Q1

Build a multi-layer neural network to solve the XOR classification problem. Use the provided `FFNN` as a starting point. Test against the data generated by `gen_xor()` from `gen_data.py`. **Show:**
a) the testing percent correct;
b) the training progress curves; and
c) the decision surface with overload training data.

Q2

Implement a feed-forward neural network function *from scratch*, using only built-in Python modules and numpy. Extract the learned weights from Q1 and run the model through your custom implementation. **Demonstrate that you get the same results.**

Do not train the model yourself. Do not implement backpropagation. Just run it forward using the PyTorch-trained weights.

You may work in a group of 1 or 2. Submissions will be graded without regard for the group size. You should turn in a document (`.txt`, `.md`, or `.pdf`) answering all of the red items above. You should also turn in Python scripts (`.py`) for *each* of the blue items.