**PROBLEM SPECIFICATION AND DESIGN DOCUMENT**

**Cassandra Lin and Christine Yi**

**a) Statement of the project**

The program is a card memory game that will allow the user to match identical card pairs in a GUI.

The user can choose one of three levels: Easy, Medium, and Hard.

The program will display a square grid of initially face up cards (standard playing deck).  The number of cards depends on the level (4,16, and 36, respectively).  After a few seconds, the cards will be turned face down. The user will then have to match identical card pairs.

On user click, the card selected will turn face up. The user will click a second card. If they match, both cards remain face up.  If not, both cards are turned face down again.

Game ends when user has matched all cards or if the Quit button is pressed .

Once the game ends, the user’s time (measured in seconds that it took the user to match all the cards) will be shown, and the playAgain button will be activated. If the user’s time is less than the highest time in the all-time high scores text file, a message will be displayed and the user’s time will be added to the text file. There are different text files for each level.

**Additional features**:

User Menu: user can choose the difficulty level. Each level will display a different grid of cards Easy (2x2), Med (4x4), Hard (6x6) i.e. levels 1-3

Play again button

**b) Description of the major components:**

1. Button class

* getLabel(self) - Returns the label string of the button
* activate(self) - Sets button to active
* deactivate(self) -  Sets button to inactive
* isClicked(self, p) - Returns true if button active and point p is inside
* setColor(self) - Sets button color

    2. playingCard class

* getRank(self) - Returns card rank
* getSuit(self) - Returns card suit

    3. Deck class

* shuffle(self) - Shuffle the deck of cards
* dealCard(self) - Returns the topmost card and removes it from the deck (used to generate grids)

    4. Grid class

* \_\_init\_\_(self, level) - Creates grid depending on the level. Cards are initially face up, but turned face down after a few seconds
* coverAll(self, gwin) - Creates a rectangle that covers the window

    5. Game class

* uncover(self,grid,pt) - Reveals the card clicked and returns its label
* cover(self, gwin, grid, pt) - Flips over the card so it is face down again
* checkMatch(self, grid, pt1, pt2) - returns True if the two cards indicated by the points are identical, False otherwise
* deactivate(self,grid,pt) - Deactivates the button that is clicked
* checkGameOver(self,grid) - Returns True if all buttons are deactivated, False otherwise

    5. main()

* While Quit button not pressed
* Display welcome and level buttons
* Once user clicks a button
* Draw grid with cards face up
* Wait a few seconds (import time.sleep module)
* Draw face down cards onto original cards
* Get user mouse click. Reveal card on one user click
* Take another user click
* If match, both cards are revealed. Else, flip both cards to face down

**c) Testing and modifications**

* Tested different levels
* Play again
* Checked for robust user interaction
* Time counter
* High scores - written to text file
* Modifications for future version:
  + make the grid 2D so it is easier to locate the button that was clicked
  + Instead of checking individual points, have two variables that hold the cards, then check if the cards match
  + Sometimes Quit button opens a new window
  + If you click one card and then click somewhere else (not a card), the card will flip over.
  + We will work on these for a future version.