The data named Heart Disease is extracted from the UCI collection. It aims to be a binary classifier for predicting whether that particular person has heart disease or not. This dataset contains 14 features.

Column Descriptions:

1. `id` (Unique id for each patient)
2. `age` (Age of the patient in years)
3. `origin` (place of study)
4. `sex` (Male/Female)
5. `cp` chest pain type ([typical angina, atypical angina, non-anginal, asymptomatic])
6. `trestbps` resting blood pressure (resting blood pressure (in mm Hg on admission to the hospital))
7. `chol` (serum cholesterol in mg/dl)
8. `fbs` (if fasting blood sugar > 120 mg/dl)
9. `restecg` (resting electrocardiographic results)
   -- Values: [normal, stt abnormality, lv hypertrophy]
10. `thalach`: maximum heart rate achieved
11. `exang`: exercise-induced angina (True/ False)
12. `oldpeak`: ST depression induced by exercise relative to rest
13. `slope`: the slope of the peak exercise ST segment
14. `ca`: number of major vessels (0-3) colored by fluoroscopy
15. `thal`: [normal; fixed defect; reversible defect]
16. `num`: the predicted attribute

- **Attention:**
  <u>You can use this line of code and turn the problem to binary classification</u>

```
data['num']=np.where(data['num']>0,1,0)
```

1. Build a decision tree with the ID3 algorithm and the given train data set. Calculate the classification accuracy on train and test sets.

    a) Randomly select 45% of the training data and train the tree with it. Then test it on the entire test data and finally report the accuracy of the classification on the training and test data along with the size of the tree. Repeat this process of randomly dividing the data for the training process 3 times and determine the accuracy values each time, and finally get the average accuracy.

    b) Measuring effect of the training data size: In addition to randomly selecting 45% of the training data that you did in the previous step, also randomly select the values of 55%, 65%, 75% of the training data and get exactly the same results as you asked in the previous step. Get the result by running the program 3 different times.
    Finally, use the entire training data and report the results. In the report, discuss what effect the amount of training data had on the accuracy of the classification on the test data and the size of the decision tree.

2. Performing post-pruning and reducing the number of tree nodes: tree pruning should be done using the pruning reduced error method on the data in the following order:

a) Randomly select 75% of the training data as training data and the remaining 25% as validation data. Then build a decision tree with the help of training data with ID3 and perform pruning with validation data. Draw the loss curve of classification on all 3 sets of validation, train and test for different number of tree nodes.

b) This time, perform K-fold cross validation with K=4 and select the training and validation data accordingly, then build the decision tree with the help of the training data using the ID3 algorithm and perform pruning with the validation data. Finally, the average result on all 4 systems is considered. Draw the classification loss curve on all 3 validation, train and test sets for different number of tree nodes in each state and also the average of 4 states for training and test sets.

Extra Credit: [50 Bonus points]

The implementation of this exercise without the decision tree libraries or github repositories (implementation from scratch) includes 50% additional marks.