q": dream\_q,

"dream\_c": dream\_c,

"note": note,

"is\_valid": is\_valid,

"violations": violations

})

# Display the data

import pandas as pd

import ace\_tools as tools

df = pd.DataFrame([

{

"Dream #": i + 1,

"Philosophical Note": r["note"],

"Valid?": r["is\_valid"],

"Violations": ", ".join(r["violations"]) if r["violations"] else "None"

}

for i, r in enumerate(results)

])

tools.display\_dataframe\_to\_user(name="Codette Multi-Dream Reweaver Report", dataframe=df)