**Data pre processing:**

* Using excel, I created a testing data set with 100 instances of the given data sample. The instances were selected at random. The remaining 900 instances of the given data sample were used as my training data set.
* I eliminated some attributes/columns that I believed did not impact the decision on whether or not a loan was granted. The attributes eliminated were:
* Percentage income
* Housing
* Job
* Dependents

**Programming approach:**

I loaded my data from the .csv file into a 2D array. For specific continuous attributes, that required to be discretized, I had a method inside my ArrayLoader.java file that created a range for every attribute that needed to be discretized. The output was a 2D array containing all the information from the training data and ranges for continuous attributes.

Next, I divided my data into 2 different objects, Attributes and Attribute Values. I stored my attributes and attribute values in a HashMap<Attribute, HashSet<AttributeValues>>

Splitting data as these objects enabled me to calculate the entropy.

Next, I calculated my information gain, initially for my whole dataset. I then added all the different IG into an ArrayList, sorted it and picked the max IG. I established this Attribute as my root for the Decision tree. Using a Queue, I stored all the attribute values related to my attribute with max IG. I created a new data set (2D array)which was a subset of the initial data set but was restricted to 1 attribute feature of the max IG selected. Next, I ran this new restricted data set into my read attributes method that created a new HashMap of attribute and corresponding Attribute values.

I only created new data set if the old data set had more than 60 instances to prevent my data from over fitting. If there weren’t 60 or less instances, I visited my Queue and selected the next attribute value to analyze given my max IG attribute.

As I calculated max IG, I added the attributes to my decision tree, which is an N-Ary tree and added its children as well.

**Data sets used:**

* 100 randomly picked instances of the given data, removed the classified column
* 900 of remaining instances of given data were used as training data used to build model
* 100 of 900 training data used to see how well model runs on subset of training data without classification
* Actual classification of the sub 100 of the 900 training data used to calculate accuracy
* Actual classification of the randomly selected 100 instances used to calculate accuracy

**Observations:**

Against bank loan data:

* When I ran a subset of my training data against decision tree (without the <60) constraint on number of instances in Data set, my accuracy was higher for training data (93%) (sub set of 900) in comparison to when I ran my test data(70%) against the decision tree. However, when I placed the <60 constraint, my test data had a higher accuracy (75%) than the subset of training data (68%)

Again weather data:

* I ran my weather data with<60 and with < 3 constraint, with <60, my model was 64% accurate. With < 3, my model was 100% accurate. Both times, I ran all the 15 instances from the textbook which was also used to train the model.