Answer the following questions

- **What algorithm?**
  - Deep Q Network, a model free RL algorithm.
  - Learns the value in a particular state.
  - Specific variant: involves experience replay and a deep CNN.
- **What sort of data structures and classes will we need?**
  - Input:
    - Take maximum value for each pixel color value over the frame being encoded and the previous frame.
    - Extract Y channel, that is Luminance, from RGB frame to rescale to 84 x 84.
  - DQN class, for Deep Q network.
  - ReplayMemory class, memory representation for agent.
  - Agent Class, performs, remembers, and learns actions.
  - Create a separate py file for the environment.
  - Create a separate py file for settings.
- **What model architecture?**
  - Architecture:
    - Input: 84 x 84 x 4 image produced by pre-processing map psi.
    - First hidden layer – convolutional, 32 filters of 8 x 8, with stride 4. Rectifier nonlinearity.
    - Second hidden layer – convolutional, 64 filters of 4 x 4, stride 2. Rectifier nonlinearity.
    - Third hidden layer – convolutional, 64 filters of 3 x 3. Stride 1. Rectifier nonlinearity.
    - Final hidden layer – fully connected. 512 rectifier units.
    - Output layer - fully connected. Single output for each valid action. Valid actions varied between 4 and 18.
  - Training Details
    - 49 Atari games.
    - A different network for each game. However, the architecture was not changed.
    - Reward clipping.
      - Positive rewards clipped at 1.
      - Negative rewards clipped at -1.
      - 0 rewards unchanged.
    - If there is a live counter, Atari emulator sends the number of lives at the end of the game. This signal was used to mark the end of the episode during training.
- **What are the hyper parameters:**
  - Mini batch size 32
  - Replay memory size 1,000,000
    - 50,000 frames takes up about 17GB or RAM. Scale accordingly. I have 32 GB, but I only want to use up to 16 GB
  - Agent history length 4
  - Target network update frequency 10,000

    - Discount factor 0.99
    - Action repeat 4
    - Update frequency 4
    - Learning rate 0.00025
    - Gradient momentum 0.95
    - squared gradient momentum 0.95
    - min squared gradient 0.01
    - initial exploration 1
    - final exploration 1,000,000
    - no-op max 30
- What general results will we get?
    - Trained on 49 different Atari games.
    - Expect a score of 75% of what a human will score on 50% of the games.
    - Expect a score better than any other RL algorithm can achieve.