Milestone 2

due at midnight on Sun Feb 22 (125 points)

Your task for this activity is to implement a simple Android 'memory' card game. You will want to select a single image for the back side of cards, and eight distinct images for the front sides. (Use a 4×4 grid, so there are 16 cards total.) When the user taps one of the cards, it should reveal its front side. Then when we tap a second card, it reveals its front side and waits a few seconds. After the wait, if the front sides match, both cards are removed from the game. If not, they are returned to their back sides.

We will have covered most of the elements of this game in class, so see the videos for a reminder.

- Feb 9: demonstrated how to use `ImageView`
- Feb 11: how to size your images appropriately for the device or emulator. My tiles were approximately:
    - `drawable-mdpi`: 69×98
    - `drawable-hdpi`: 103×146
    - `drawable-xhdpi`: 137×195
    - `drawable-xxhdpi`: 206×293

  but yours may differ slightly — try to keep the same aspect ratio when you shrink them from the original. As a reminder, the `drawable` folders may be found within `app/src/main/res` in your project folder.
- Feb 11: how to keep track of which cards have which fronts, using a `tiles[]` array. When user touches a card, we logged its front ID.
- Feb 18: how to shuffle the `tiles[]` array, and how to delay removing or turning over the cards by a few seconds (so the user has a chance to see the fronts, after touching them).
  Update: the card images I'm using are available from `cards.zip`. Look in `Cards/Classic/drawable-*dpi` for the already-sized images.

To keep the app relatively simple, here are some aspects that you can **ignore** for now, or consider them **optional** if you're finished with the above and want to work ahead.

- You could keep score and show other status messages in a `TextView`.
- We're not worrying yet about saving state when the activity is killed and recreated. So it doesn't matter what happens if you switch applications or you rotate the screen.
- For layout purposes, you can assume the device screen will always be in portrait mode, and don't worry about landscape.
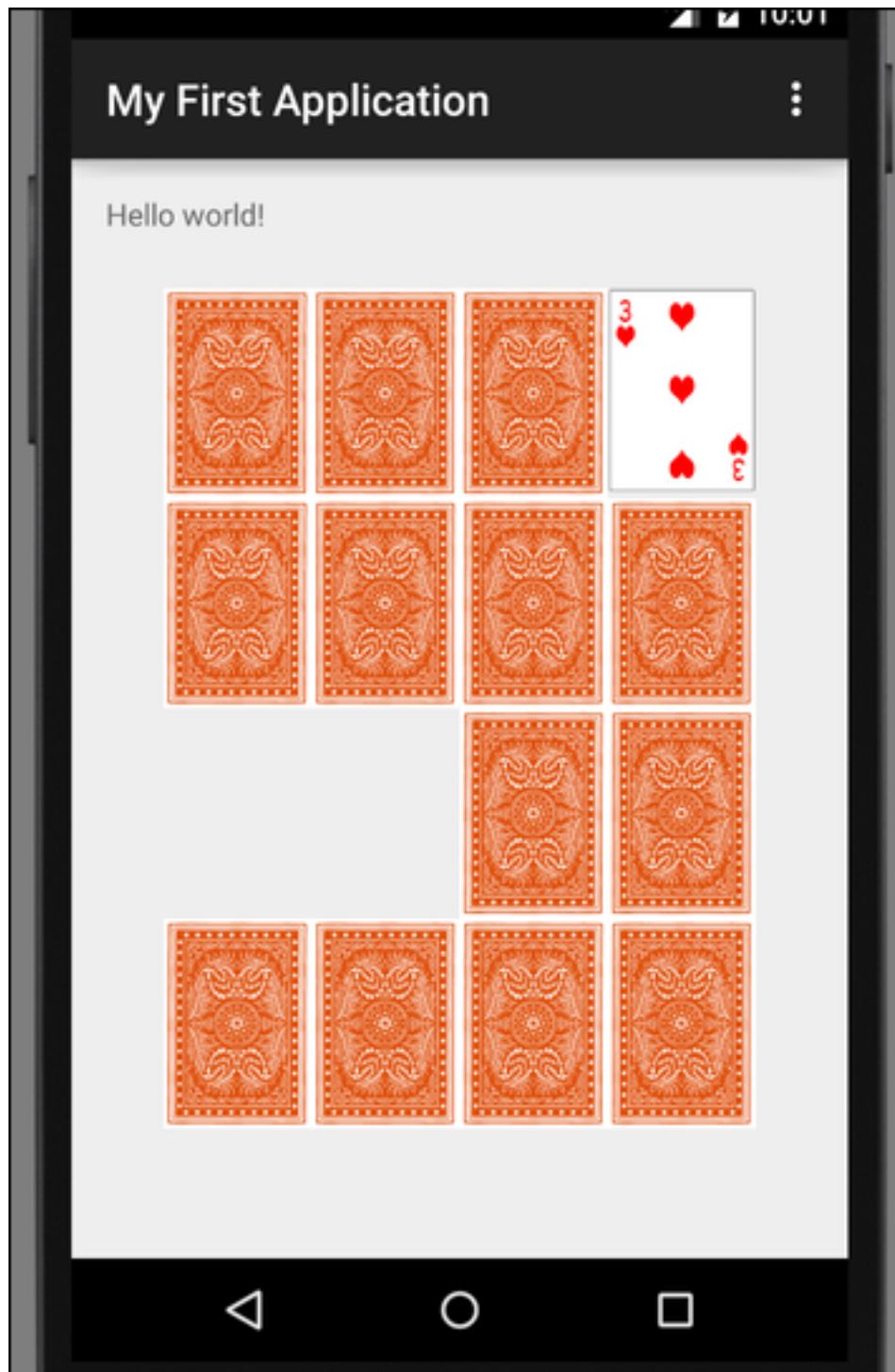- Eventually we may want a preferences screen, and some audio feedback when a card is touched or when the second choice matches or doesn't.

Figure 1: A game in progress