

# Android: n-Puzzle Walkthrough

Tommy MacWilliam

Harvard University

February 22, 2012

# Announcements

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListView

Bitmaps

Gameplay

Saving State

Menus

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

# Today

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ dynamic GUIs
- ▶ bitmaps
- ▶ gameplay
- ▶ saving state
- ▶ menus

# Dynamic GUIs

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ UI can also be created from pure Java
  - ▶ necessary when you don't know what your GUI will be until runtime
- ▶ make sure to `import android.widget.*; etc.`
  - ▶ wildcard puts a small hit on the compiler

# Dynamic GUIs

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

## ▶ XML:

```
<LinearLayout>  
    <Button />  
</LinearLayout>
```

## ▶ Java:

```
LinearLayout layout = new  
LinearLayout(this);  
Button button = new Button(this);  
layout.addView(button);
```

# IDs

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListView

Bitmaps

Gameplay

Saving State

Menus

- ▶ `R.id.<id>` no longer applies
  - ▶ we need to maintain IDs now
- ▶ `view.setId(int id);`
  - ▶ set the numerical ID of a view
  - ▶ `id`: unique numerical integer

# Event Handling

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ `setOnClickListener(View view);`
  - ▶ attaches an event handler that will fire on a tap
  - ▶ `view`: the object that was tapped
  - ▶ `view.getId()` will return the equivalent of `R.id.<id>`

# Relations

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListView

Bitmaps

Gameplay

Saving State

Menus

- ▶ `view.getParent () ;`
  - ▶ get the parent element of a view
  - ▶ so we can get the layout containing this element
- ▶ `layout.getChildCount ()`
  - ▶ get the number of children of a layout
- ▶ `layout.getChildAt (int index)`
  - ▶ **index: numerical index of element**



# Subviews

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ `layout.addView(View view)`
  - ▶ add a subview to a view
  - ▶ view: view to add
- ▶ `layout.removeView(View view)`
  - ▶ remove a subview from a view
  - ▶ view: view to remove
- ▶ `layout.removeViewAt(int index)`
  - ▶ remove a subview from a view
  - ▶ index: index of child to remove

# Dynamic GUIs

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

▶ `AndroidWalkthroughApp1`

# List View

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

List Views

Bitmaps

Gameplay

Saving State

Menus

- ▶ displays a scrollable list of data
- ▶ template for each item fined in `res/layouts`
- ▶ every `ListActivity` has a `ListView`

# ArrayAdapter

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ ListView **displays data**, ArrayAdapter **holds data**
- ▶ `a = new ArrayAdapter(Context c, int resource, List objects)`
- ▶ `getListView().setListAdapter(a)`

# ListViews

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

▶ AndroidWalkthroughApp6

# Breaking up an image

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ Google is so helpful
- ▶ `http://imgtfy.com?q=android+break+up+image`

# Bitmaps

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ must be able to create a background from:
  - ▶ image smaller than n-Puzzle board
  - ▶ image larger than n-Puzzle board
  - ▶ image with portrait orientation
  - ▶ image with landscape orientation
  - ▶ image with square orientation

# Bitmaps

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListView

**Bitmaps**

Gameplay

Saving State

Menus

- ▶ **the Bitmap class works with images in**  
res/drawable
- ▶ `BitmapFactory.decodeResource (Resources r,  
int id)`
  - ▶ `r: getResources () ;`
  - ▶ **id: ID of drawable (i.e. `R.drawable.<image>`)**



# Resizing Bitmaps

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ images we load into our app might not fit on the screen
- ▶ `Bitmap.createScaledBitmap(Bitmap bitmap, int width, int height, boolean filter);`
  - ▶ `bitmap`: image to scale
  - ▶ `width`: **desired width of image**
  - ▶ `height`: **desired height of image**

# Maintaining Aspect Ratio

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ simply modifying the width and height to fit the device will distort the image
- ▶ need to modify width/height at the same rate
  - ▶ equivalent of resizing a photo by dragging a corner, not an edge

# Cropping Bitmaps

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ image also might not be square
- ▶ `Bitmap.createBitmap(Bitmap bitmap, int x, int y, int width, int height);`
  - ▶ `bitmap`: image to crop
  - ▶ `x, y`: origin for cropped image
  - ▶ `width, height`: dimensions for cropped image, in pixels

# Getting Screen Size

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListView

Bitmaps

Gameplay

Saving State

Menus

- ▶ `getResources().getDisplayMetrics()` returns an object containing properties corresponding to screen size
  - ▶ `getResources().getDisplayMetrics().pixelHeight`
  - ▶ `getResources().getDisplayMetrics().pixelWidth`

# Error-Checking

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ `Bitmap` objects can be pretty memory intensive
- ▶ surround code in `try {} catch {}` blocks to avoid `OutOfMemoryError`
- ▶ `bitmap.recycle()` when an image is no longer needed
  - ▶ garbage collects the `Bitmap`, freeing up memory

# Bitmaps

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

**Bitmaps**

Gameplay

Saving State

Menus

▶ AndroidWalkthroughApp3

# Tags

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ **each View can have an associated object**
  - ▶ use object to store any data you want in view
- ▶ `view.setTag(Object o)`
- ▶ `view.getTag()`

# n-Puzzle

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

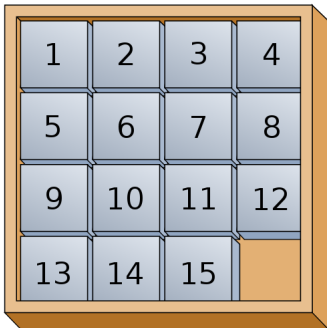
Bitmaps

Gameplay

Saving State

Menus

- ▶ also called Game of Fifteen or Fifteen Puzzle





# Representing the Board

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ board is an  $n \times n$  grid
  - ▶ 1 blank tile
  - ▶  $n^2 - 1$  image tiles
- ▶ access tiles using  $(x, y)$ ?
- ▶ access tiles using  $(\text{row}, \text{column})$ ?

# Representing a Tile

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ think about good software design!
- ▶ board tile has  $x$ -position,  $y$ -position, etc.
  - ▶ tiles are actually images, but Game of Fifteen uses numbers
- ▶ encapsulate that information!

# Representing the Blank

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ blank tile is a special case
  - ▶ what's the best way to represent it?
- ▶ blank tile is most important
  - ▶ is repeatedly searching for the blank tile efficient?

# Layout

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ layout describes how elements are displayed
  - ▶ `LinearLayout`: children aligned in single direction
  - ▶ `TableLayout`: children placed into defined rows and columns
  - ▶ `RelativeLayout`: children specify relation to parent

# TableLayout

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListView

Bitmaps

Gameplay

Saving State

Menus

- ▶ similar to HTML tables
- ▶ `<TableRow>` defines rows
- ▶ specify same number of elements per `<TableRows>`

# TableLayout

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

▶ `AndroidWalkthroughApp5`

# GridView

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ ViewGroup, **just like a ListView**
- ▶ has an associated adapter to display data
  - ▶ just like a ListView
- ▶ events triggered with `setOnItemClickListener`
  - ▶ `onItemClick`

# Swapping Tiles

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ tile should be swapped iff next to empty tile
- ▶ be sure to check:
  - ▶ two non-blank tiles cannot be swapped
  - ▶ tile is never swapped off board
  - ▶ swapping occurs visually and internally



# Swapping Tiles

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ find empty tile on board
- ▶ find tapped tile on board
- ▶ swap if two tiles are adjacent
  - ▶ checking up, down, left, and right separately doesn't seem DRY...

# Swapping Tiles

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ **ImageView displays an image**
  - ▶ `setImageBitmap()`, `setImageDrawable`
- ▶ **image displayed within ImageView can change**
  - ▶ tiles can also move

# #Winning

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ game is won when image is reconstructed
  - ▶ in Game of Fifteen, equivalent to increasing order
- ▶ if any tile is not in order, then game is not won

# #Winning

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListView

Bitmaps

Gameplay

Saving State

Menus

- ▶ show congratulatory message on win
- ▶ remember `Intents` are used to swap between activities

# Saving State

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListView

Bitmaps

Gameplay

Saving State

Menus

- ▶ need to save:
  - ▶ the state of the board
  - ▶ number of moves made
  - ▶ difficulty

# Saving State

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ `onPause()` called whenever `Activity` is removed from foreground state
  - ▶ seems like a good place to save state
- ▶ `onCreate()` called when `Activity` is first created
  - ▶ seems like a good place for `View` setup, can retrieve saved state here
- ▶ `onResume()` called immediately before `Activity` is put into foreground state
  - ▶ state retrieval also possible here
- ▶ be sure to call `super.onPause()`, `super.onResume()`, and `super.onCreate()`!

# Retrieving State

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListView

Bitmaps

Gameplay

Saving State

Menus

- ▶ `SharedPreferences` allows us to retrieve saved state
  - ▶ preferences for Activity retrieved via `getPreferences(int mode);`
- ▶ get saved key/value pairs via `sharedPreferences.get<type>(String name, int value)`
  - ▶ where `<type>` can be `Boolean`, `String`, `Int`, etc.
  - ▶ name: key corresponding to desired value
  - ▶ value: value to return if name is not found

# Saving State

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ `SharedPreferences.Editor` **used to save state**
  - ▶ **created via** `getPreferences(int mode).edit()`
- ▶ **save key/value pairs via**  
`sharedPreferencesEditor.put<type>(String name, int value)`
  - ▶ **where** `<type>` **can be** `Boolean`, `String`, `Int`, **etc.**
  - ▶ **name:** key corresponding to desired value
  - ▶ **value:** value to associate with name
- ▶ `sharedPreferencesEditor.commit()` **writes changes**



# Saving/Retrieving State

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

**Saving State**

Menus

▶ `AndroidWalkthroughApp4`

# Menus

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListView

Bitmaps

Gameplay

Saving State

Menus

- ▶ menu should appear when we press the menu button
- ▶ menus go in `res/menu`
- ▶ `<menu>` has multiple `<item>` children

# Menus

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

Menus

- ▶ `onCreateOptionsMenu(Menu menu)` fired when user requests to open menu
  - ▶ `MenuInflater` used to display menu
- ▶ `onOptionsItemSelected(MenuItem item)` fired when user selects a menu item
  - ▶ `item.getItemId()` returns an ID that can be used with `R.id.<id>`

# Menus

Android:  
n-Puzzle  
Walkthrough

Tommy  
MacWilliam

Dynamic GUIs

ListViews

Bitmaps

Gameplay

Saving State

**Menus**

▶ `AndroidWalkthroughApp2`