IHPCSS 2019
HYBRID CHALLENGE

RIKEN R-CCS RIKEN Center for Computational Science

PRACE

SciNet
ADVANCED RESEARCH COMPUTING at the UNIVERSITY OF TORONTO

XSEDE
Extreme Science and Engineering Discovery Environment

WHAT IS IT

Haven't heard of this challenge yet? This is for you…

THE CHALLENGE

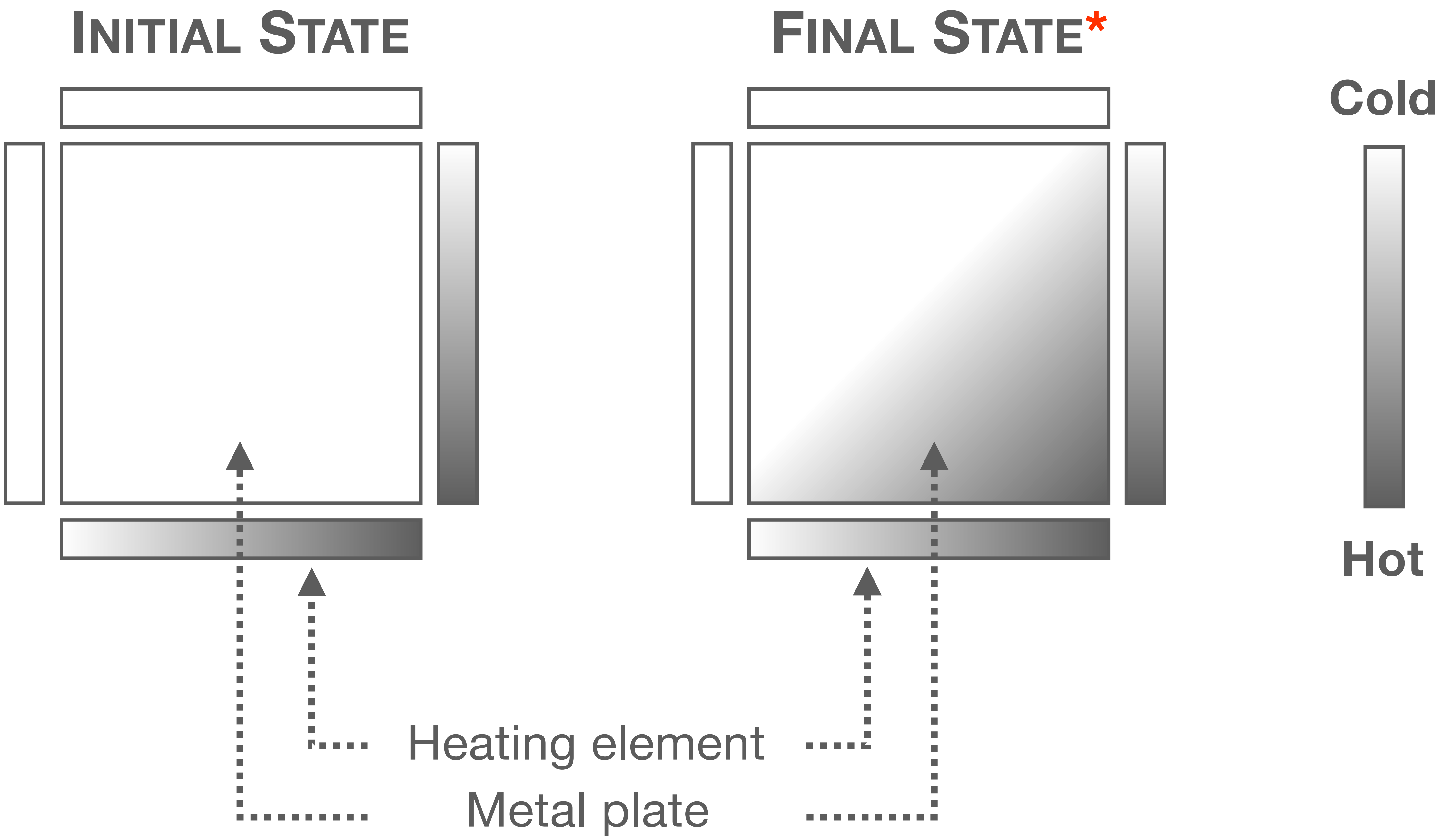
Simple problem

Simple aim

Practice what you’ve learned

Relaxed atmosphere

SIMPLE PROBLEM – STEADY STATE HEAT EQUATION



Final state is reached when the variation of temperature on the metal plate between iteration n and iteration n+1 is less than a certain threshold, indicating that convergence has been reached.

SIMPLE AIM - BE AS FAST AS YOU CAN

Challenge Setup

Size of metal plate: 14560 x 14560

To reach convergence: 3586 iterations

Your Mission

You are given 4 nodes

How fast can you run it to convergence?

PRACTICE WHAT YOU'VE LEARNED

Asynchronous Livelock

OpenMP C Create MPI

Synchronous

Multithreading Data movement

Non-blocking

GPU Overlapping OpenACC

Hybrid FORTRAN Communicator

Pragma Deadlock Reduction

Kernel Slurm Halo swap Data dependence

Copy CPU Directives

RELAXED ATMOSPHERE

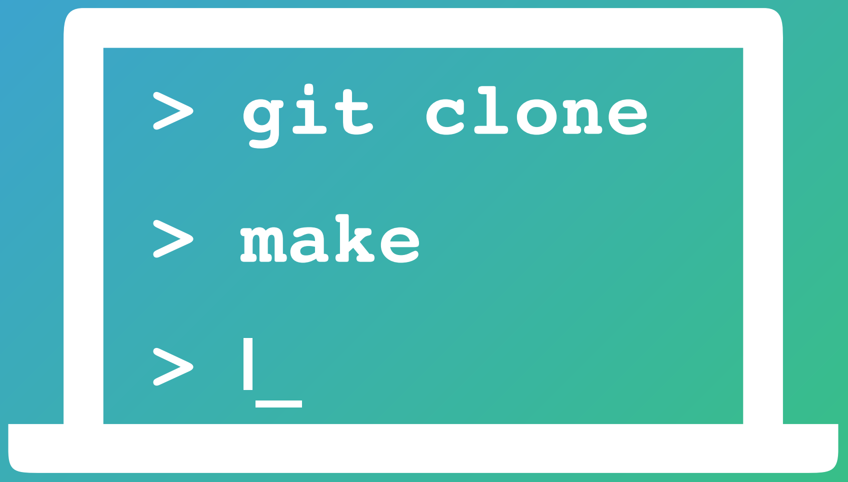
✓ No obligation to participate

✓ No registration required

✓ You are free to submit or not at the end

HOW TO GET STARTED

You take part? This is how you get ready



```
> git clone
> make
> |_
```

HOW TO GET STARTED – THE BIG PICTURE

1 Clone the repository

2 Make

3 Run locally

4 Run on compute nodes

5 Verify

HOW TO GET STARTED - CONCRETELY, ON BRIDGES

1) Get the repository (serial, OpenMP, MPI, OpenACC etc… source codes)

```
> git clone https://github.com/capellil/IHPCSS_Coding_challenge.git
```

2) Compile all source codes

```
> module load cuda/9.2 mpi/pgi_openmpi/19.4-nongpu && make
```

3) Run locally (if no output_file given, it prints to console)

```
> ./run.sh <language> <technology> <size> [output_file]
```

4) Submit job to bridges nodes (output_file is mandatory)

```
> ./submit.sh <language> <technology> <size> <output_file>
```

5) Verify an output file

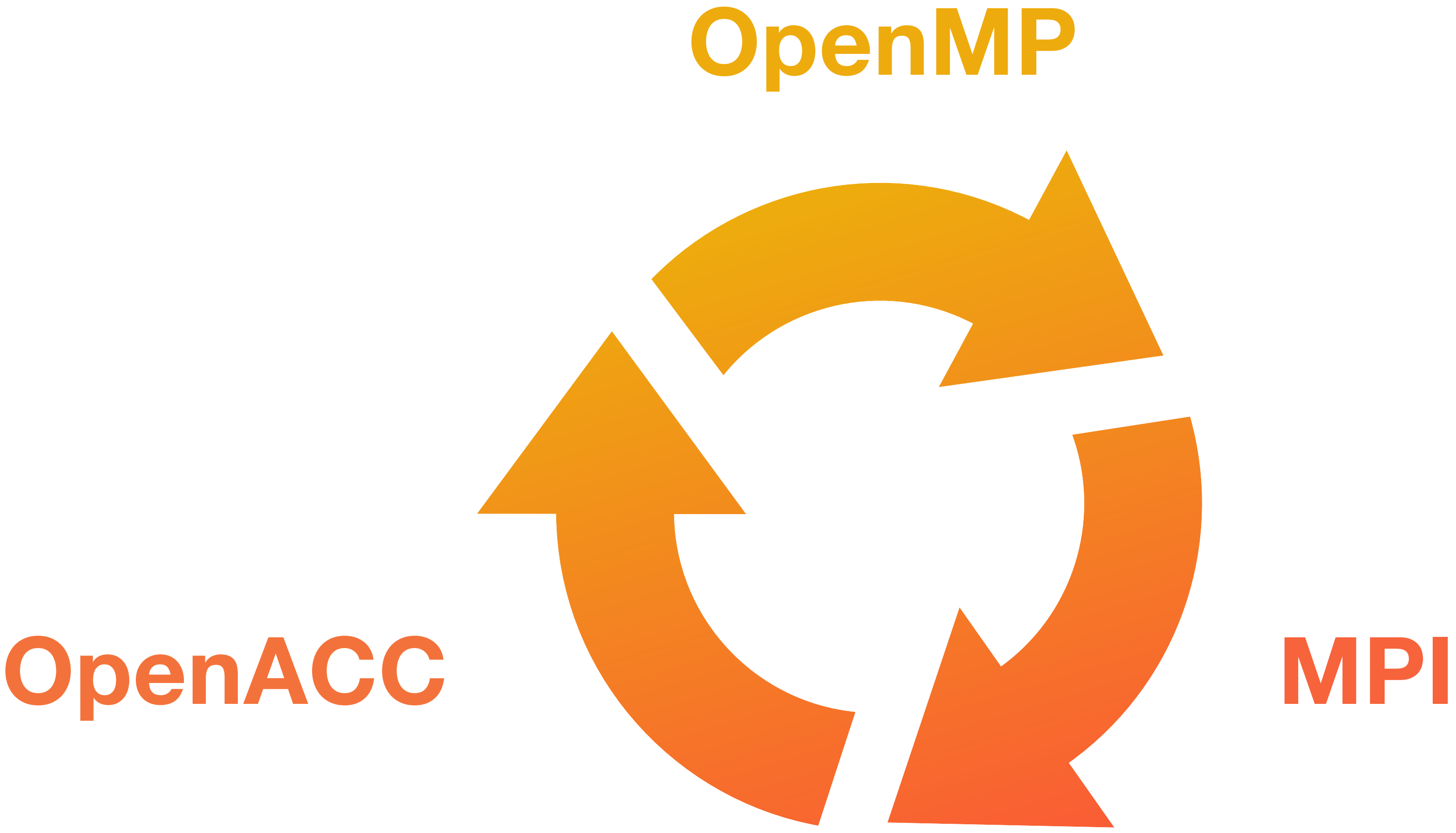
```
> ./verify.sh <output_file>
```

```
<langage> = C | FORTRAN
<technology> = serial | openmp | mpi | openacc | hybrid_cpu | hybrid_gpu
<size> = small | big
```

COMPETITION SETUP

Could be useful information…

COMPETITION SETUP – THE TECHNOLOGIES



Note

Without MPI, you have a single-node solution. Are you allowed to do it? Absolutely! But, how likely are you to be faster than someone using all 4 nodes…

COMPETITION SETUP – THE CATEGORIES

Fastest CPU code

Fastest GPU code

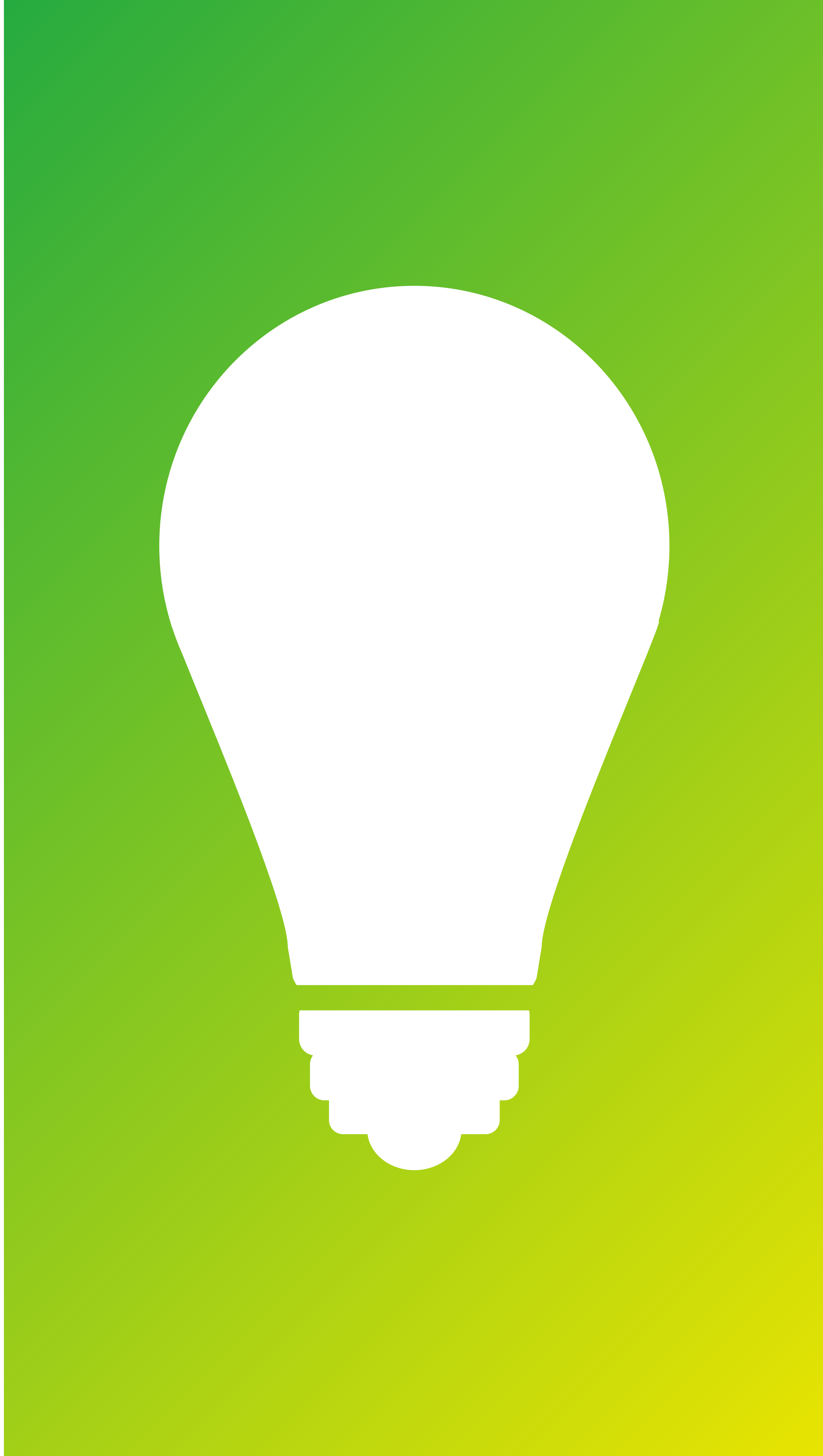
COMPETITION SETUP - CONDITIONS FOR A VALID SUBMISSION

Team of 1, 2 or 3 members

Submission deadline
Friday 12th of July, at noon

LAST TIPS

Almost ready to go…



LAST TIPS

Need more information?

The GitHub README is your friend!

https://www.github.com/capellil/IHPCSS_Coding_challenge

Still have questions?

- l.capelli@ed.ac.uk
- tanujkr.aasawat@riken.jp

GOOD LUCK & ENJOY

1) Yes, the fastest GPU team will really win this trophy

2) Yes, the fastest CPU team will win an identical one