Program #1 - The Original Memory Game

"The classic card matching game of visual recall"

Logistics:

- Due: **Mon Apr 20, 2015**
- Worth: **12 points** (12% of your class grade)

1. Introduction

A long time ago, in a galaxy far, far away… I played The Original Memory game a million times (give or take) with my daughter. So, for Program #1, let's build one in Java.

We'll use lots of different things in P1:

- GUI stuff happens in our `JFrame` and `JPanel` objects (section 7.2, page 366)
- The is-a and has-a relationships between classes, polymorphism, etc (Chapter 11, page 653)
- `JLabel` or `JButton` objects with images (section 13.5, page 812)
- An `ArrayList` or two (section 8.12, page 515)
- Definitely, a `GridLayout` (section 7.4, page 406)
- Some simple menus (section 13.8, page 824)
- To catch mouse clicks… a `MouseListener` or `MouseAdapter` (section 14.6, page page 901)
- I even used an `enum` (section 9.9, page 571)

There's a lot here for a first program (cough), so we'll take a little extra time and award a couple of extra points. Go!

2. How to Play

Initially, all the cards are face down. Then...

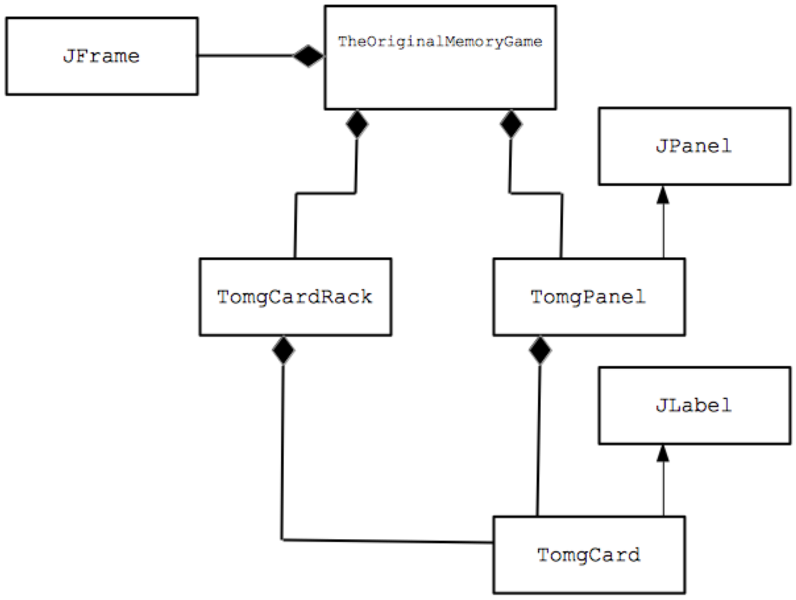
1. The user selects a card to turn it over.
2. The user selects another card, and it turns over.
3. If the 2 cards match, then they are removed from the board. If they don't match, then they are turned back over face down once more.
4. Return to step 1… until all the cards are removed.

You should:

- **Offer levels** - Your user should choose from at least two game levels: Small (3 cards x 4 cards) and Medium (6 x 8 cards). My program offers a third "Jumbo" level of 8x9 cards.
- **Keep score** - What is "the score" in a game like this? Well, there's only one player, so it's up to you. The number of cards selected before winning seems relevant and easy to track… the fewer the better. Or you could use a timer. Or, you could actually make it a 2-player game. Your choice.
- **Card theme** - Pick the theme of your cards. I like the Original Memory cards. You may prefer something else.
- **Menu** - Provide users a simple menu to get help, exit, get the current score, and restart the game.

3. Design

Here's my UML Class Diagram:



Here's a quick comment on each relationship in the UML diagram:

- ❖ `TheOriginalMemoryGame` has-a `JFrame`: This frame is the main window used to draw and play the game.
- ❖ `TheOriginalMemoryGame` has-a `TomgPanel`: This panel is added to the frame and it's where all the action takes place… drawing cards, flipping them, etc.
- ❖ `TheOriginalMemoryGame` has-a `TomgCardRack`: The game uses a rack of cards to play.
- ❖ `TomgPanel` is-a `JPanel`: Cards/labels are added here, mouse events are handled, etc.
- ❖ `TomgPanel` has-a `TomgCard`: The panel uses a list of cards when playing. Cards are labels, so they are added to the panel using a `GridLayout`.
- ❖ `TomgCardRack` has-a `TomgCard`: The rack holds all the card pairs.
- ❖ `TomgCard` is-a `JLabel`: The label's image is the card's front or back image.

I could have added `TomgCard` has-a `IconImage` to my UML because each card has a front and back image, but the diagram was already a little congested... judgement call.

We'll definitely work on this design some more in class. There are tons of choices and decisions for you to make. Design!
I'll also create a separate design notes file to track our discussions and discoveries.

3. Grading

Create a `program1` folder in your k: drive.
Place these files in that folder:

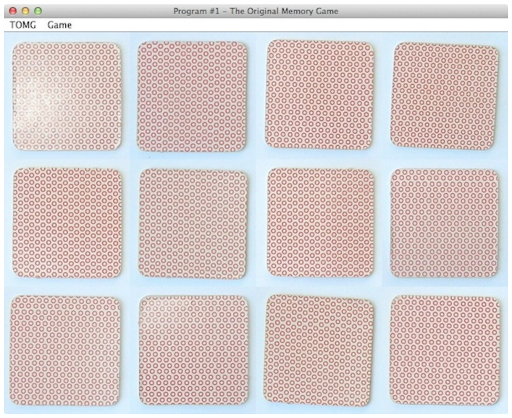
- A `README` file describing the state of your program.
- All the Java files that comprise your Program #1 solution

All your code must follow our 161 Coding Guidelines. Ugly code will be penalized with a 0-100% reduction in points. A program that doesn't even compile is worth 0 points.

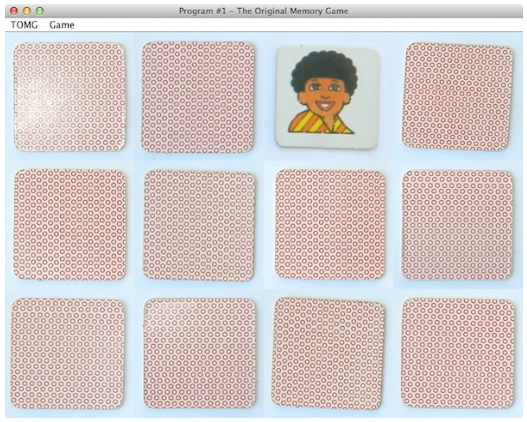
Good luck.
yow, bill

Here are a couple of screenshots to give you a better feel for things, perhaps.

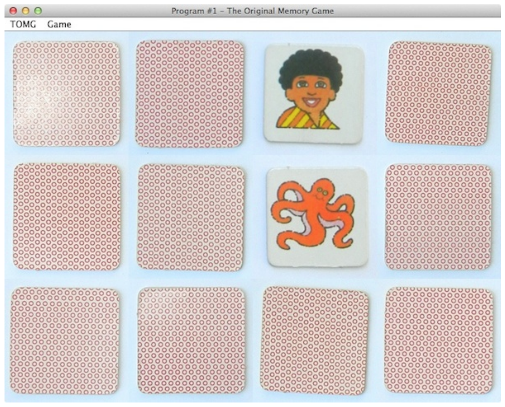
1. Initial board has all cards face down



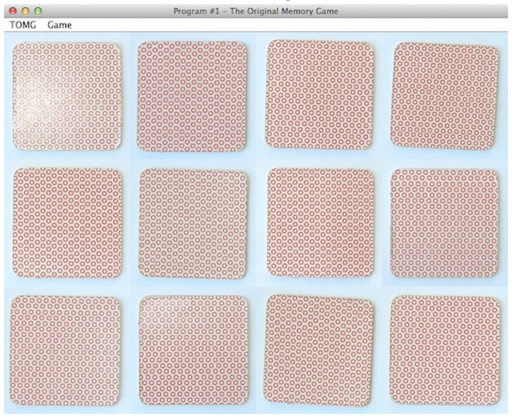
2. Pick a card and it flips over



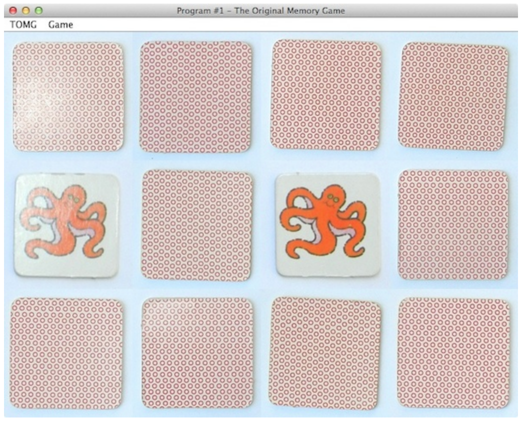
3. Pick another card, not a match



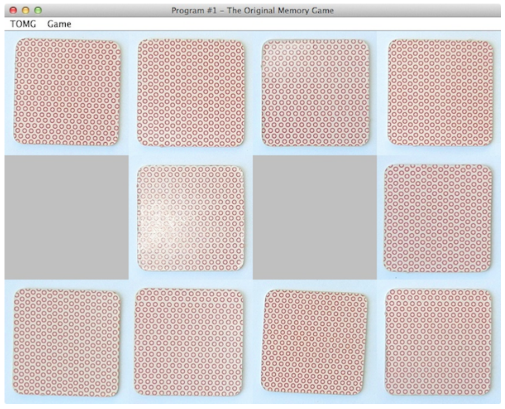
4. Both cards flip back over



5. Pick two more cards… a match!



6. Matched cards are removed



I've still got some graphic design work to do (cough again). I was going for "authentic", not "dingy", but hopefully, you get the general idea.