



COMP7506 A Smart Phone Apps Development

Assignment 1

Dr. T.W. Chim

E-mail: twchim@cs.hku.hk

Facebook:

<https://www.facebook.com/drtwchim>



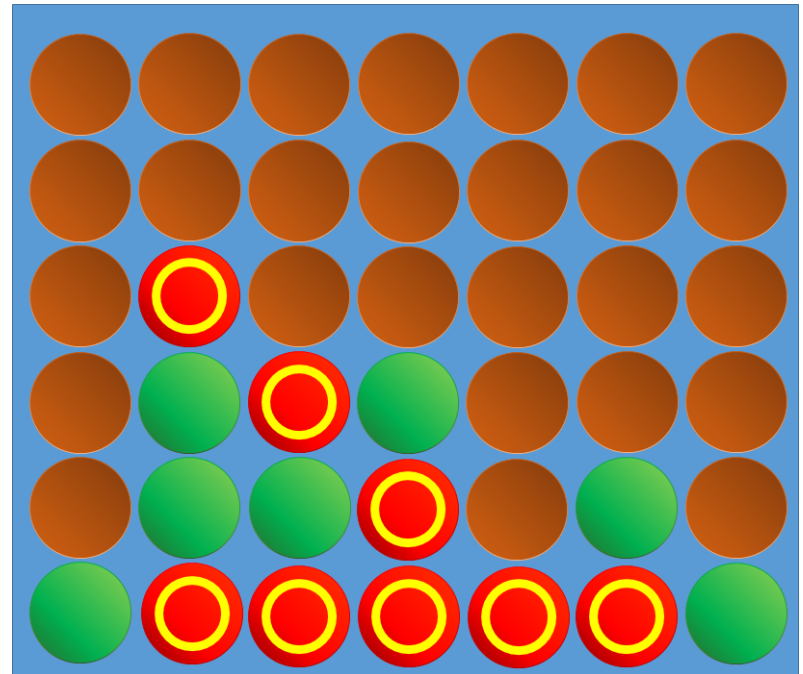
Connect4

- Simple chess game for 2 players
 - 2 different colors of chess pieces
 - Each player select 1 color
 - Take turn to place chess pieces
 - Chess pieces occupies the first available empty position of the selected column
- Aim: To connect at least 4 chess pieces of the same color in a straight line either vertically, horizontally, or diagonally



Implementation

- Basic features (12%)
 - Basic game functions
 - Switch from the start activity to the game play activity
 - Start a new game
 - Switching the move turns between 2-color chess pieces
 - Showing the current game status (current turn, winning, game draw)
 - Showing all the winning chess pieces



Implementation

- Additional features (4%)
 - Retract to the previous move(s)
- NOTE: NO AI is required
 - Human vs human mode only
- Refer to the assignment description sheet for more details

Bonus Marks (up to 2%)

- At most 2% marks of this assignment will be given to
 - Nice user interface.
 - Fluent game flow.
 - Additional game features (e.g. a new page summarizing the total number of game played, number of red/green-chess wins, time consumed in each game, audio etc.).
- **NOTE:**
 - The bonus marks are given relatively by comparing the works of students in the class.
 - The upper bound of the marks in this assignment is 20%. Suppose a student gets 19% marks plus 2% bonus marks, the student will only get 20% instead of 21% in the final marks.

Readme (4%)

- Readme documentation should be also included in order to describe the design, implementation, limitation, reference, etc. of the apps.

Two Tasks

- Design of user interface
 - Can based on Android Studio's UI builder
 - Need to label each component so that we can refer to them from the main program
 - 2 pages
 - The first page should contain candidate's name and university number for identification purpose. This page should also provide a button "Start Game" for starting the game.
 - The second page is the game main frame which will be loaded when the "Start Game" in the first page is pressed.
- Design of game logic
 - Create listener for each UI component

Layout of Game Board

- TableView (static data) vs. GridView (dynamic data)
- In our case, TableView should be good enough.
- Each cell can be an ImageButton:

```
<ImageButton
```

```
    android:id="@id/btn_board_0x0"
```

```
    android:background="#0080a0"
```

```
    ...
```

```
    android:src="@drawable/empty_t"
```

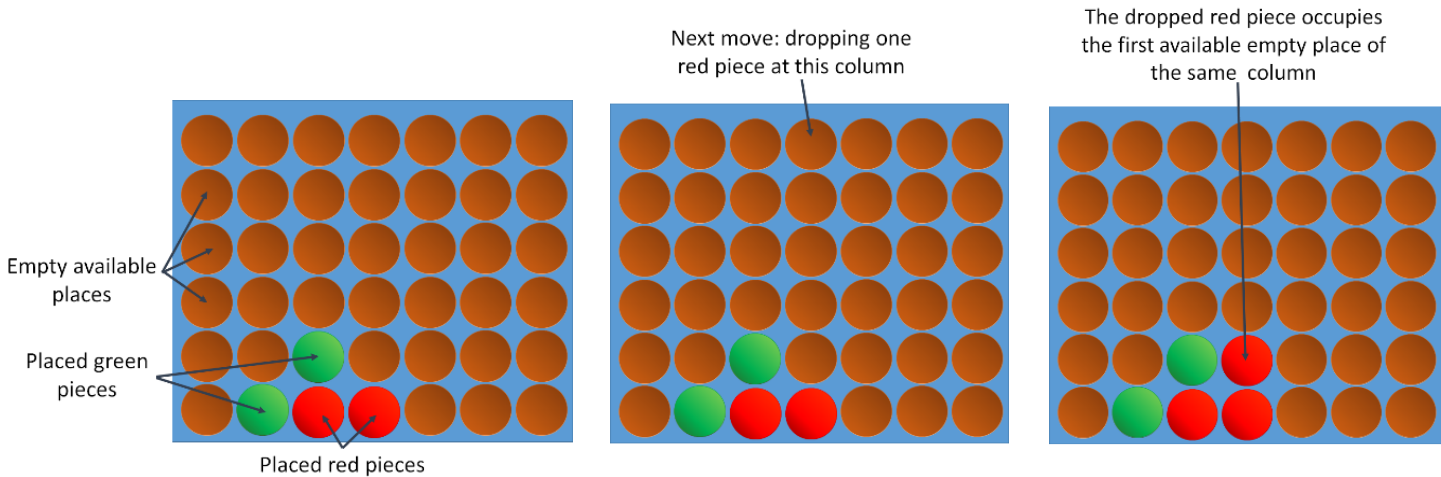
```
>
```



Representation of Chess State

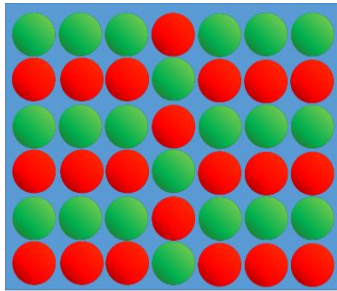
- 1-D array (`chessState[42]`)
 - Difficult to determine which elements are along a straight line
- 2-D array (`chessState[6][7]`)
 - More natural
 - Easy to determine which elements are along a straight line

Movement

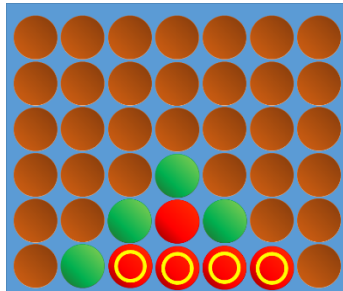


- Red-piece player moves first, and then green-piece player and so on.
- By pressing anywhere of a column, the corresponding color will be placed to the first non-empty cell of that column.
- You just need to change the images. There is no need to implement any animation for the dropping process.
- No response should be given if a column is already full.
- A "Retract" button to undo previous movements (additional feature).
 - Hint: You may need an array to store movements in order.

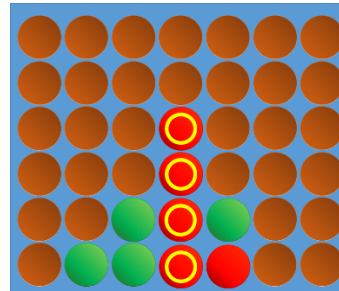
Checking of Winning Condition



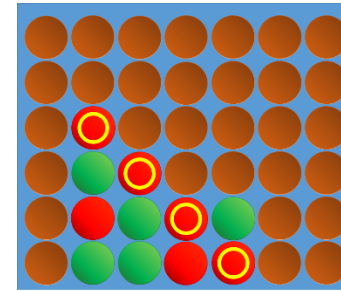
(a) Game draws when no more available move and no one win



(b) Red wins with 4 pieces connected horizontally



(c) Red wins with 4 pieces connected vertically



(d) Red wins with 4 pieces connected diagonally

- You should check the winning condition after each move.
- Cases:
 - Case 1: If the chess board is full, the game draws.
 - Case 2: If 4 pieces of any color connect horizontally, that color wins.
 - Case 3: If 4 pieces of any color connect vertically, that color wins.
 - Case 4: If 4 pieces of any color connect diagonally, that color wins.
- Change the images according to the winning condition. Proper information should be displayed also.

Demo Sample Apps

- Video Demo on Youtube
 - <https://www.youtube.com/watch?v=zuqFXXg9beY>
- NOTE: This is just a demo program. Student may have their own UI and app design.

Deliverables

- A zipped package <student's uid>.7z/zip
 - The folder of the Android Studio project
 - A readme file (docx, doc, pdf)
- NOTE: try your delivering project in the PC in HW312
 - Non-compilable codes will not be marked.

Deadline

- On or before 7th Nov., 2016 (Mon) 11:55pm.
 - Late penalty: Marks will be deducted by $N \times 2\%$, where N is the number of days after submission deadline (minimum marks = 0).

We are happy to help you!



"If you face any problems in understanding this assignment, **please feel free to contact me** or our TA.
We are very happy to help you!
We wish you enjoy this assignment 😊."