## Where we are

Time	Details	Type
8:30 - 8:35	Introduction, welcome and objectives	Presentation
8:35 - 8:55	An overview of MLIR and LLVM	Presentation
8:55 - 9:20	The xDSL framework	Presentation
9:20 - 9:25	Introduction to the hands-on activity	Presentation
9:25 - 10:00	Logging into ARCHER2 and hands-on practical activity	Hands-on
10:00 - 10:30	Morning break	
10:30 - 10:35	Welcome back and overview of second part	Presentation
10:35 - 11:45	Hands-on practical activity	Hands-on
11:45 - 11:55	Wash up from practical activities, highlighting key take-away points	Presentation
11:55 - 12:00	Conclusions & next steps to continue working with the technologies	Presentation





#### Hands on activities

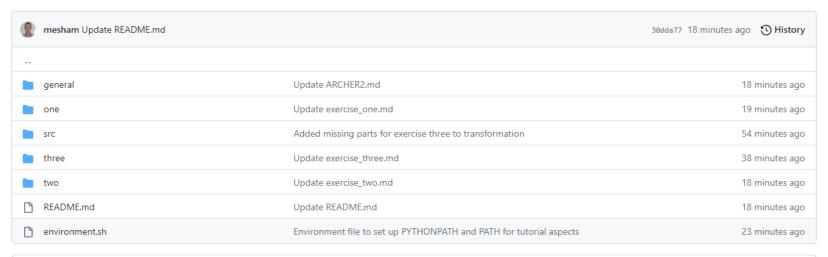
- We are going to develop a (very simple) Python compiler, that takes Python code and compiles this to an executable that will run on the ARCHER2 supercomputer (a Cray EX)
  - We will add some threaded parallelism and vectorisation in also as we progress
- The purpose of these exercises is to walk you through using MLIR, LLVM, and xDSL end-to-end, and to firm up the ideas already discussed in the lectures
  - Dialects and how to define our own operations
  - How to apply transformations
  - Development of our own transformations
  - Exploration of the IR
  - Interfacing with MLIR and the options we have there





### Practical material

README.md



0

#### xDSL introduction tutorial practicals

These are the practical exercises for the xDSL introduction tutorial, where we have the following directories

- one is where participants will obtain an overview of building the IR, compiling, and executing for a simple Hello World Python example.
- two is where we get more in-depth into the details of the dialects and transformations as we add support for the Python For construct, supporting loops in our simple Python compiler using the scf.for operation.
- three is where we leverage threaded parallelism via OpenMP and vectorisation by transforming our for loop into an *scf.parallel* operation and then use existing MLIR transformations to lower to the *omp* or *vector* dialects.
- src contains the source code (dialect, transformations, and tinypy\_opt tool) that will be used throughout these exercises. If you are participating in one of our organised tutorials then this will all be preinstalled for you to use.

https://github.com/xdslproject/training-intro/tree/main/practical

# First step....

- Logging into ARCHER2
  - Each of you has a guest account, the presenters will hand these out to you now
  - You will need to SSH into login.archer2.ac.uk from your laptop
- Details at
  - https://github.com/xdslproject/training-intro/blob/main/practical/general/ARCHER2.md
- Practical one at
  - https://github.com/xdslproject/training-intro/tree/main/practical/one



