CS 4804 Homework 5
Solution Sketches

1. **(10 points)** One answer is that after branching on an attribute, the entropy with respect to that attribute will be zero, so there is no additional gain (in information) from testing that attribute again. This assumes that each node in a decision tree branches on all possible values of a given attribute. If, however, the node tested a condition such as *Temperature* $<$ *27?*, we might have occassion to test *Temperature* again. For instance, in the branch labeled *No* we might have an additional test such as *Temperature* $<$ *30?*

2. **(10 points)** Let us assume that each node tested a single attribute, in which case we can think of a path from the root to a leaf as a conjunction of boolean attributes. A DNF is a disjunction of such conjunctions, so the number of paths in the decision tree will be equal to the number of clauses in the DNF. Just as different clauses can have different number of literals in them, the leaves in the decision tree might be at various levels. The depth of the tree will be equal to the length of the longest clause in the DNF.

3. **(80 points)** The exact answer to this question depends on the specific procedure by which training and test sets were created. Some of the observations you can make are:

   - With high probability, *Odor* (attribute #5) will be selected as the attribute with the highest gain, which will lead to a 9-way fork at the root. Of these branches, all but one will lead to a subtree with entropy zero, so no further branching would be needed. The branching after this point is sensitive to the way the datasets were partitioned but one common choice is *Spore-print-color*.
   - You will find that the best tree is at most three or four levels deep. In other words, performance on the test dataset would degrade if you tried to refine the tree beyond this point.
   - Test set error must be slightly above training set error and closely track the latter. The mushrooms are rather well behaved, so you will be able to extrapolate your learning to the unseen mushrooms as well.
   - The learned tree will mimic many of the simple rules given in the ".names" file, especially those dealing with *Odor* and *Spore-print-color*.

   The rigorous way to validate decision trees is to employ cross-validation, where we compare training set and test set performance over *many* possibilities of picking these sets and distill all the answers into a single measure of how much to refine the tree.