1. You have a large combination of categorical features. Your friend suggests that you should give each possible combination of features a number and use the binary representation as a vector instead of the one hot encoding in order to reduce the number of features, and claims that it will not affect the performance of the tree.
   a. You should disagree because this will not allow the decision trees to get the same accuracy as a one hot encoding.
   b. **You should disagree because this will make the decision tree unnecessarily deep because you have to go through all combinations.**
   c. You should agree because if you use one hot encoding it would take too many steps to check each feature as it’s own node in the decision tree, making your tree significantly deeper.
   d. You should agree because it would take too much time to calculate the entropy for each value in the one hot encoding.

2. Your friends trained a random forest algorithm and are always ending up with the same results from each tree for every new input. You want to check that the problem is not that easy for their algorithm. Which steps will you check that they did correctly first?
   Check that resampling with replacement is done correctly and check that sampling features is done correctly at each node.

3. True or false? Decision tree algorithms are more negatively affected than neural nets if the data used for training is dissimilar to what is used in testing.
   True. Decision trees are more sensitive to the training data, and they are not as good at extrapolation, and incapable of transfer learning.

4. Your friends are training a random forest algorithm to diagnose some medical conditions based on blood work. They are complaining that their random forest algorithm is very bad at recognizing some rare cases that occur to some patients. What do you recommend?
   **Using boosting.**
5. True or false? When training a Decision Tree algorithm similarly to training a neural net it is good practice to see if you can overfit on a small portion of the dataset first before you start training on the entire dataset to save time spent on training.

   False, for most cases you can train a decision tree algorithm on the entire dataset in much less time than a neural network.

6. Which of the following is the best feature to pick at a node in a decision tree algorithm?
   a. Feature 1: splits the dataset into two equal parts. The first part is all false examples and the second part is half true labels and half false labels.
      Entropy before: 0.81, entropy after: 0.5
   b. Feature 2: splits the dataset in a 70/30 split. The 70% split is 80% true labels and the 30% split is 75% false labels.
      Entropy before = 0.95, entropy after = 0.7*0.72 + 0.3*0.81 = 0.747
   c. Feature 3: Splits the dataset in a 90/10 split. The 90% split is 70% true labels and the 10% split is 100% false labels.
      Entropy before = 0.95, entropy after = 0.9*0.88 + 0.1*0 = 0.792