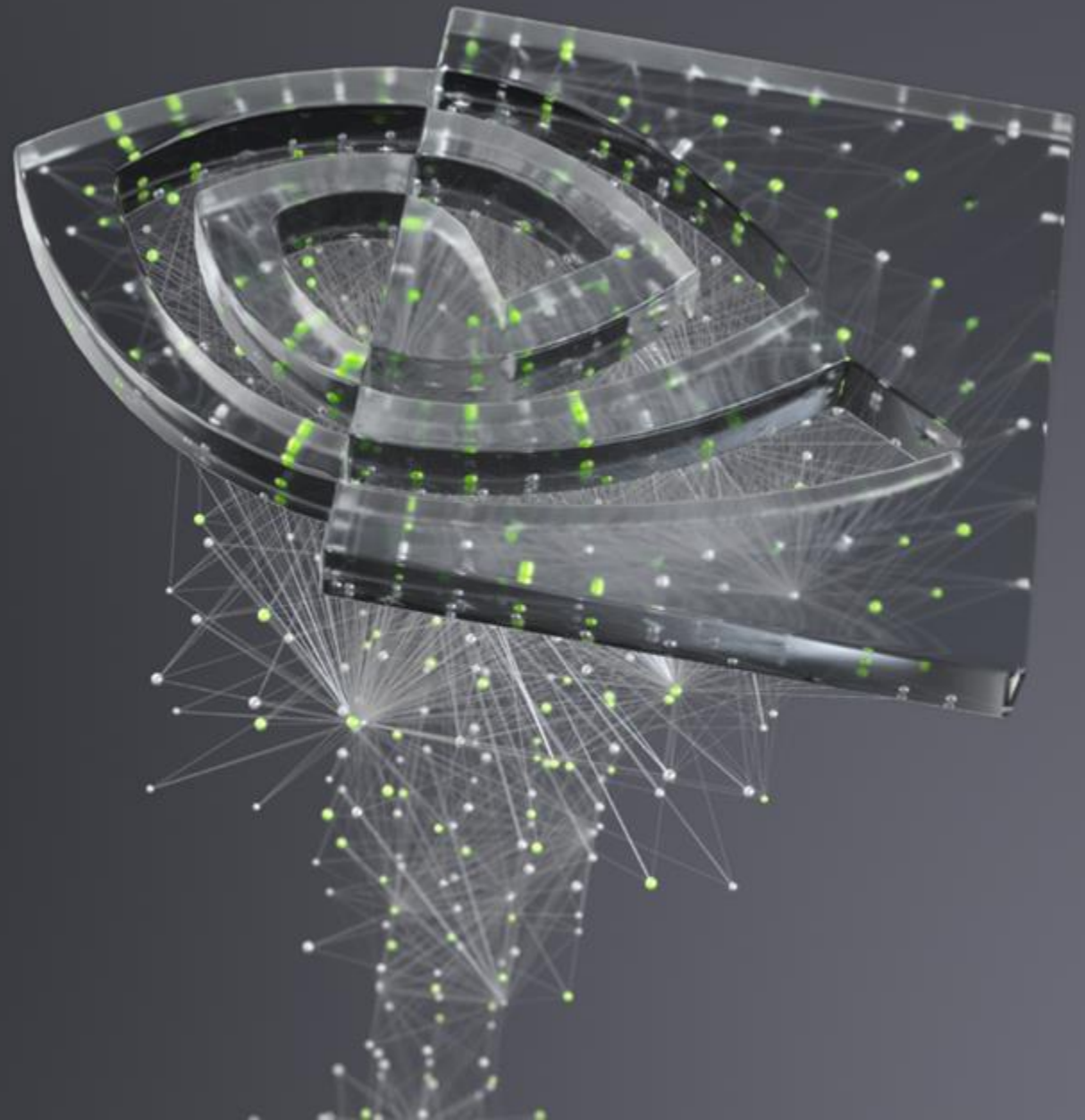


NVIDIA

GPU BOOTCAMP

MINI CHALLENGE

APPLICATION

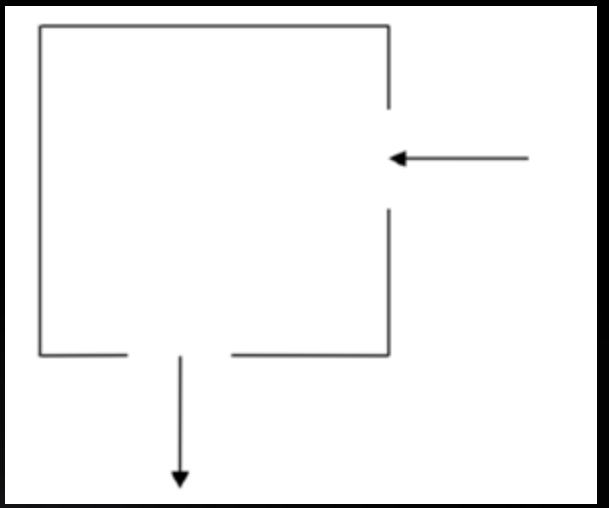
Simple 2D regular-grid CFD simulation

Simulation of an incompressible fluid flowing in a cavity using the 2D Navier-Stokes equation

```
set the boundary values for Ψ
while (convergence == FALSE)  do
    for each interior grid point do
        update Ψ by averaging with its 4 nearest neighbours
    end do

    check for convergence
end do

for each interior grid point do
    calculate ux calculate uy
end do
```



The objective of this exercise is not to dwell into the Maths part of it but to make use of different approaches to GPU programming to parallelize and improve the performance.

CFD
Pseudo Code

```
int main(int argc, char **argv) {
        initialization loop
        boundary calculation loop
        Jacobi loop
        swap array loop
}
```

- cfd.cpp
- boundary.cpp
- jacobi.cpp

MORE ABOUT CODE

- Uses Makefile
- To run the code ./cfd 64 500
  - Where ./cfd is application name
  - 64 is size of scaling
  - 500 is number of max iteration

Output:

... finished

After 500 iterations, the error is 0.00211211 ▫ Check this value to confirm your porting

Time for 500 iterations was 18.8579 seconds

Each iteration took 0.0377159 seconds

HINTS

- Divide different methods to port among team members
- Use profiler to check the hotspots and bottlenecks in your code
- Make use of compiler flag to cross check if indeed parallelization was done e.g. –Minfo
- Key files to look out having maximum loops:
  - cfd.cpp
  - jacobi.cpp
- Download and take backup


THANK YOU
NVIDIA.