Preparation MCM Final checklist

Preparing for the **Mathematical Contest in Modeling (MCM)** is as much about workflow management as it is about math. It sounds like you have a solid grasp of the technical pillars—Linear Algebra, Probability, and Machine Learning—but the real challenge is the 96-hour sprint.

Here is your structured **MCM Preparation Self-Checklist**, organized by category to help you track your progress.

📊 Phase 1: Mathematical Foundations

High-level theory is great, but for MCM, you need to be able to apply these to messy data.

- **Linear Algebra Review:**
- ☐ Matrix decomposition (SVD/QR).
- ☐ Eigenvalues and Eigenvectors (essential for stability analysis).
- **Probability & Statistics Review:**
- ☐ Common distributions (Normal, Poisson, Exponential).
- ☐ Stochastic processes (Markov Chains).
- ☐ common statistical estimation methods: MSE, ML, MMSE
- **ECE 314 Lab Deep-Dive:**
- ☐ Complete **Correlation** lab questions.
- ☐ Complete **MMSE (Minimum Mean Square Error)** lab questions.
- ☐ Complete **PCA (Principal Component Analysis)** lab.

🧪 Phase 2: Paper Reproduction (The "Trial Run")

The goal here is to bridge the gap between "reading" a model and "building" one.

Problem C: Data & ML Focus

- ☐ **Active Reproduction:** Choose a past Problem C (Data Science/ML) paper.
- ☐ **Data Processing:** Manually clean, normalize, and handle missing values in the dataset.
- ☐ **Statistical Analysis:** Re-run the ML models described in the paper and compare your results to theirs.

Problems A & B: Modeling & Simulation

- [ ] **AI-Assisted Coding:** Use AI to co-generate a complete, readable codebase for a continuous (A) or discrete (B) problem.
- [ ] **Simulation Run:** Successfully execute the code locally or in the cloud.
- [ ] **Visualization:** Generate at least one high-quality plot (e.g., sensitivity analysis, spatial heatmaps, or time-series plots) from the simulation results.

🛠 Phase 3: Technical Workflow & Tooling

Speed is the name of the game. Don't waste time on formatting during the contest.

- **Information Retrieval (IR):**
- [ ] Conduct a simulated search for Problems A and B.
- [ ] Build a "Literature Repository" (a structured collection of 5–10 high-quality papers/datasets).
- **Visual Production:**
- [ ] Practice generating professional **Tables** (LaTeX `booktabs` style).
- [ ] Practice generating **Figures** (TikZ for diagrams or Python/Matplotlib for data).
- **The Template Test:**
- [ ] Download a standard MCM LaTeX template.
- [ ] Successfully compile it with your team's names and a "dummy" Abstract/Summary sheet.
- [ ] Try to flesh it out with at least the basic part of a canonical MCM-style paper, use this as the skeleton of your final paper. (problem restatement literature review, our work, assumption, notation)

💡 Pro-Tips for your Prep

> **On AI Collaboration:** When using AI for Problems A/B, don't just ask for "the code." Use a prompt like: *"Explain the mathematical logic of this ODE system first, then provide a modular Python script with comments for each parameter."* This ensures you actually **understand** what you are submitting.

Todo

Pack your preparation work and push to this repo (I don't want to tell YOU again https://github.com/flashfire1001/MCM-2026Season) 真的不想再重复说了. 可能你们

> 觉得这个仓库所有者是我, 有点不爽, 但是比赛完了你 fork 走不就完了吗? 反正比赛结束之后也不会维护了.
> 怎么协作? 具体的做法是, 本地 clone 一份, 每次先 pull 一下, 之后做完自己的工作 push 一下. 这样实现了多个人同时修改一个代码(资料)仓库.

最后, 为什么要把资料上传到仓库? 因为一方面监督学习, 一方面可以让我们都知道整个团队的技能 (真不想共享资料的话, 你们写在 log/里也好啊! 至少我们知道你的学习进度, 知道每个人什么水平, 有什么技能, 我们能干什么)

如果你打算上传, 要把什么上传上去? 可以是你对论文的看法, 可以是你的学习笔记, 也可以是你的学习资料. 还可以是你完成的部分工作, 包括但不限于: 拥于检索论文的网站, 用于画图的代码块, 拥有生成 latex 表格的代码块, 用于生成 mermaid/canvas 图片的代码块, refined MCM 论文 template, 请求 AI 帮助你破题的高质量 prompt……

总之, 别掉线失联, 别摸鱼摆烂, 这话对我们团队每一个人.