Major Project: Milestone 0 (Topic Selection)

Weighting : 0%

1) Create a simple C/C++ program that simply prints the title of your project topics, e.g. using std::cout<<

2) Submit that program as a slurm job on one of the clusters (doesn't matter which one). [Tutorial 01 will help]

3) Upload the slurm log (.out file) as your 'Milestone 0'. This log will simply show that your job ran, and that it printed out the name of your project.

This is not a real assessment item, this is just to ensure everyone is on track and has chosen a topic. It will also get you acquainted with the basics of creating a simple C/C++ program and submitting slurm jobs.

Here is a list of possible project topics. These topics are somewhat general but there is scope for variations, especially if you have implemented the basic model easily and are keen to do something more advanced/specific.
It is also possible to propose your own topic, but you will probably need to supply the background knowledge and will need to take on greater responsibility for making sure the topic isn't too large or complex. Talk to the lecturers if you are unsure.
A good source of ideas for project topics:

- Netlogo (netlogoweb.org)

NB - To maximise time for working on the projects we want people to decide on their topic by the end of Week 2.

Project Name	Description
Epidemic model	Implement agent-based simulation to approximate epidemic/SIR differential equation models (e.g. from SCIE1000). See Sample Models->Biology->Virus in Netlogo for example.
Predator/prey model	Implement an agent-based simulation to approximate predator/prey dynamics from differential equation models (e.g. from SCIE1000). See Sample Models->Biology->Wolf Sheep Predation in Netlogo for example.
Flocking/swarming	Simulate flocking or swarming behaviour. See Sample Models->Biology->Flocking in Netlogo for example.
Cellular Automata	Implement a cellular automata. See Sample Models->Computer Science->Cellular Automata->Life for example.
Partial Differential Equations	Implement a PDE solver on an N-dimensional grid.
Interacting particles	Implement a model of interacting particles. This could be some kind of liquid or gas flow, or perhaps gravitational interactions and simulating the evolution of a galaxy (or the entire universe).
Eigenvalue solver	Implement a parallel eigensolver for large sparse matrices. [Hard]