Project description:
We are planning to build a FFNN and train that network using backpropagation. We will be inputting data taken from monthly S&P stock data to receive multiple outputs which will be various price predictions based on the input data. We will run the FFNN on GPU. The inputs will be high price, low price, open price, close price, and adjusted close. Outputs whether the price increases or decreases. There will be 3 layers, which include input, hidden, and output layers. The hidden lay will have 5 neurons.

Planned tasks:
- Build FFNN, backpropagation in GPU
- Training models
- Testing models/prediction accuracy

Expected outcomes (what do you hope to deliver by the project deadline?):
We are hoping to implement a FFNN using basic math CUDA libraries for both forward pass for prediction and backpropagation for updating the weights and biases. If time permits we are hoping to implement a base neural network with parameters for more layers and neurons.

tools/libraries/languages:
CUDA
- CUBLAS
- cuTensor
- cuRAND
- CUDA Math Library
- NVIDIA TensorRT

Task Division:
We're planning to both work on all parts of the project equally.

We can update this progress report as we go along this project.