ITC 162 Grading Criteria (please read carefully and ask for clarification if needed).

Does the code run? – 50 points

- No more than 20 points will be given if the code doesn't run.

Correctness of code – 40 points

- Does the code perform per requirements as specified in the assignment?
- Does the program produce the correct prompts to the user?
- Does the program calculate the correct values?
- Does the program display the results correctly?
    - * As applicable for the assignment

Clarity of code – 10 points

- Does the program use good names for variables?
- Is the code commented and spaced so that it is easy to read and understand? Use blank lines as appropriate.

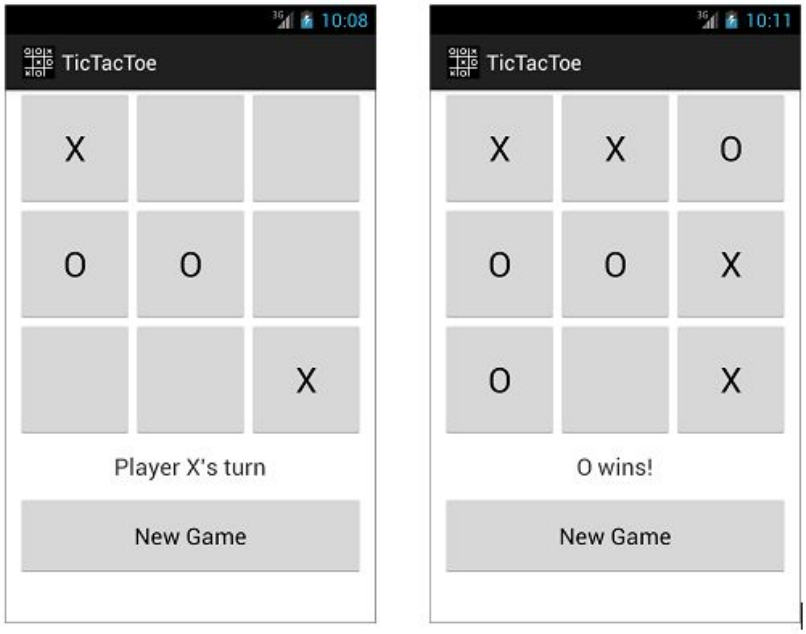
Incorrect submission

- Your assignment will not be graded if it is not submitted in the right place on Canvas or if the submission is not in the file format specified for the assignment.
- You will be allowed to resubmit, however you might lose points due to late submission.

Code the Tic-Tac-Toe app - Pair Programming Project - You may work on this project in a pair with a partner - every student should turn in their own version of

the assignment...OR, you may work on it individually.

User interface



Operation

- The app allows the user to play a game of Tic-Tac-Toe.
- The user can click the New Game button at any time to start a new game.
- The app displays other messages to the user as the game progresses such as (1) whose turn it is, (2) if a player wins, and (3) if the game ends in a tie.

Specifications

- The app uses buttons for the nine Tic-Tac-Toe squares.
- Make the app work well in portrait and landscape orientations.
- The app should save the game when the user switches orientations or navigates away from the app and then back to it.

Hints:

Layout:

1. Use a <TableLayout> in which each <TableRow> contains 3 <Button> elements.
2. For square buttons set android:layout_width and android:layout_height for each button to 100dp

Activity:

1. Use a 2D array to define the grid such as:

private Button gameGrid[][] = new Button[3][3];

2. Your game grid reference to the widget will look like:

gameGrid[0][0] = (Button) findViewById(R.id.square1);
Etc…
You will also need buttons for starting a new game and a TextView to display the game status.

3. Your activity needs to implement OnClickListener and therefore onClick(). This is where you will be implementing the logic for starting a new game and whose turn it is on the board. The turn can be implemented as a String “X” or “O”

4. Implement the following custom methods to play the game (suggested):

private void setStartingValues()

private void clearGrid()

private void startNewGame()

```
private void checkForGameOver(){
// Check for a match
        // Rows
        // Columns
        // Diagonal 1
        // Diagonal 2
```