# Image result for chaffey collegeComputer Science I

## Project 3 Board Game

#### Total Points: 50 (40 for the code 10 for the discussion)

For this project, select a simple board game to implement in C++. The implementation of the game must involve a two-dimensional array to store the game pieces on a grid. Here are some examples of acceptable choices:

* Tic-Tac-Toe
* Sudoku
* Go

You may work with a partner.

#### Project Requirements

1. You must precisely define the rules of the game so that I can understand whether your program is correct (clearly define the rules even for a well-known game such as Tic-Tac-Toe). The definition should have enough details so that another programmer can create a similar program according to your definition.
2. The program is interactive (i.e. it allow inputs from the user and provides feedback to the user). The program uses user interfaces that provide clear, well-designed, and visually pleasing feedback to the user.
3. At least one non-trivial use of an array/vector
4. Your program must use named constants where appropriate (i.e. there should be no hard-coded literals other than named constants).

#### What you need to submit for the project:

* A paper report containing the following items (only one report is needed per group):
  + 1. Title page
    2. Project name
    3. Group member names
    4. Pseudocode of your program
    5. Source code
    6. Rules of the game
    7. Describe your representative test cases along with screenshots
       1. Use sufficient screenshots to show that your program behaves correctly according to the definition.
    8. Brief discussion of your project experience on **CANVAS**
    9. On the project due date (during lab time), you will be asked to run your program and demonstrate that your program is working correctly.

#### Grading Criteria

* Satisfaction of project requirements
* Report
* Program correctness
* Documentation and coding-style - refer to the textbook’s program style
* Provide comments about your program in appropriate places. Each function should have clear and sufficient documentation for another programmer to use it without looking at the implementation.
* You must use the program template provided on Blackboard that has a program comment block containing filename, description, author, class, and date.
* Use descriptive identifiers.
* Use proper spacing and indentation.
* Use constants where appropriate.