

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

iOS: Evil Hangman Walkthrough

Tommy MacWilliam

Harvard University

April 7, 2011

Announcements

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ Lecture videos: <https://www.cs76.net/Lectures>
- ▶ Section videos: <https://www.cs76.net/Sections>
- ▶ Walkthrough videos: <https://www.cs76.net/Projects>

Today

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ Evil Hangman
- ▶ Setup
- ▶ Equivalence classes
- ▶ Tips and Tricks

Evil Hangman

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

▶ it's evil.

Evil Hangman

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ goal: dodge user's guess as best as possible
- ▶ strategy: be able to switch among the largest number of words
 - ▶ maximize ability to cheat
 - ▶ optimal? eh, good enough

Evil Hangman

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

▶ - - - -

▶ **words:** OWN, PUN, LOL, NVM, NOT, WON, NEW

Evil Hangman

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

▶ - - - -

▶ **guess:** N

▶ - - -: LOL

▶ N - -: NVM, NOT, NEW

▶ - - N: OWN, PUN, WON

▶ - N -, N N -, N - N, - N N, N N N: Ø

Evil Hangman

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

▶ - - N

▶ **guess:** P

▶ - - N: OWN, WON

▶ P - N: PUN

▶ - P N, P P N: Ø

Evil Hangman

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ greedy algorithm: optimize at each step to find optimal solution
 - ▶ useful for graph problems (shortest path, minimum spanning tree)

Setup

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ Utility Application
 - ▶ contains two views: `MainView` and `FlipsideView`
 - ▶ flip side often used for settings, etc.
 - ▶ e.g. Weather app

MainViewController

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ `showInfo:` **creates/shows** new controller
 - ▶ `presentModalViewController:` **just like** `pushViewController`, **but without the hierarchy**
 - ▶ **connected via IB** to `UIButton`
 - ▶ **Attribute Inspector** → **Type** → **Info Light/Dark**

FlipsideViewController

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ **done** : hides this controller
 - ▶ connected via IB to UIBarButtonItem
 - ▶ remember the `self.navigationItem`

Delegation

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ don't do it yourself, tell an object to do it for you
- ▶ make sure that object knows what to do!

FlipsideViewControllerDelegate

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ FlipsideViewControllerDelegate **declares** flipsideViewControllerDidFinish
- ▶ MainViewController **implements** FlipsideViewControllerDelegate
 - ▶ flipsideViewControllerDidFinish **defined to hide the** FlipsideViewController
- ▶ FlipsideViewController's **delegate is** MainViewController
- ▶ FlipsideViewController **tells the delegate when** DidFinish

Equivalence Classes

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ Google didn't work last time, let's try Wikipedia!
- ▶ `http://en.wikipedia.org/wiki/
Equivalence_class`

Equivalence Classes

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ $[a] = \{x \in X \mid x \sim a\}$
- ▶ clear? okay we're done, have a nice night

Equivalence Classes

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

▶ no, there's no `NSEquivalenceClass` either

Equivalence Classes

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ define a group of words sharing a given letter at a location
- ▶ order matters!
 - ▶ $-\ - N, N - -$ are different equivalence classes
 - ▶ don't forget about the option to say the letter is not present

Equivalence Classes

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ group words into equivalence classes based on user input
- ▶ how do we define a group?
 - ▶ `NSMutableDictionary`
 - ▶ `NSMutableArray`
 - ▶ `NSMutableSet`

Equivalence Classes

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ okay we have our groups, which do we use?
 - ▶ need deterministic mapping between word and equivalence class
- ▶ each class needs a unique identifier
 - ▶ not `rand()` !
 - ▶ equivalence class → identifier → objective-c collection

Equivalence Classes

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ make sure your design can determine the best equivalence class!
 - ▶ aka the largest equivalence class
- ▶ knowing the largest class, make sure you can determine
 - ▶ what words are in the class
 - ▶ what letter(s), if any, should be displayed on-screen

Time Consumption

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ try to make program as fast as possible
- ▶ look up a given index/key = fast
- ▶ iterate through entire collection \neq fast
- ▶ re-checking words that cannot be right \neq fast

Space Consumption

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ try to make program use as little RAM as possible
- ▶ `words.plist` is pretty big!
- ▶ keep data structures as small as possible
 - ▶ huge collection with lots of unused space \neq fast
 - ▶ remove anything unneeded immediately

Design

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ keep time and space consumption in mind at all points
 - ▶ don't get hung up on performance though!
- ▶ also have to keep track of:
 - ▶ already used letters
 - ▶ number of guesses made

Hidden Text Fields

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ **first responder:** `UIView` currently with focus
- ▶ `resignFirstResponder:` **blur element**
- ▶ `becomeFirstResponder:` **focus element**
- ▶ `hidden` **property of `UIView` controls visibility**

NSString

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ `length`: length of string
- ▶ `characterAtIndex`: get single character at numerical index
- ▶ `substringToIndex`, `substringFromIndex`: substrings with a single boundary
- ▶ `substringWithRange`: substring with two boundaries

NSString

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ what if I just want a `char*`?



```
cStringUsingEncoding:NSUTF8StringEncoding:  
get const char* from NSString*
```

- ▶ now we have an array of chars
- ▶ `char a = 'a'; char b = 'b'; (a < b) == YES;`
- ▶ <http://asciitable.com>

NSString

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ words in plist are in ALL CAPS
- ▶ lowercaseString
- ▶ capitalizedString

plists

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ **both** `NSArray` **and** `NSDictionary` **respond to** `initWithContentsOfFile`
- ▶ **don't forget about** `NSBundle`, `mainBundle`, **and** `pathForResource ofType:`
- ▶ **don't parse the plist a bunch of times!**
 - ▶ sloooooow

NSUserDefaults

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ **just like** `SharedPreferences`!
- ▶ store persistent key/value pairs without the hassle of a database

NSUserDefaults

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ `[NSUserDefaults standardUserDefaults]: get defaults associated with the app`
- ▶ `setObject:forKey: save a key/value pair into defaults`
 - ▶ **just like `NSMutableDictionary`: pair created if not already existing**
- ▶ `objectForKey: retrieve value associated with key`
- ▶ `removeObjectForKey: remove item from defaults`

NSUserDefaults

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

- ▶ can read/write anything saveable in a plist
 - ▶ NSData, NSString, NSNumber, NSDate, NSArray, or NSDictionary
- ▶ convenience methods
 - ▶ arrayForKey, dictionaryForKey, integerForKey, **etc.**
 - ▶ setBool, setInteger, **etc.**

NSUserDefaults

iOS: Evil
Hangman
Walkthrough

Tommy
MacWilliam

Evil Hangman

Setup

Equivalence
Classes

Tips and
Tricks

▶ example time!