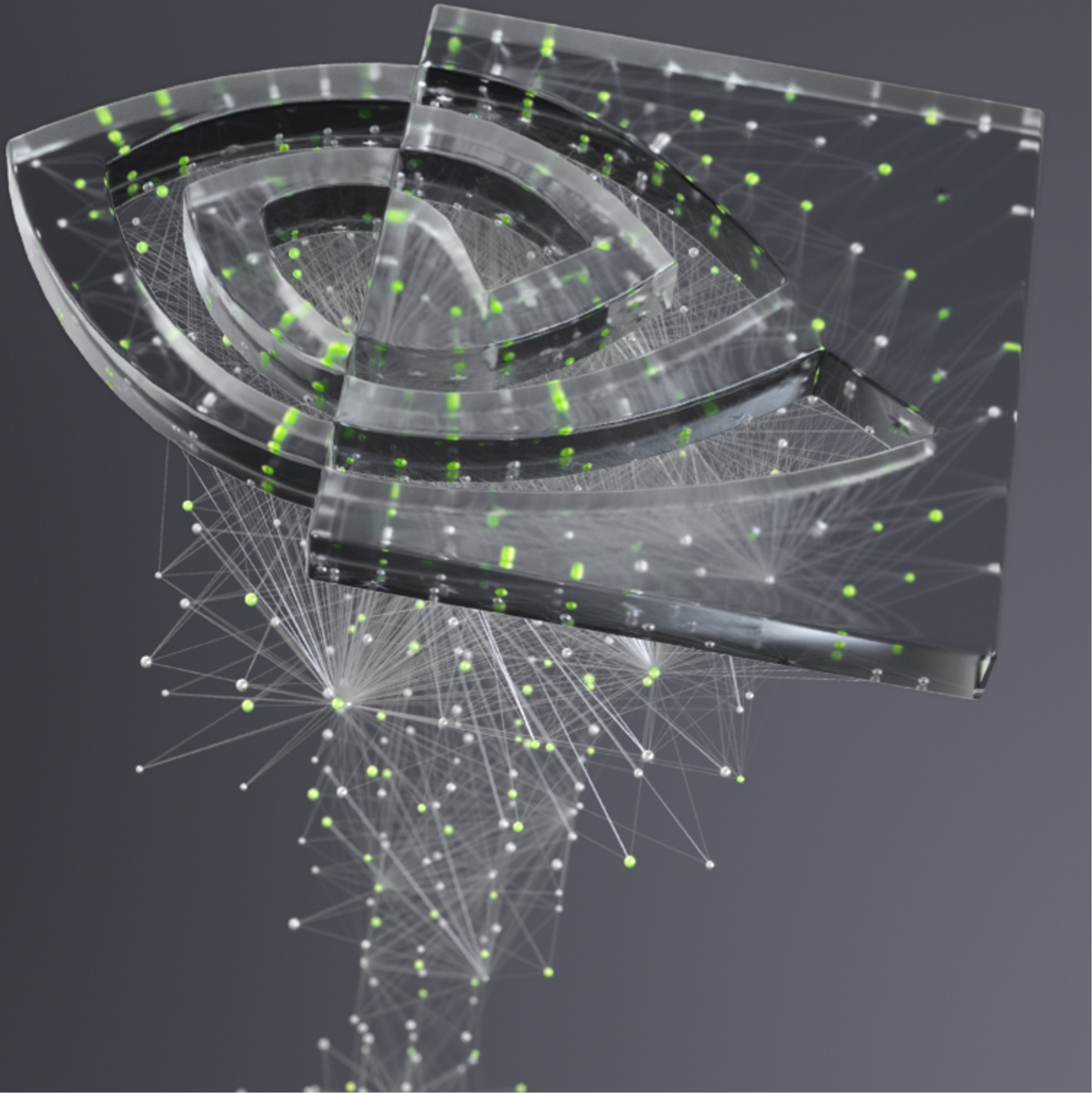
NVIDIA



# GPU BOOTCAMP

## MINI CHALLENGE

# HINTS

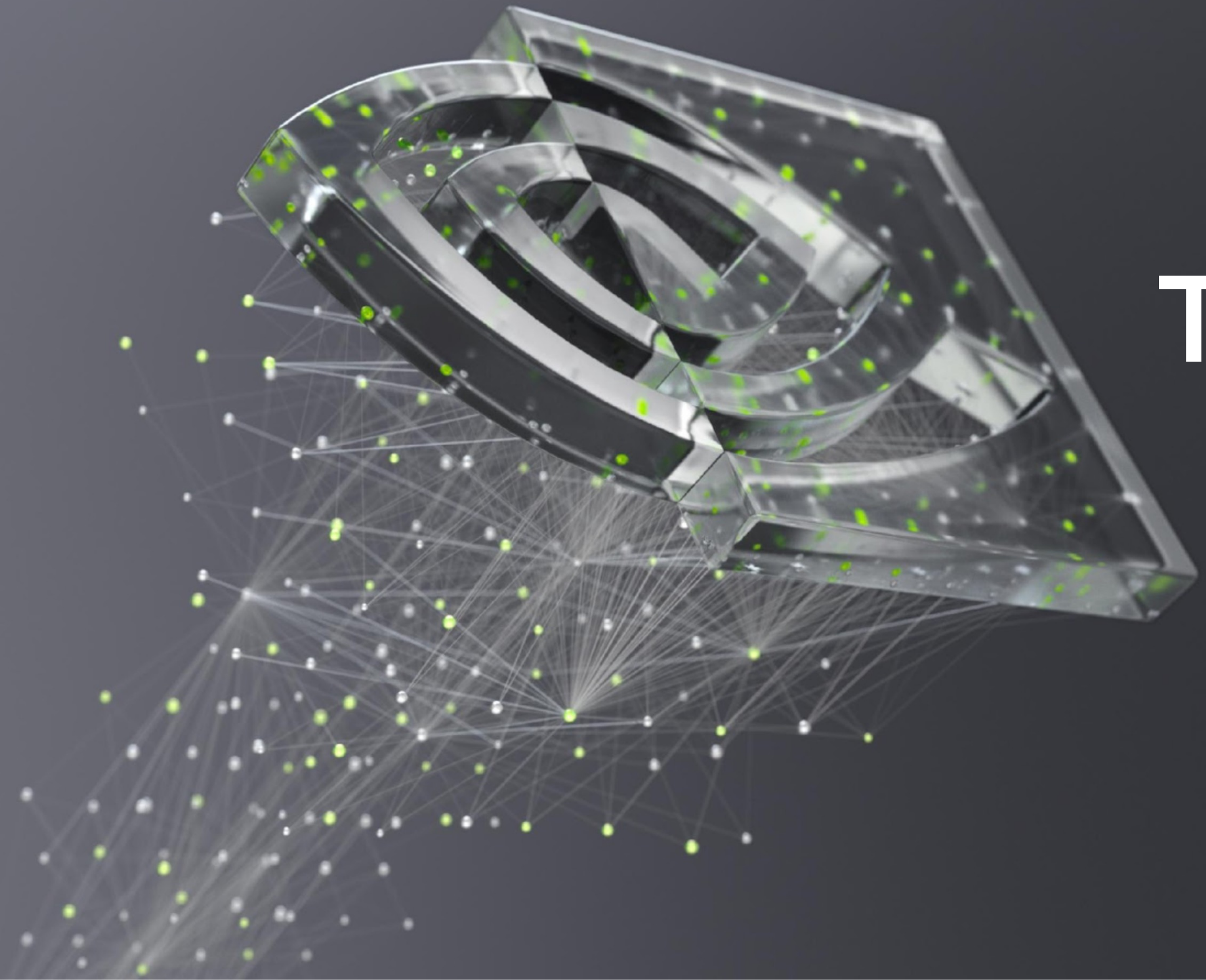
- Make use of compiler flag to cross check if indeed parallelization was done e.g. –Minfo
- Key functions to look out having loops: (THREAD ME)
    - semi_discrete_step
    - compute_tendencies_x
    - compute_tendencies_z
- Figure out the variable in the for loop should be private, atomic, or shared
- Download and take backup

# STEPS

- Set sim_time and out_freq to be 10
- Add nvtxRangePushA() and nvtxRangePop to profile the program
- Run sequential version and save the reference data (reference.nc)
- Modify the Makefile
  - STD ISO: -stdpar=gpu (remember to include algorithm, execution, and thrust libraries)
  - OPENACC: -acc –ta=tesla,managed,lineinfo (unified memory)
  - CUDA: nvcc / nvfortran –cuda (remember to include cuda_runtime library)
- Do parallelization and optimization
- Make sure the data correctness

# CHALLENGE

- NWAYs challenge
  - Implement STD / OPENACC / CUDA to six for-loops
- Performance challenge
  - Speed up miniweather with
    (nx_glob, nz_glob, sim_time) = (4000, 2000, 10) and (8000, 8000, 10)



# THANK YOU

NVIDIA.