Final Project Reports

Group Members: Ishaan Kishore, Jeffry Lai, Spencer Harley

# Designing Countries, Maps, and Continents

## Added by: Spencer Harley

## Added on: 2024-02-15

### ADDED - Graph:

This class uses a Dictionary that will hold the Key which represents the Node of the graph. Each Node has a List of other Nodes that will represent the list of adjacent Nodes (neighboring countries).

### Map:

This class just holds a Graph of every continent and a Graph of every country. This data could be later used to render the map or do other processes on all data.

### Country:

This class holds a name, owner, population, and position.

The name can be used for identification, debugging, or display.

The owner can be used for game mechanics such as distributing extra armies for controlling a full continent.

The population can be used while calculating how many dice should be used during combat.

The position can be used for rendering or automatically assigning connected countries.

### Continent:

This class holds the amount of extra armies awarded per turn for complete ownership, and a list of every country that is part of the continent.

The Graph of participating countries can be used to assign unique properties to whole continents, or be used during calculations for game mechanics that involve continents.

The function CheckIfSoleOwned returns true if the entire country is owned by one player, and returns false otherwise.

The function GetFirstOwner returns the owner of the first country in the list of countries. This can be used after checking for sole ownership to determine who should be rewarded with the extra armies.

# Graphics V1

## Added by: Ishaan Kishore

## Added on: 2024-03-02

For the graphics of the game, we're using OpenTK (A C# wrapped for OpenGL) as that's the framework we have the most experience in. As of this submission, when you begin the game, you should be greeted with the game window with the game map properly loaded. To achieve this, we made use of a basic vertex shader along with a fragment shader which handles alpha-based transparency.

For the next graphical submission, we plan on adding player units which can be moved, along with some UI that shows required information. For the final submission, we will have a properly functioning main menu that will lead into the game, along with adding juice to the visuals to give the game a better overall feel.

# Designing Player Class

## Added by: Jeffry Lai

## Added on: 2024-03-02

The Player class holds all the player information, ranging from the player colour, turn order, list of controlled territories, etc. In the player class, there are functions/methods that allow the player to gain reinforcements and trade sets of cards.

# Player Class Update 1 – Reinforcement, Attack, & Fortification

## Added by: Jeffry Lai

## Added on: 2024-03-08

The Player class has been updated to so that the player’s turn all occurs as a single function that the main program runs. Currently all the implementations for the player’s turn can only be seen through the console. This will later be added graphically.

# Map (Country & Continent) and Graph Updates

## Added by: Jeffry Lai

## Added on: 2024-03-08

The Map class along with the Country and Continent classes have all be updated to function better after more insight on the implementation has occurred. The Map now fills the Graph. As for the Graph class, a few quality-of-life updates were added for easier use and accessing necessary information required for the Risk game’s functionality.

# Main Program Update

## Added by: Jeffry Lai

## Added on: 2024-03-08

The main program file has been updated with the necessary code that allows for the risk game to be played (mostly) as a text-based game. This is planned to be implemented graphically. Currently the main program generates the map graph, asks for the number of players (minimum of 3 & maximum of 6) and sets up the players (giving them their territories and automatically dividing their starting armies evenly to each territory they control. After which, the map is shown and each player can start their turn.

Note: Due to the text-based format of the game in its current state, you do have to close the graphic map to start playing.