**1. Title:** End of Semester Project

**2. General Explanation of the Project:** Super-Pacman game in C++ using inheritance and polymorphism and the SFML-2.5.1 graphics library.

**3. List of Files I Created:**

* AbstractPacman – Abstract class of Pacman from which Pacman and SuperPacman inherit
* AbstractPresent – Abstract class of presents from which all types of presents inherit
* AddLife – Class that adds lives
* Board – Game board class
* Button – Button class from which 3 buttons displayed in the main menu inherit
* Controller – Class that runs the game
* Cookie – Class of the object that Pacman needs to eat
* Demon – Demon class from which the smart and random demons inherit
* Door – Door object class
* ExitButton – Exit button class displayed at the beginning of the game
* ExtendTime – Present class that adds time to the game
* FreezeDemons – Present class that freezes the demons
* GameStatus – Class that constantly updates the game state secondary menu
* Graphics – Singleton class for graphics and sounds
* HelpButton – Help button class displayed at the beginning of the game containing instructions
* Key – Key object class
* Level – Level class
* Menu – Main menu class
* Moving – Class responsible for movements
* NewGameButton – New game button class
* Object – Class from which objects inherit
* Pacman – Class that inherits from AbstractPacman class
* Player – Player class that includes Pacman and SuperPacman
* Present – Present class
* RandomDemon – Demon class that moves randomly
* SmartDemon – Demon class that moves intelligently (using BFS)
* Static – Class from which static object classes inherit
* SuperPacman – SuperPacman class
* TimeClock – Clock class
* Title – Class that prints the game title/logo
* Wall – Wall class

**4. Main Data Structures and Their Roles** Graph algorithm breadth-first search (BFS) that finds the shortest paths.

**5. Noteworthy Algorithms:**

1. The chase of the smart demons is done using a BFS tree algorithm. The smart demon seeks to reach Pacman in the fastest way, so it searches for the shortest path between itself and Pacman that doesn't include walls and doors. The demons chase Pacman only when he is in his normal form. When he is SuperPacman, they move randomly.
2. In addition to demons that eat Pacman, Pacman in SuperPacman form can also eat demons.
3. Using SFML VIEW so I support different level sizes and adjust the level size to the window size.
4. As long as a player or demon is blinking, it cannot be eaten or moved. I do this at the beginning of a level or after a collision.

**6. Design:**

* Object class
  + Inheriting classes: Moving, Static
* Moving class
  + Inheriting classes: Demon, Player
* Static class
  + Inheriting classes: Door, Cookie, AbstractPresent, Key, Wall
* AbstractPacman class
  + Inheriting classes: Pacman, SuperPacman
* Demon class
  + Inheriting classes: RandomDemon, SmartDemon
* AbstractPresent class
  + Inheriting classes: AddLife, ExtendTime, FreezeDemons, Present
* Button class
  + Inheriting classes: ExitButton, NewGameButton, HelpButton

Design Patterns:

* Singleton classes: TimeClock, Graphics
* DOUBLE DISPATCH implementation of collisions

**8. Known Bugs: None.**

**9.** **After compilation, run super\_pacman.exe file**