Option 1

Pong

Requirements:

Project Must be Named: [FirtstNameInitial][LastNameInitial]Pong For Example: TGPong

Difficulty Level: Easy

You must implement the game Pong using Unreal Engine. This is NOT a team assignment. You may discuss gameplay ideas and algorithms with your friends, but don't share any code or assets. Similarly, you may take gameplay ideas and algorithms from the web, but don't cut and paste any code nor use others art assets (Will result to a Zero grade if code is copy/pasted).

Pong consists of four types of objects:

- **A Board, that has Two players, a upper and lower boundary, and two boundaries on the left side and right side which both players (Human and AI) must protect. The board also spawns the ball at the beginning of the game and also after either the human player or the AI player loses.**

- **A Player Paddle, at the left side of the screen, which can move up and down, and is controlled by the keyboard.**

- **An AI Paddle, which is controlled by the computer and moves up and down towards the ball.**

- **A Ball, which bounces around, colliding with the other objects. If the ball collides with the left side of the screen, the player loses. If the ball collides with the right side of the screen the AI Player loses. If the ball collides with any other edges of the screen it bounces in a realistic manner. If the ball collides with the paddle, the ball bounces off at an angle proportional to where it hit the paddle. The ball's velocity should slowly increase as the game progresses, to increase the difficulty.**

For a better sense of Pong, try the provided game executable or find a Pong game to play on the web.

You are encouraged to vary the rules, as long as it doesn't simplify the gameplay or physics of the game.

Option 2

Breakout

Requirements:

Project Must be Named: [FirtstNameInitial][LastNameInitial]Breakout
For Example:
TGBreakout

Difficulty Level: Medium

You must implement the game Breakout using Unreal Engine. This is NOT a team assignment. You may discuss gameplay ideas and algorithms with your friends, but don't share any code or assets. Similarly, you may take gameplay ideas and algorithms from the web, but don't cut and paste any code nor use others art assets (Will result to a Zero grade if code is copy/pasted).

Breakout consists of three types of objects:

- **A paddle, at the bottom of the screen, which can move left and right, and is controlled by the mouse.**

- **An array of bricks of various colors at the top of the screen, which are spawned onto the level when the game starts.**

- **A ball, which bounces around, colliding with the other objects. If the ball collides with the bottom of the screen, the player loses. If the ball collides with any other edges of the screen it bounces in a realistic manner. If the ball collides with a brick, the brick is destroyed and the ball bounces in a realistic manner. If the ball collides with the paddle, the ball bounces off at an angle proportional to where it hit the paddle. Eg if it hits the paddle on the extreme left side, it will bounce leftwards, and if it hits the paddle in the middle, it will bounce straight up. Note that this is not physically accurate. The ball's velocity should slowly increase as the game progresses, to increase the difficulty.**

For a better sense of Breakout, try the provided game executable or find a Breakout game to play on the web.

You are encouraged to vary the rules, as long as it doesn't simplify the gameplay or physics of the game.