Day 1:

* 1st Stage of development started which was implementing the basis of Quoridors game that

included players’ movements and visualization of the game in console.

Details:

Classes:

* Main: used in order to initialize the controller
* Controller: used in order to handle interactions between model and view – which also handled inputs for players’ movements
* Board: it resembles an abstraction of the game which implemented Observable interface
* Block: an abstraction of each block on the game board
* Direction: an enumeration which resembled direction of each wall
* Player: this class resembles each player and its location and movements
* Wall: an abstraction of each wall
* View: class used to visualize the game board in console

Interfaces:

* Observer
* Observable

Design Patterns Used:

* Observer
* MVC

Day 2:

* 2nd Stage of the development which included adding functionality to walls and implementing a Breadth-First-Search algorithm to check whether a player has been surrounded by walls and completed visualization of the game board in JavaFX.

Details:

Classes Added:

* Sake: an enumeration which included 4 main directions for checking validity of moving to a block in an especial direction from current location.

Day 3:

* 3rd Stage of development which included adding single player mode to the game with

the difficulty levels of easy and hard and also added GUI for game’s main menu.

Easy acts randomly and hard uses BFS Algorithm.

Details:

Classes added:

* Choice: an enumeration that resembled user’s choice
* Function: a functional interface written in order to be used for implementation of the strategy used by the Agent and for the sake of brevity.