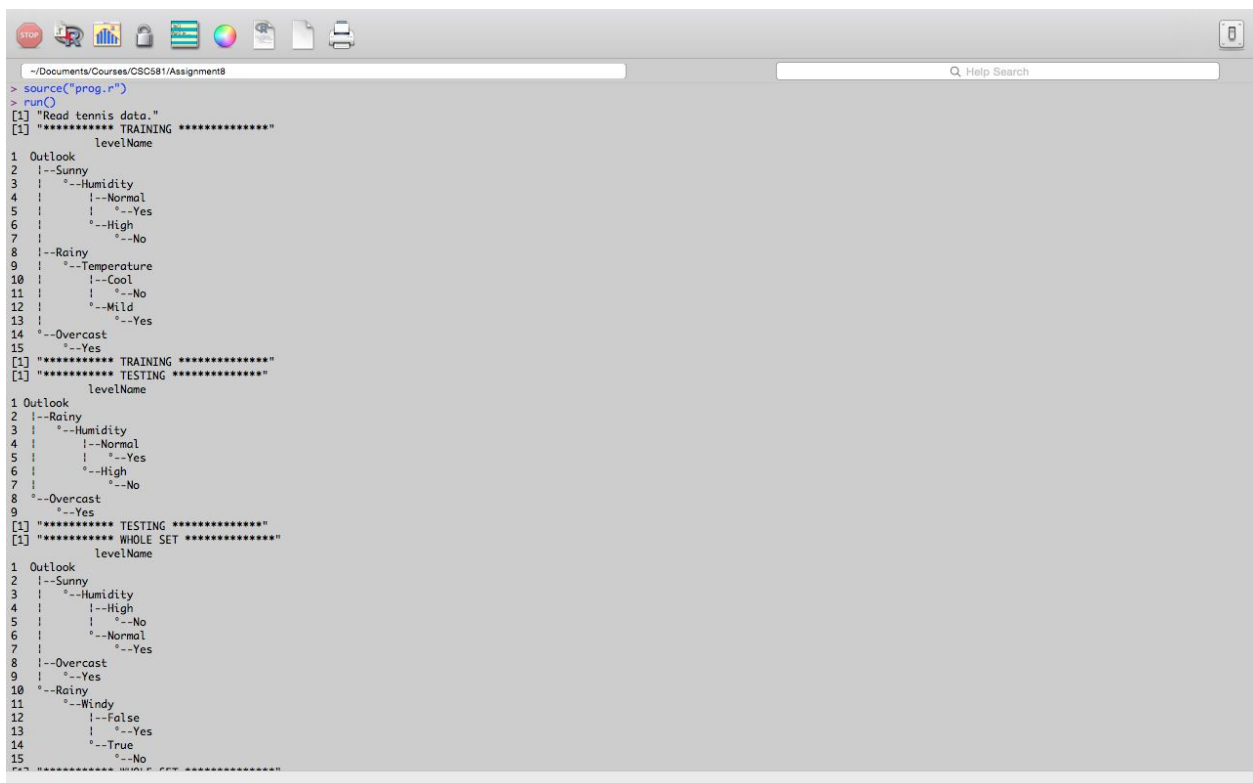


# Implementing the ID3 Algorithm

Program prog.r contains the source for the implementation. To handle basic tree services, I used the data.tree library.

Figures 1 & 2 show runs of training, test, and the whole data set. I did a 75/25 split of the tennis set for training/test. You can see from the runs how wildly different the decisions trees are for different samples of data. It highlights that just slight changes to the Yes/No values for an attribute can affect greatly the entropy for that attribute, shifting the order in which it is chosen for entropy reduction.



```
> source("prog.r")
> run()
[1] "Read tennis data."
[1] "***** TRAINING *****"
      levelName
1 Outlook
2 |--Sunny
3 |  |--Humidity
4 |  |  |--Normal
5 |  |  |  |--Yes
6 |  |  |  |--High
7 |  |  |  |--No
8 |--Rainy
9 |  |--Temperature
10 |  |  |--Cool
11 |  |  |  |--No
12 |  |  |  |--Mild
13 |  |  |  |--Yes
14 |--Overcast
15 |  |--Yes
[1] "***** TRAINING *****"
[1] "***** TESTING *****"
      levelName
1 Outlook
2 |--Rainy
3 |  |--Humidity
4 |  |  |--Normal
5 |  |  |  |--Yes
6 |  |  |  |--High
7 |  |  |  |--No
8 |--Overcast
9 |  |--Yes
[1] "***** TESTING *****"
[1] "***** WHOLE SET *****"
      levelName
1 Outlook
2 |--Sunny
3 |  |--Humidity
4 |  |  |--High
5 |  |  |  |--No
6 |  |  |  |--Normal
7 |  |  |  |--Yes
8 |--Overcast
9 |  |--Yes
10 |--Rainy
11 |  |--Windy
12 |  |  |--False
13 |  |  |  |--Yes
14 |  |  |  |--True
15 |  |  |  |--No
```

Figure 1.

Looking at the confusion matrix (Figure 2), the training set shows a 100% accuracy rate. The decision tree seems to be a good model for classifying whether or not someone plays tennis given weather-related factors.

Observed	Predicted	
	+1	-1
+1	6	0
-1	0	4

Figure 2.