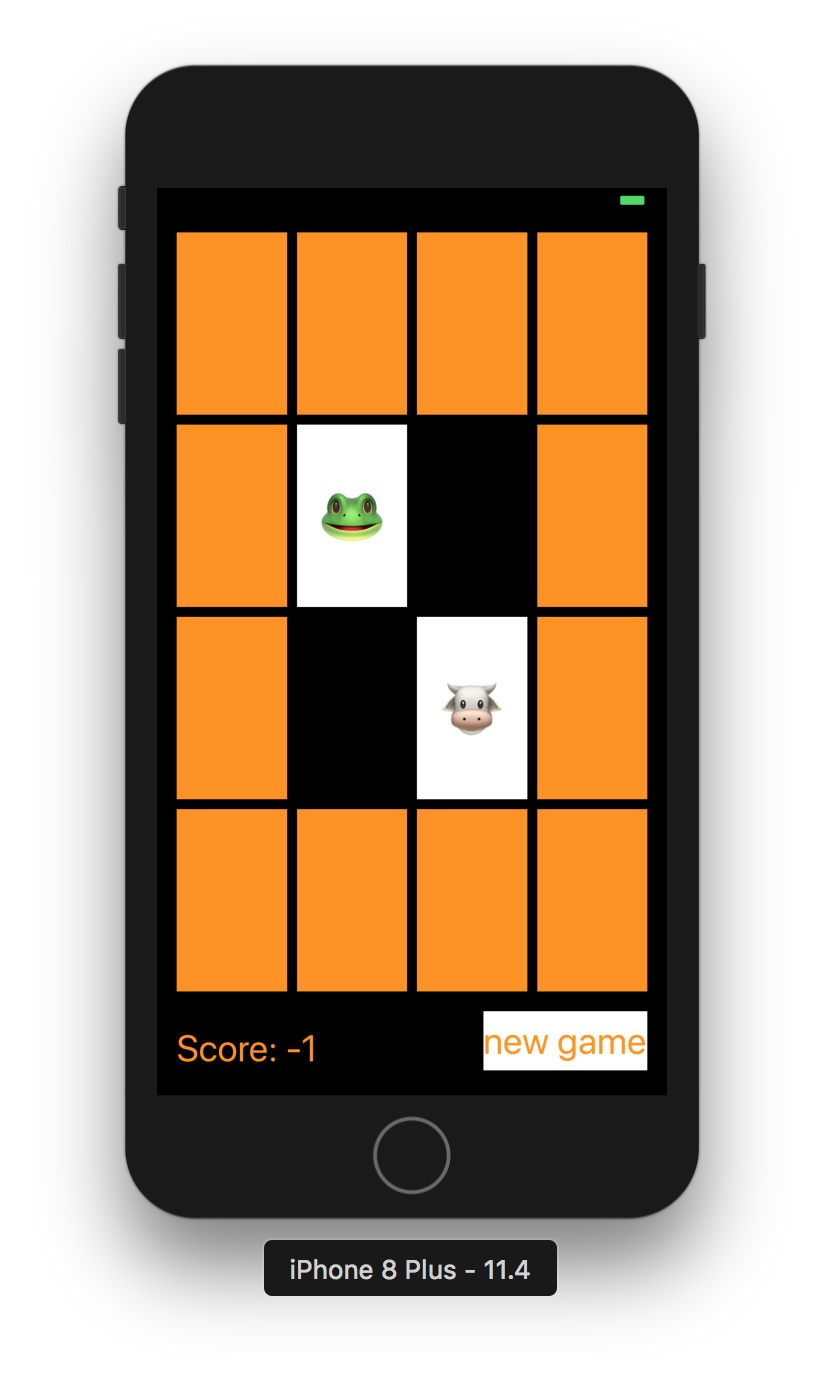
实验编号： 7 **四川师大《IOS》实验报告 2018** 年 **10** 月 **24** 日

### **计算机科学学院** 2016 级 4 班 实验名称： Game单MVC \_

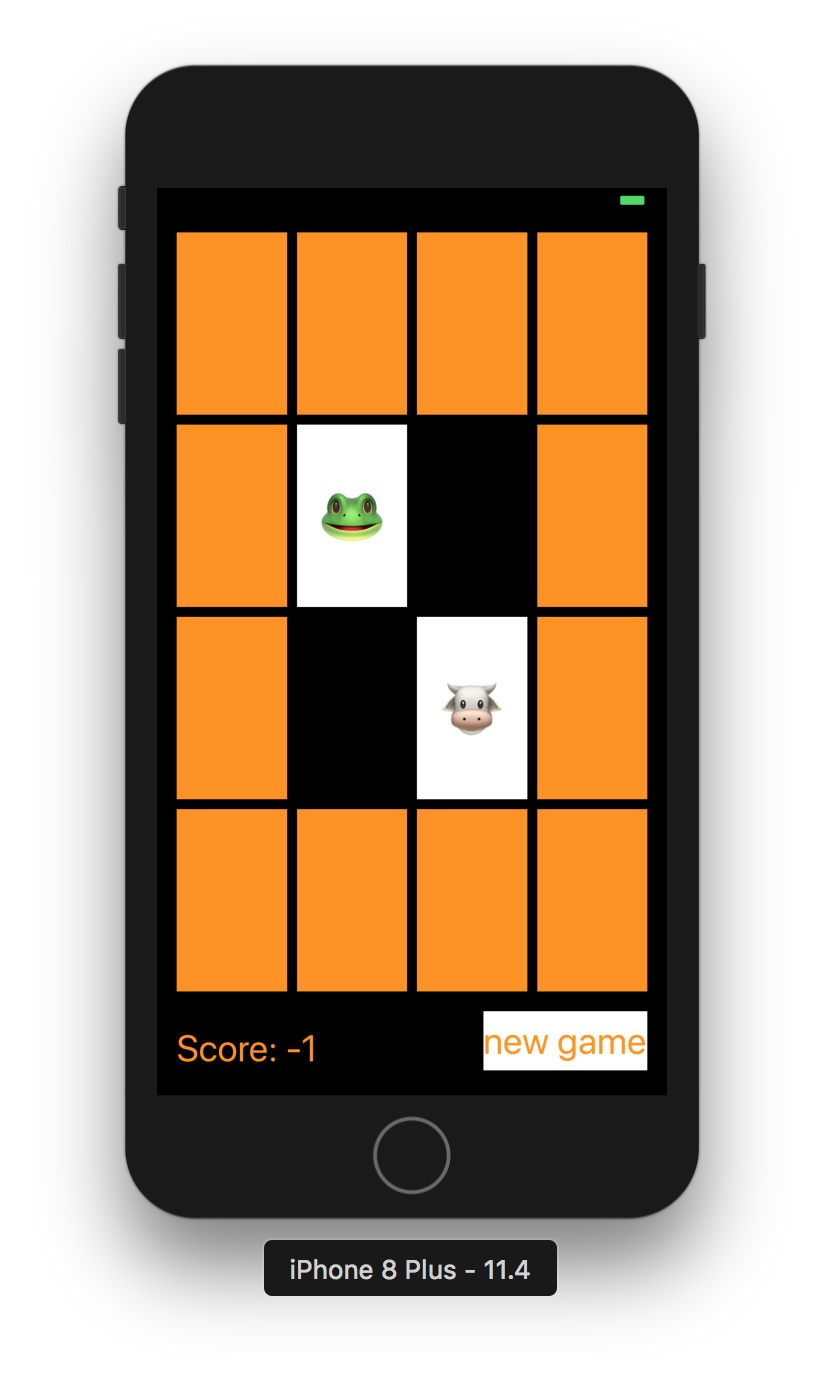
姓名：蒋星 学号：2016110418 指导老师：\_\_李贵洋\_\_ 实验成绩:\_\_\_\_\_

**实验 七 \_\_\_\_\_\_** Game单MVC **\_\_\_\_\_\_\_\_**

1. 实验目的及要求
2. 实现一款功能完整的game（Concentration）；
3. 掌握单MVC的主要思想；
4. 实验要求
5. 认真填写实验报告，要求附加部分运行界面和主要代码；
6. 对设计好的程序，检查输出是否符合预期，如有错请分析错误原因并解决；
7. 实验内容
8. 参照Stanford视频1和2完成一个game（Concentration）的制作；
9. 在(1)的基础上进一步完成Stanford Assignment 1的完整要求；
10. 采用autolayout布局解决横竖屏自适应如下所示；



1. 实验主要流程、基本操作或核心代码、算法片段（该部分如不够填写，请另加附页）
2. 参照Stanford视频1和2完成一个game（Concentration）的制作；
3. 在(1)的基础上进一步完成Stanford Assignment 1的完整要求；
4. 采用autolayout布局解决横竖屏自适应如下所示；



* 程序代码：

//

// ViewController.swift

// mytest7

//

// Created by apple on 2018/12/16.

// Copyright © 2018年 蒋星. All rights reserved.

//

import UIKit

class ViewController: UIViewController {

lazy var game = Concentraction(numberPairsOfCards:

(cardButtons.count+1)/2)

@IBOutlet var cardButtons: [UIButton]!

@IBOutlet weak var filpCountLable: UILabel!

var flipCount=0

{

didSet

{

filpCountLable.text="Flips: \(flipCount)"

}

}

@IBAction func touchCard(\_ sender: UIButton) {

flipCount+=1

if let cardNumber=cardButtons.index(of: sender)

{

game.chooseCard(at: cardNumber)

updateViewFromModel()

}

else

{

print("there is no card")

}

}

func updateViewFromModel() {

for index in cardButtons.indices

{

let button = cardButtons[index]

let card = game.cards[index]

if card.isFaceUp

{

button.setTitle(emoji(for:card), for: UIControl.State.normal)

button.backgroundColor = UIColor.white

}else

{

button.setTitle("", for: UIControl.State.normal)

button.backgroundColor = card.isMatched ? colorLiteral(red: 0.9610558152, green: 0.5509537458, blue: 0.01276976243, alpha: 0): colorLiteral(red: 0.9610558152, green: 0.5509537458, blue: 0.01276976243, alpha: 1)

}

}

}

var emojiChoices = ["🎃","👻","🧙‍♀️","🐳","😈","👹","🤖","🐹","🍣","🍜","🍤","🍡","🍱","🎂","🍨","🍜"]

var emoji = [Int:String]()

func emoji(for card:Card)->String

{

if emoji[card.identifer]==nil,emojiChoices.count>0

{

let randomInex = Int(arc4random\_uniform(UInt32(emojiChoices.count)))

emoji[card.identifer]=emojiChoices.remove(at: randomInex)

}

return emoji[card.identifer] ?? "?"

}

func flipCard(withEmoji emoji:String, on button: UIButton){

if button.currentTitle == emoji{

button.setTitle("", for: UIControl.State.normal)

button.backgroundColor = colorLiteral(red: 0.9610558152, green: 0.5509537458, blue: 0.01276976243, alpha: 1)

}else{

button.setTitle(emoji, for: UIControl.State.normal)

button.backgroundColor = colorLiteral(red: 1, green: 1, blue: 1, alpha: 1)

}

}

@IBAction func palyAgain(\_ sender: UIButton) {

game = Concentraction(numberPairsOfCards: (cardButtons.count+1)/2)

updateViewFromModel()

flipCount=0

}

}

//

// Card.swift

// mytest7

//

// Created by apple on 2018/12/16.

// Copyright © 2018年 蒋星. All rights reserved.

//

import Foundation

struct Card {

var isFaceUp = false

var isMatched = false

var identifer:Int

static var identfierFactory = 0

static func getUniqueIdentifer()->Int

{

identfierFactory += 1

return identfierFactory

}

init() {

self.identifer = Card.getUniqueIdentifer()

}

}

//

// Concentration.swift

// mytest7

//

// Created by apple on 2018/12/16.z

// Copyright © 2018年 蒋星. All rights reserved.

//

import Foundation

class Concentraction

{

var cards=[Card]()

var indexOfOneAndOnlyFaceUpCard: Int?

func chooseCard(at index: Int) {

if !cards[index].isMatched{

if let matchIndex = indexOfOneAndOnlyFaceUpCard, matchIndex != index{

if cards[matchIndex].identifer == cards[index].identifer{

cards[matchIndex].isMatched = true

cards[index].isMatched = true

}

cards[index].isFaceUp = true

indexOfOneAndOnlyFaceUpCard = nil

}else{

for flipDownIndex in cards.indices{

cards[flipDownIndex].isFaceUp = false

}

cards[index].isFaceUp = true

indexOfOneAndOnlyFaceUpCard = index

}

}

}

init(numberPairsOfCards:Int) {

for \_ in 1...numberPairsOfCards

{

let card = Card()

cards += [card,card]

}

}

}

* 运行截图：
* 



1. 实验结果的分析与评价（该部分如不够填写，请另加附页）

Github地址：https://github.com/KikcerGoer/study-swift/tree/master/homework

注：实验成绩等级分为（90－100分）优，（80－89分）良，(70-79分)中，（60－69分）及格，（59分）不及格。