**Samantha Voigt & Brenna Carver**

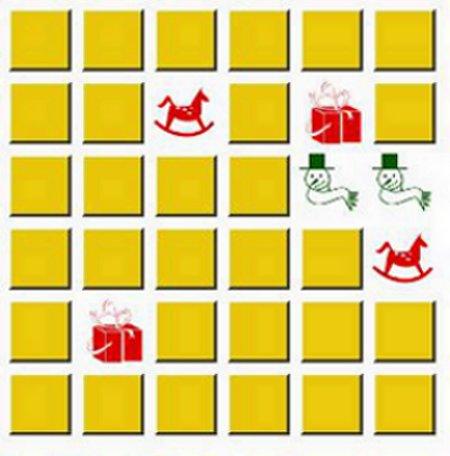
***Summary:***

We will be making a memory game, where a “card” can be “flipped” (by clicking on a

button underneath the card) to reveal an image that has a match somewhere else

among the other cards in the game. Two cards can be “flipped” at one time and if the image on the cards match they will remain flipped, showing the image, for the rest of the game. If the images on the two flipped cards do not match, there will be a small delay (using tkinter’s .after method, so the user can see what the two images were before the rest of the code runs) and the two cards will be flipped back over so that they are not showing their images. The user will then be able to flip over two more cards, repeating the process. There will be a number of clicks variable that increments every time a button is clicked so that the user can see how many clicks it took them to win the game. When there are no more cards to flip over (all cards have been matched to their pair), the user is told that they won the game, and is prompted by a button if they want to play again. If the user does play again, all the cards are again “flipped” over, and are shown in a different random order. Then the game proceeds as it did before. At all times a ‘Quit’ button will be available, which will close the window and destroy the previous game.

***Picture:***



Our GUI will look slightly different than the photo above, but the photo encapsulates what the Memory Game looks like.

***List of main classes:***

Shuffle:

One of our classes will be a Shuffle class. It will randomly assign ID numbers to the photos and randomly distribute them on the game board. These ID numbers will relate to the name of the photos in our file.

ShowAndHide

Our second class will be called ShowAndHide. This class will have methods that will turn a blank image into a different image and turn the image blank again. It will do so by taking the filenames of the images and configuring them to be ‘blank.gif’ or vice versa.

Main

The third class will hold the bulk of our methods and will be a child of Tk. It will create widgets, grid the photos, and handle events, meaning that when a user clicks a photo, the click variable will increment by one, it will check to see if it is the first or second photo to be clicked, and if it is the second photo, it will check if the IDs of the first and second photo are the same. If they are the same, the photos will stay shown and the match variable will increment by one; otherwise, they will be hidden again. When the match variable reaches the number of cells divided by two, the game will be declared over.

***Actions:***

Photo Click

When the user clicks the photo (which is actually a button widget with an image, either the blank.gif or the image that needs to be matched, instead of text), the count of the total number of clicks is incremented up by one. It will use a boolean value to see if this is the first image out of two to be clicked. If it is, it will change the image, change the boolean to False, save the ID of the image into a stringVar, and disable the button. If the boolean is False, and another image is clicked, it will change the image of that new button, check to see if the IDs of the two images match (using the saved ID of the image from the first click). If the IDs do match, it will do nothing else except disable the buttons and update the match counter by one. If the IDs do not match, the images will change back to being ‘blank.gif’. In both cases, the boolean will be changed back to True and the stringVar that holds the ID will be changed to the empty string.

Quit button click

On the quit button click the window holding the game will be destroyed.

On Play Again button click

All of the cards will be changed back to blank.gif, and the shuffle class will be used

again to create a new game. All of the instance variables will be reset and everything

that was changed throughout the course of the previous game will be set back the default so that a new game can begin.