Project Proposal [30 pts] - **Ritisha De**

Write up a proposal file (in the file proposal.txt, or .doc, or .pdf) which should include the following components:

- **Project Description** [5 pts]: The name of the term project and a short description of what it will be. (NO CHANGES)

  Project Name: Ultimate Pac-Man

  Description: A fun and user focused twist on an old classic! Users play as Pac-Man whose goal is to stop the ghosts from eating a certain amount of chips before time is up. While the ghosts move around to eat the chips, Pac-Man (user) moves around to eat the ghosts. Everytime Pac-Man eats a ghost, the ghost restarts in the center of the board. Pac-Man's score is determined by the amount of chips he eats but the user will only win the game if they successfully stop the ghosts from eating the required amount of chips under the time limit. As the user passes each level, the difficulty increases as time decreases and the ghosts move faster. New abilities are available to Pac-Man every level if he eats at least ¾ of the required number of chips such as immobilizing the ghosts for a few seconds. Once the user passes the first 4 levels, they have the chance to create their own board and challenge themselves!

- **Competitive Analysis** [5 pts]: A 1-2 paragraph analysis of similar projects you've seen online, and how your project will be similar or different to those. (NO CHANGES)
  Project: https://www.youtube.com/watch?v=SuXpUP2hYP8&feature=youtu.be

  My project will have the same structure of this project in that it has the ghosts and Pac-Man as user player. My Pac-Man will also eat chips to increase its score, however, its main goal is to prevent the ghosts from eating enough chips. That is where the main differences occur. While Pac-Man in the above project is meant to avoid the ghosts and focus on eating as many chips without getting eaten, in my project the roles are reversed. Both projects will have a user created board option where the user can draw their own board by clicking on spaces. However, to keep the user from creating impossible paths to chips, the user created board class will incorporate backtracking to stop them from clicking on certain spaces. My game will also have new features that only show up after Pac-Man has eaten ¾ of the required chips. These new features are being able to immobilize the ghosts for a few seconds, and eating a green chip that deletes an entire Row.

- **Structural Plan** [5 pts]: A structural plan for how the finalized project will be organized in different functions, files and/or objects. (Changed only chip requirements)

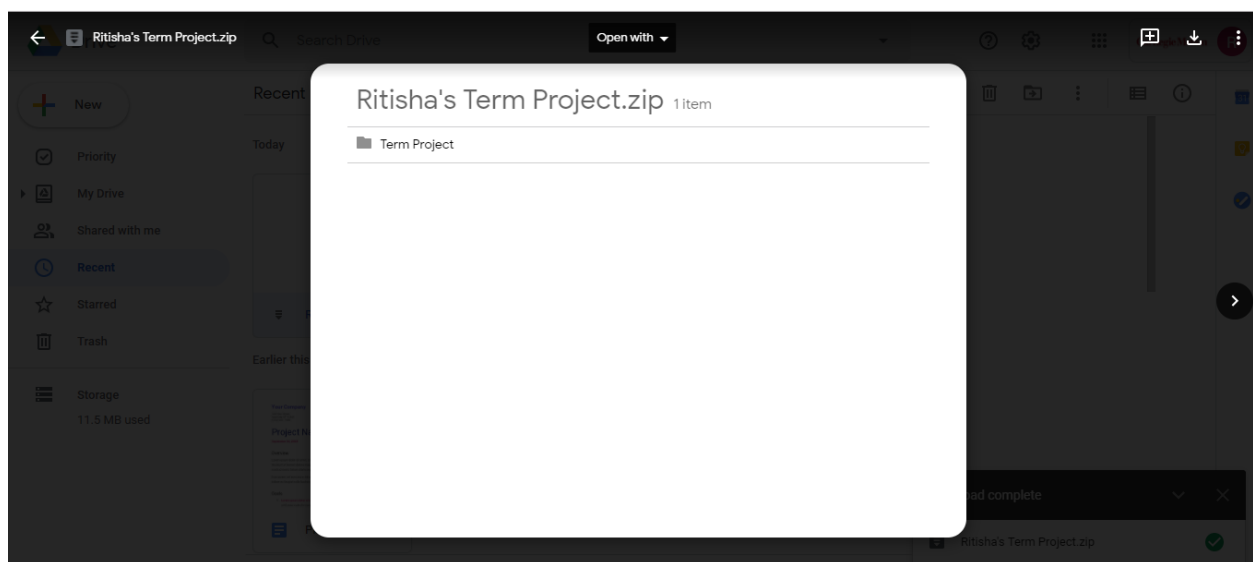|	Class	Functions
Players	**All players**	Init
		Move
		Draw
	Pac-Man	Check for collision with wall, ghost, and chips
	Ghost	Check for collision with wall and other ghosts
Boards	**All Boards**	Init
		Draw
		Chips / ChipList
		Cells/ Blue blocks
		Score Board for ghosts and Pac-Man
		Countdown
	Level 1 Board	Immobilize Ghosts when ⅕ chips eaten by Pac-Man
	Level 2 Board	Big Green Special Chip Appears when ¾ chips eaten by Pac-Man
	Level 3 Board	User Created Board options

File: Board
- All classes are imported here and correctly implemented

- **Algorithmic Plan** [5 pts]: A detailed algorithmic plan for how you will approach the trickiest part of the project. (NO CHANGES)

  Trickiest Part: Map Generation
  - Will require backtracking to determine where I can place blue blocks to ensure that there is a path to every chip
  - Will have general functions to create blue blocks

- Will have a certain limit maybe to have only ⅓ of the board to be blue / blocked
- **Timeline Plan** [5 pts]: A timeline for when you intend to complete the major features of the project. (NO CHANGES)
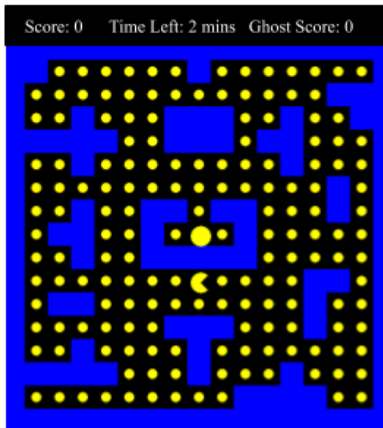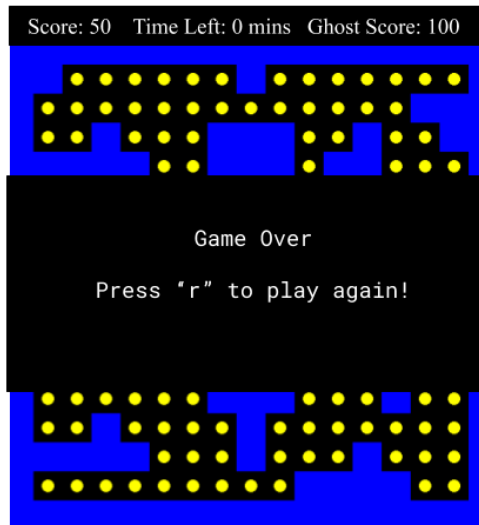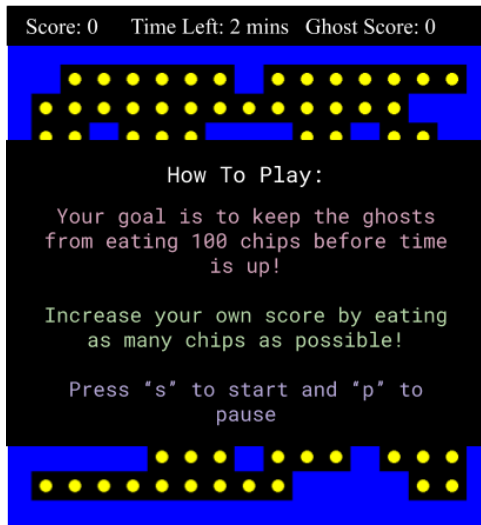
TP1	Regular Board with functioning Pac-Man and Ghosts Completed: - Functioning Pac-Man that can move with key presses - Functioning Ghosts with randomized movements - Level 1 board with chips and Pac-Man
TP2	Multiple Boards for each Level and User Created Board class
TP3	Full Reversed Board with Backtracking to have the ghosts avoid pac-man and have randomized boards for the first few levels

- **Version Control Plan** [5 pts]: A short description and image demonstrating how you are using version control to back up your code. **You must back up your code somehow!!!**
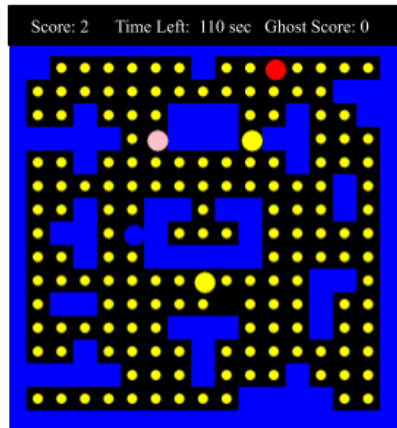
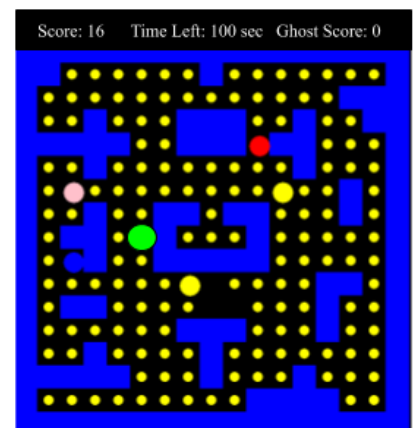**I have my code backed up on Google Drive :)**

Storyboard: (NO CHANGES)

Score: 0   Time Left: 2 mins   Ghost Score: 0

How To Play:

Your goal is to keep the ghosts from eating 100 chips before time is up!

Increase your own score by eating as many chips as possible!

Press "s" to start and "p" to pause

Score: 50   Time Left: 0 mins   Ghost Score: 100

Game Over

Press "r" to play again!

Score: 0   Time Left: 2 mins   Ghost Score: 0

Starting board: Game is begun by pressing "S" & paused with "P"

Score: 2   Time Left: 110 sec   Ghost Score: 0

As Pac-Man moves, the chips disappear

Score: 16   Time Left: 100 sec   Ghost Score: 0

If Pac-Man eats the green ball, all the chips in that column disappears

Score: 20   Time Left: 70 sec   Ghost Score: 25

As Ghosts moves, chips disappear

Score: 35   Time Left: 20 sec   Ghost Score: 75
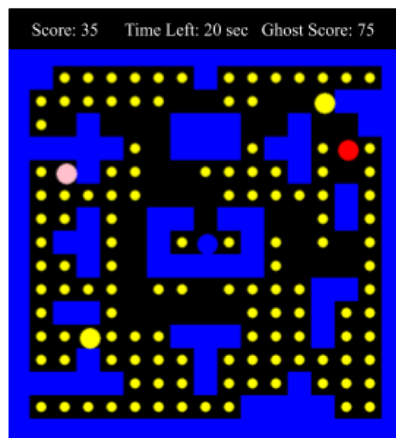
If Pac-Man eats a ghost, the ghosts restart in the middle like the blue one above

Score: 0   Time Left: 2 mins   Ghost Score: 0

User Created Board by clicking on board to create blue blocks
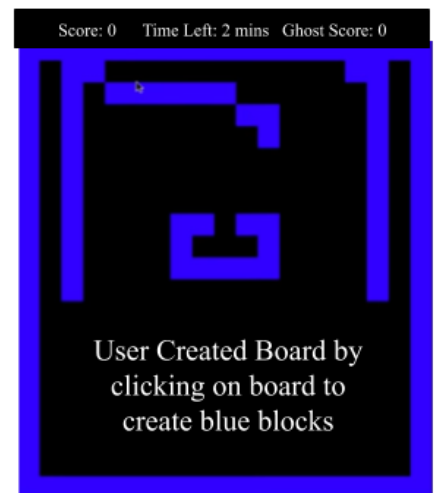
TP 2 Updated:

- **Randomized board**
- **Smarter ghosts**
- **When PacMan eats ⅕ of the total chips, all the ghosts stop moving and their color becomes white**
- **When PacMan eats ½ of the total chips, he automatically wins**
- **Updated gameover, instructions, scoreboard graphics**
- **When the ghosts eat ¾ of the chips they win**