What is GPU Hackathon

As part of the workshop every year, KSL facilitates a coding event in which teams of developers port their own applications to run on GPUs, or optimize their existing GPU applications. Each team usually consists of 3 to 5 developers who are intimately familiar with the application and workflow. They work with one or more GPU programming experts mentoring these sessions. Our mentors come from NVIDIA and KSL supercomputing applications team.

Hackathon will span over Day 3 to Day 4. There are a variety of ways to port "suitable" codes to GPUs. They range from calling GPU enabled libraries in your code to refactoring using a parallel programming model for GPUs, e.g. introducing OpenACC, OpenMP for offloading or even writing a CUDA version. Also possible is to port parts of a Pythonic workloads conducive to run on GPUs can make use of packages like Numba and CuPy. We will, however concentrate on using OpenACC to accelerate applications written in C/C++ or Fortran.

Why participate

- Explore if your code can run or optimize on GPUs to achieve faster time to solution
- Allow yourself to set aside time for training and coding sprint, in the presence of experts for face to face consulting on your code
- By the end of the hackathon, each team will ideally have ported some part of their code to run on GPUs or have a clear roadmap of how to/what to do next.

How to apply/participate

In order to participate, you must be developers of the code. You will need to write a short description to introduce your application and project. This may include:

- Background: about your team, your application/code, science/industry problem you are addressing, etc.
- A brief status of your code: programming language, scalability, HPC use vs cloud, GPU use etc.
- What is the motivation to speeding up your code? How will it impact your project's goals?
- Submit your proposal

How to Prepare:

- Given that your proposal is accepted, you will be invited to share code, preferably as a repository hosted in e.g. GitHub, GitLab, Bitbucket, SVN etc.
- If your code is more than 10,000 lines, please pick a few representative kernels to start working on during this event. You may want to include most compute intensive parts of your code in this benchmark.

- Test a representative case with valid answers before coming to the event. It is important because the workflow will include iterations of optimizations and testing both for validity and for performance.

Send you application to training@kaust.edu.sa

Deadlines:

Submission: April 3rd, 2020

Invitation to attend: April 8th, 2020

- **Mentor must be able to access to code repository: April 14th, 2020**