[AI & Prog] Final Project

Group: 40
Max Haverkämper s2639688
Benjamin Jansen s2640163

Content:

1. **Algorithms:**
   The Algorithm we used was the Greedy Search Algorithm. It was treated in model 6. We used that algorithm for pathfinding in a maze.
   The Object with the algorithm will hunt the player and get the best path to get to him with the help of the Greedy Search.

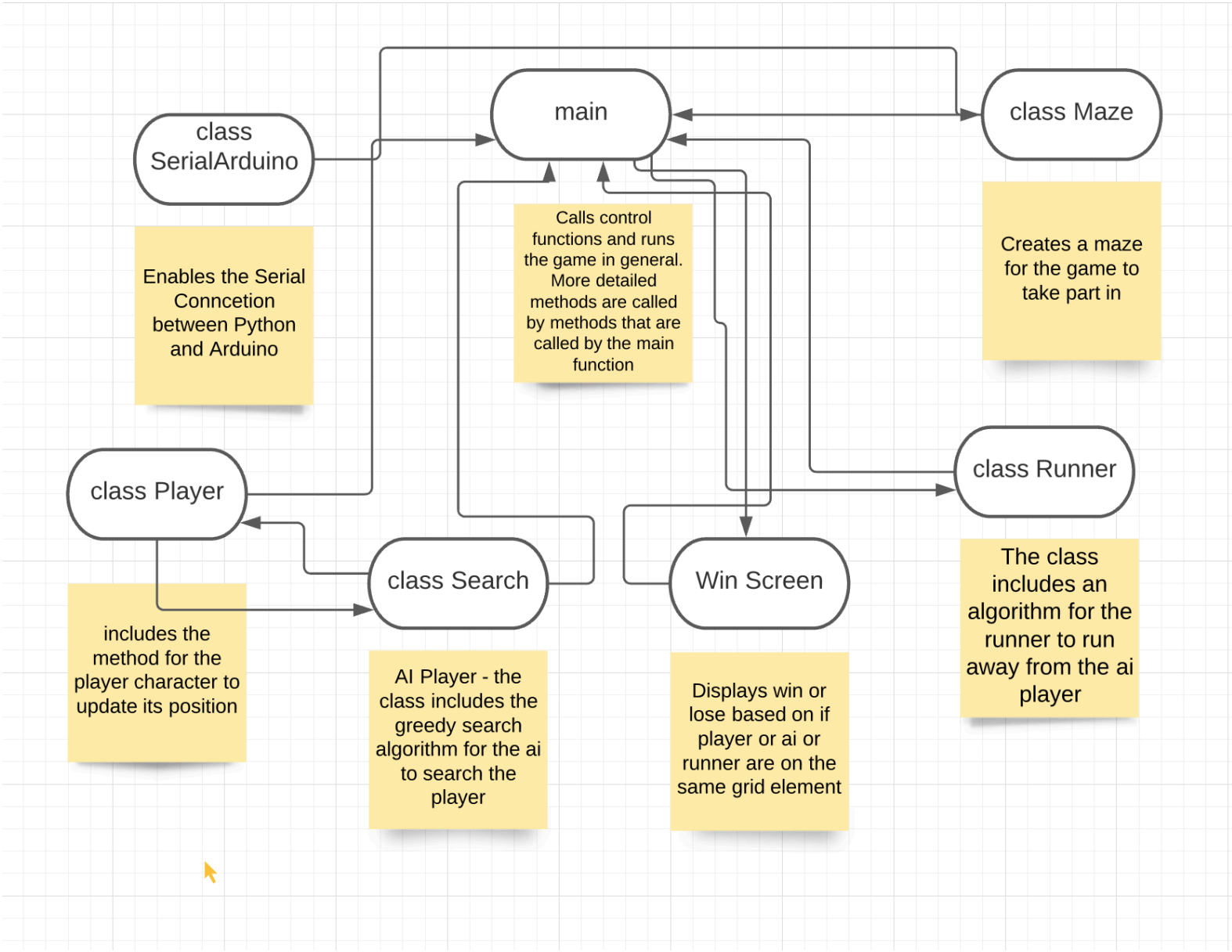
2. **Non standart input:**
   An Arduino with a potentiometer and a button is used to built a controller that let you control the position of the player inside of the maze.

3. **Use of a library:**
   The library “pyserial“ make it able to have a serial connection with the Arduino and python. The Arduino sends 2 different states to python. The state of the potentiometer and the one of the button. The value of the potentiometer will control the direction of the steps the player can make. The button make it able to execute a turn.

Concept of the program:

The game is a turn-based game in a maze.
The player is being controlled by an Arduino with a potentiometer and a button attached to it.
An Ai try to hunt the player with the Algorithm Greedy Search to get the shortest way to find the player.
In order to win the player needs to catch the target. The target is a object in the maze that runs away from the Ai in order to give the player a better chance to win and catch that target. The Manhattan distance of the neighbors in maze is being calculated and the node with the farthest distance is being selected for the next step that the target takes.
The Player will loose when Ai catches him and win if he catches the target.

Structure:


main
class Maze
class SerialArduino
Enables the Serial Conncetion between Python and Arduino
Calls control functions and runs the game in general. More detailed methods are called by methods that are called by the main function
Creates a maze for the game to take part in
class Player
class Runner
class Search
Win Screen
includes the method for the player character to update its position
AI Player - the class includes the greedy search algorithm for the ai to search the player
Displays win or lose based on if player or ai or runner are on the same grid element
The class includes an algorithm for the runner to run away from the ai player