



VANDERBILT UNIVERSITY
College of Connected Computing

Day 0 — Setup & Running the Tools

FMAIV: Formal Methods & AI-Assisted Verification

Taylor Johnson — Associate Professor of Computer Science, Computer Engineering & Electrical Engineering; Associate Dean for Graduate Education, College of Connected Computing · taylor.johnson@vanderbilt.edu · taylortjohnson.com

Ben Wooding — Postdoctoral Scholar, Institute for Software Integrated Systems · ben.wooding@vanderbilt.edu · woodingben.com

Vanderbilt University

Setup & Tools

Day 0 — Setup

Running everything in the browser — no install

Run it live: [GitHub Codespaces](#) · [per-day Colab notebooks](#) · [full reference:](#)
`GETTING_STARTED.md`

Three ways to run — pick one, no install needed

Way	What you get	Best for
GitHub Codespaces	a full Linux box in the browser, every tool preinstalled	running <i>everything</i>
Google Colab	one notebook at a time, just a Google login	Z3, model checking, the NN demos
Browser apps	no account at all	quick one-off experiments

Local installation is fully supported as a backup — see the repo README.

GitHub Codespaces — open it

1. Sign in to GitHub (free; students get more hours via GitHub Education).
2. **github.com/ttj/fmaiv** → **Code** ► **Codespaces** ► **Create codespace on main** (or the **Open in Codespaces** badge in the README).
3. Wait ~1–2 min the first time; a banner lists the ready tools.

Preinstalled: `z3` · `NuSMV` · `lean` / `lake` · `cbmc` · `cryptol` · `saw` · `smvis`, plus `torch` / `auto_LiRPA` for the Day-4 neural-network notebooks.

The layout & the terminal

- **Left:** file explorer (`day01/` ... `day04/` , `notebooks/`).
- **Middle:** the editor.
- **Bottom:** the **terminal** — where you type commands (**Terminal ▶ New Terminal**, or `Ctrl+``).

```
pwd                # where am I? (the repo root)
ls day01/examples  # list a folder
python3 day01/examples/z3_smoke.py  # run a file
```

Day 1 — SAT/SMT with Z3

```
python3 day01/examples/z3_smoke.py           # first sat/unsat query
python3 day01/examples/z3_pigeonhole.py      # an UNSAT proof
python3 day01/examples/z3_entailment.py      # prove by refuting the negation
python3 day01/examples/puzzles/sudoku.py     # also: nqueens, magic_square, kenken
z3 day01/examples/z3_smt_basics.smt2        # the SMT-LIB text format
```

Each puzzle ships a `*_starter.py` to complete yourself.

Day 2 — Model checking with NuSMV

Command is **NuSMV** (capital letters). The slides say *nuXmv*; the open **NuSMV** is the drop-in — same language, same verdicts.

```
NuSMV day02/examples/counter.smv          # "-- specification ... is true/false"  
NuSMV -bmc -bmc_length 12 day02/examples/bmc_depth.smv
```

Interactive console: **NuSMV -int**, then **read_model -i <file>** → **go** → **check_property** → **pick_state -r** → **simulate -r -k 5** → **show_traces** → **quit**.

Day 3 — Theorem proving with Lean 4

```
cd day03/examples/CounterDemo
lake build                # checks every proof; green = all pass
lake build CounterDemo.Counter # one module
cd -
```

Open any `.lean` file in the editor to see the **proof state live** (Lean 4 extension preinstalled). Mathlib-free → builds in seconds.

Day 4 — Programs: CBMC, Cryptol, SAW

```
cd day04/examples
cbmc counter.c counter_check.c --unwind 26 --unwinding-assertions # VERIFICATION SUCCESSFUL
cryptol -c ":prove bounded_invariant" counter.cry # Q.E.D.
clang -c -emit-llvm -O0 -o popcount.bc popcount.c && saw popcount.saw # Proof succeeded!
cd -
```

Day 4 — Frontier: neural-network robustness

Two notebooks (they use PyTorch). In the Codespace, open and **Run All**:

- `notebooks/05_day4_nn_robustness.ipynb` — a 2-D toy: certify a robustness margin, see the decision boundary.
- `notebooks/06_day4_nn_mnist.ipynb` — multi-class MNIST: IBP vs CROWN vs α -CROWN.

CPU-only; the script form is `python3 day04/examples/nn/robustness.py`.

The notebooks — one per day

In the Codespace: click a notebook in `notebooks/`, then **Run All** (or Shift+Enter per cell). Or run `jupyter lab` for a full tab.

<code>01_day1_logic_sat_smt</code>	Z3
<code>02_day2_model_checking</code>	NuSMV
<code>03_day3_theorem_proving</code>	Lean
<code>04_day4_program_verif</code>	CBMC · Cryptol · SAW
<code>05/06_day4_nn_*</code>	NN robustness · MNIST

No Codespace? Colab & browser apps

- **Colab** (just a Google login): each notebook has an **Open in Colab** badge; the first cell installs what's missing. Table in [notebooks/README](#).
- **Z3 in the browser**: <https://microsoft.github.io/z3guide/>
- **smvis** (NuSMV + state/BDD pictures): https://bit.ly/fmaiv_smvis
- **Lean web editor**: <https://live.lean-lang.org/> · Natural Number Game

Bring an AI assistant + where to go next

- **AI coding assistant** (we use it throughout): install **Claude Code**, **OpenAI Codex**, or **Gemini Code Assist** from the Extensions panel (`Ctrl+Shift+X`).
- **Capstone:** `capstone/` — one property, all four tools.
- **Full reference:** `GETTING_STARTED.md` — every command here, plus troubleshooting.

Cheat sheet: `pwd` / `ls` to look around · `NuSMV` is capitalized · `cd -` returns to the repo root · `help` inside a tool's console.