

LCS-ES Group Meeting #13

Exercise 1: Find the Animal

Download and unzip `animals.zip` from the Blackboard site. Study the Java source code, which is an incomplete implementation of an animal guessing game. As marked `/**` in the code, the model is missing three segments. First, the code to initialize the array of animals from the file `animals.data`, second the code to update the scores file, and finally code to lookup an animal by its weight (using a binary search).

The input file format is as follows

- Line one includes the number of animals (n) described in the file.
- Lines 2 to $n+1$ each include an animal name (a string), its weight (an integer) and a file name (a string). For example,
`bear 600 bear.jpg`

After you complete the game code, try to find the animal by clicking on the animal icons. What strategies can you use? What strategies might a computer program implement? Is there a best Strategy?

Note that this program is meant to run as an application not an applet because it needs file access. Note also that it replaces the view and controller from MVC with a `UIDelegate`, which includes the functionality of both. This is common in Java programs.

Exercise 2: Hangman

Download and unzip `hangman.zip` from the web page. Study the code. The model has two incomplete methods, which need to read in the list of words from the data file `hangman.data` and then the method that randomly picks the next word for the player to guess.

Consider adding code to output scores to a file.

The input file is simply a list of words one per line.

