MATCH IT
MEMORY GAME

BY
RAJRUPA (BONNY) BAKSHI

FINAL PROJECT - CSCI E-10B
INTRODUCTION TO COMPUTER SCIENCE USING JAVA II
May 8, 2015

TABLE OF CONTENTS

ASSUMPTION

In order to play the game, we assume that the players have the following:

1) Basic knowledge of how to use a computer based graphical user interface (GUI)
2) A mouse or a track pad to click the cards
3) Installed Java runtime environment on the machine on which the game is being played
4) A clear understanding of the rules of the game

CLASS MATCHIT

INHERITANCE:

java.lang.Object
    java.awt.Component
        java.awt.Container
            java.awt.Window
                java.awt.Frame
                    javax.swing.JFrame
                        MatchIt

IMPLEMENTED INTERFACES:

java.awt.event.ActionListener, java.awt.image.ImageObserver, java.awt.MenuContainer, java.io.Serializable, java.util.EventListener, javax.accessibility.Accessible, javax.swing.RootPaneContainer, javax.swing.WindowConstants

---

CLASS DESCRIPTION:

public class **MatchIt** extends javax.swing.JFrame
implements java.awt.event.ActionListener

This class will create a *MatchIt* object, that plays a memory game in which users match pictures based on their memory.

The game consists of 8 randomly shuffled pairs of cards, arranged in a 4x4 grid, that start out face down. On the face of the paired cards there are various pictures and the goal of the game is to match two cards with the same picture until all 16 cards are matched.

To play the game, the player clicks on a card to reveal the picture on it. The timer of the game starts immediately after the first click. The card remains face up, till the player clicks on another card. Once the second card is clicked, there are two possibilities:

1) The picture of this card matches the first card, in which case the cards remain open for the remaining duration of the game and are considered “paired”

2) The second card, does not match the first card, in which case both the cards return to their original state of being face down precisely after 1 second. As such, the player in this scenario gets 1 second to memorize the picture on the cards, along with their respective locations on the grid.

In order to pair a card, the player needs to successively click on two cards with the same picture. In essence, when the player clicks on a card with a picture that had previously been revealed to him, he recalls the location of the other pair and clicks on the position of the grid where he previously saw the card in order to pair them.

To win the game, the player must pair all 16 cards at which point all the cards will remain face up and the clock that started with the player’s first move, will now stop and record the total time taken to complete the game. In successive games, the player will endeavor to beat his previous best time.

CONSTRUCTOR AND DESCRIPTION

MatchIt()
Constructs a new *MatchIt* and sets up the user interface and performs initialization.

METHOD SUMMARY

Modifier and Type	Method and Description
Void	actionPerformed(java.awt.event.ActionEvent e) Respond to button clicks and will process the corresponding method.
Void	bestTime() Calculates and displays best time
Void	flipCard() Flips over the cards to face down. static void
Void	processClick() Process the button click by analyzing the odd and even click.
Void	resetGame() Resets all elements to its initial state and shuffles the cards by calling shuffle method
Void	shuffle() This method shuffles the image icons randomly
Void	winGame() Checks for win and displays a message to let the user know that they have won that particular game

Static Void	main(java.lang.String[] args)

NESTED CLASS

CLASS MATCHIT.BASEPANEL

INHERITANCE:

java.lang.Object
    java.awt.Component
        java.awt.Container
            javax.swing.JComponent
                javax.swing.JPanel
                    MatchIt.BasePanel

IMPLEMENTED INTERFACES:

java.awt.image.ImageObserver, java.awt.MenuContainer, java.io.Serializable, javax.accessibility.Accessible

public class MatchIt.BasePanel
extends javax.swing.JPanel

Inner class of MatchIt used for providing a base panel with background image

---

CLASS MATCHIT.TIMETHREAD

INHERITANCE

java.lang.Object
    java.lang.Thread
        MatchIt.TimeThread

IMPLEMENTED INTERFACES:

java.lang.Runnable

public class MatchIt.TimeThread
extends java.lang.Thread
implements java.lang.Runnable

Inner class of MatchIt used for calculating players time in the game.

CLASS MATCHIT.TITLEPANEL

INHERITANCE

java.lang.Object
    java.awt.Component
        java.awt.Container
            javax.swing.JComponent
                javax.swing.JPanel
                    MatchIt.TitlePanel

IMPLEMENTED INTERFACES:

java.awt.image.ImageObserver, java.awt.MenuContainer, java.io.Serializable, javax.accessibility.Accessible

public class MatchIt.TitlePanel
extends javax.swing.JPanel

Inner class of MatchIt used for Title panel with background image

---

ALL CLASS HIERARCHY

- java.lang.Object
  - java.awt.Component (implements java.awt.image.ImageObserver, java.awt.MenuContainer, java.io.Serializable)
    - java.awt.Container
      - javax.swing.JComponent (implements java.io.Serializable)
        - javax.swing.JPanel (implements javax.accessibility.Accessible)
          - **MatchIt.BasePanel**
          - **MatchIt.TitlePanel**
      - java.awt.Window (implements javax.accessibility.Accessible)
        - java.awt.Frame (implements java.awt.MenuContainer)
          - javax.swing.JFrame (implements javax.accessibility.Accessible, javax.swing.RootPaneContainer, javax.swing.WindowConstants)
            - **MatchIt** (implements java.awt.event.ActionListener)
  - java.lang.Thread (implements java.lang.Runnable)
    - **MatchIt.TimeThread** (implements java.lang.Runnable)

FLOW OF EVENTS

[**Start – 1st game**]
All cards are faced down
All cards shuffle randomly

Click events
Check whether all cards are flipped open

Click a card (card 1)
  Start Timer [this happens only once, at the click of the very first card]
  Flip card 1 to face up

Click another card (card 2)
  Flip card 2 to face up

Compare card 1 image with image on card 2
  If card 1 image matches image on card 2
    Keep them flipped open
  Else flip both the cards to face down

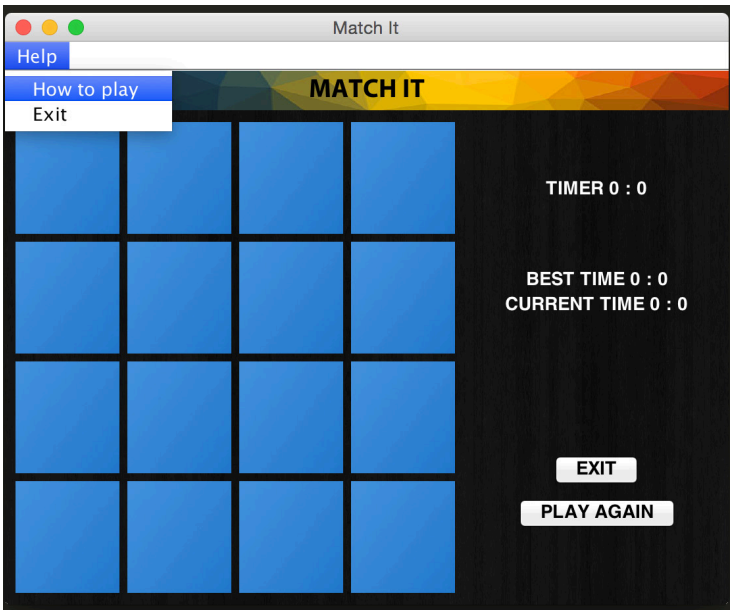
Repeat the process till all cards are flipped open in which case the Timer Stops

Player's total time is recorded

Successive games [Play Again]
Shortest time is recorded as Best Time
The player will try to beat his previous best time

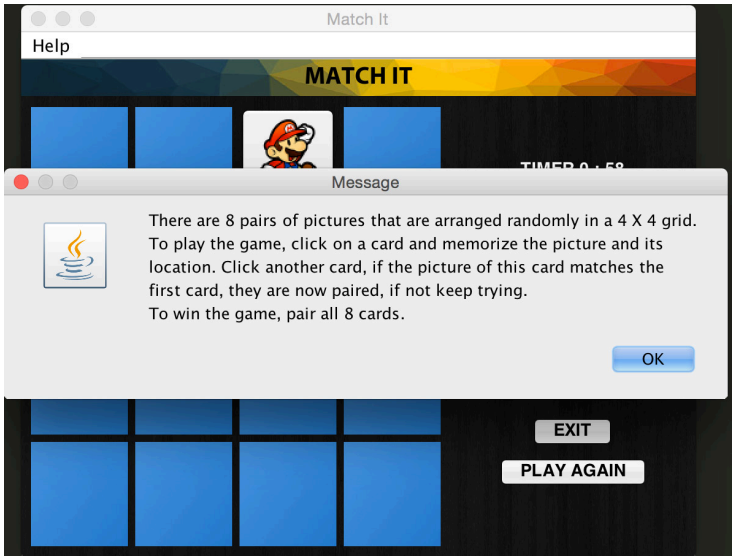
GAME DESIGN

HELP MENU – HOW TO PLAY



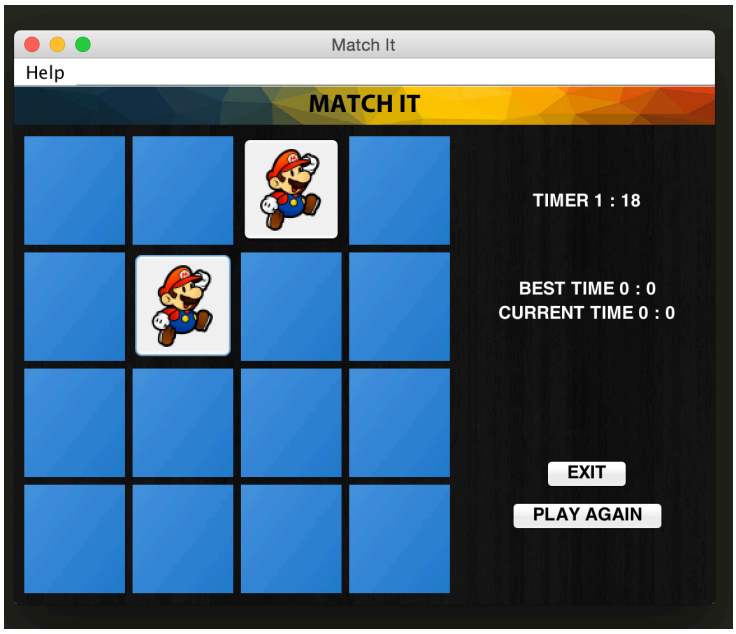
DESCRIPTION – For instruction on how to play the game click on "How to play" in "Help" menu.

JOPTIONPANE WITH INSTRUCTIONS ON HOW TO PLAY



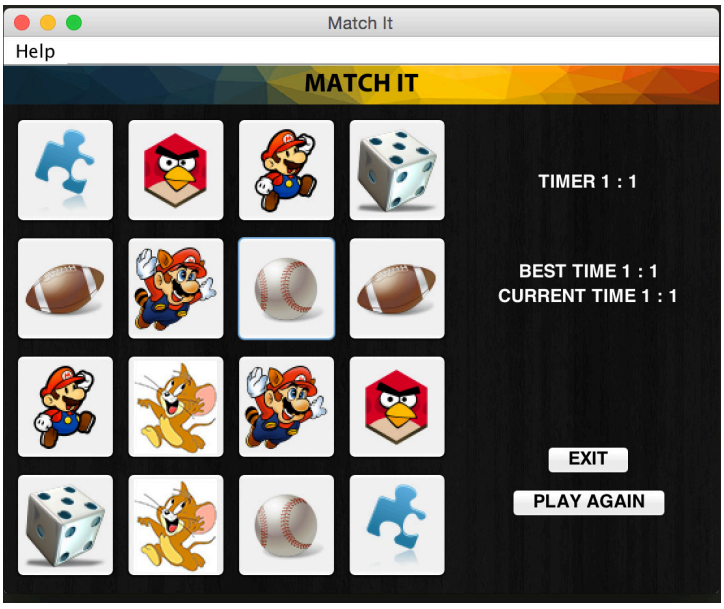
DESCRIPTION – Clicking on "How to play will open a JOptionPane message dialog with instruction on how to play the game

CARDS MATCHED



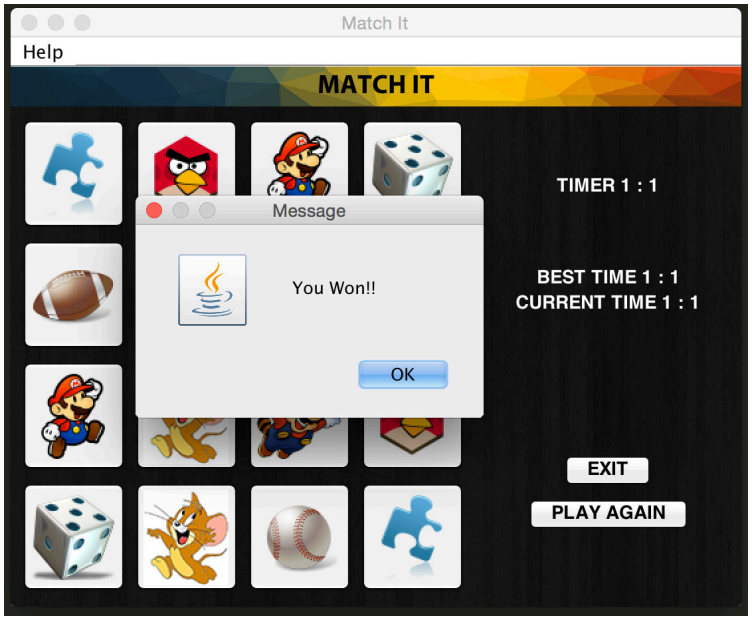
DESCRIPTION – When two cards are matched they stay flipped open

ALL CARDS MATCHED



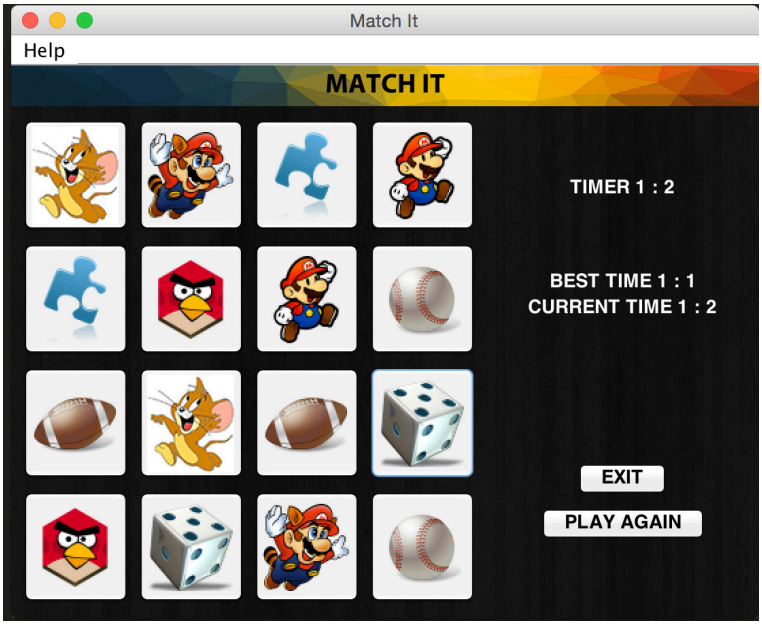
DESCRIPTION – When all cards are matched, current time displays the total time taken by the player to win that particular game.

WINNING MESSAGE



DESCRIPTION – When all cards are matched, program displays a winning message

BEST TIME



DESCRIPTION - The player can play again to beat his best time. If the current time is less than the best time, then the best time becomes the current time.