Problem
For this assignment, a sequence-to-sequence model was trained that translated natural language questions into a logical form. The models are trained on the Geoquery dataset and utilized LSTMs as the network cells.

Part 1: Basic Encoder-Decoder
The basic encoder-decoder model requires roughly 70% token accuracy and 10% denotation accuracy, which the model is able to achieve. This model can be run with the command “python main.py”. The output is shown below:

```
Exact logical form matches: 16 / 120 = 0.133
Token-level accuracy: 2982 / 3908 = 0.763
Denotation matches: 19 / 120 = 0.158
```

When implementing the decoder module, I used the pytorch Embedding and LSTM module for the network cells, and a linear and softmax layer to produce a decoder output. The training loop loops through the epochs and the training samples individually, calls the encoder, repeatedly calls the decoder, and accumulates error terms before optimizing. Finally, the inference loop calls the encoder, then repeatedly calls the decoder until the EOS token is generated.

After implementing the model components, the training and inference loops, I tried many hyperparameters to tune the model performance. My final model uses a hidden and embedding size of 200, and optimizes with the Adam optimizer. It also uses teacher forcing to help the model converge faster.

This model was not able to effectively learn the dataset, often predicting correct logical forms with the wrong constants. It also generates many syntax and executor errors. The system could be improved by using a larger model, training models for longer, or implementing the attention mechanism done in part 2.

Part 2: Attention
After 10 epochs, the attention model reaches 76.3% token and 46.7% denotation accuracy. This model can be run with the command “python main.py --attention”. The output is shown below:

```
Exact logical form matches: 47 / 120 = 0.392
Token-level accuracy: 2981 / 3908 = 0.763
Denotation matches: 56 / 120 = 0.467
```

After 30 epochs, the attention model reaches 77.6% token and 50% denotation accuracy. The default number of epochs is 10, so the 30 epoch model can be trained with the command “python main.py --attention --epochs 30”. The output is shown below:

```
Exact logical form matches: 54 / 120 = 0.450
Token-level accuracy: 3034 / 3908 = 0.776
Denotation matches: 60 / 120 = 0.500
```

The implementation of the attention model required a new decoder model. In addition to the previous embedder, LSTM, linear, and softmax layers, a new linear layer is used to learn the

attention weights. The forward method of this model had to take two new parameters - the previous context and the encoder outputs. The implementation of attention is general, and is incorporated using the previous state/context. Once incorporated, the outputs are passed to the final linear and softmax layer as previously.

This model provided large improvements over the previous non-attention models. Many of the constants are now correct in the logical forms, however there are still some syntax and executor errors being predicted. The system could further be improved by encoding the sentences in a higher dimension, or training the model for longer.