Design Summary

# Difficulty and solution

* The soccer game has multiple objects including players, ball, fields with too many features.

I use OOP to represent these objects. For the most complicated objects player, which has different categories such as field player (attacker, defender) and goalkeeper, I design a basic class named player. This class has some methods represent some common feature all kinds of player share at a high level. For example, passing, moving and react on timer fired event. Then I define field player and goal keeper respectively. Field player also has two subclasses attacker and defender. In these two classes, what I do is just override its behavior in attacking and defense.

* Another issue is how to recognize two key pressed simultaneously. In Tkinter, the event.keysym variable can only have one key symbol, so what if we have two key pressed at the same time?

I found that, the tkinter can check two keys’ released event in the same time. So I create a dictionary called ‘pressed’ storing pressed state for each key symbol. If a key is pressed, the dictionary will store its state value as true. When the key is released, this value will be reset as false. When the timer fired, I will check the key state and recognize if a key is pressed now.

* I also have problem of building the soccer AI

I follow the guidance of the book “programming game AI by example” written by Mat Buckland.

* To create a highlight replay feature was also a challenge for me

I use a list to store positions of moving objects (players and ball) for the recent 1000 time steps. When showing the recap, I will modify moving objects’ position based on information in this list and redraw all.

# User Interface

I have played many soccer games in the market. Few of them provide the feature for friends playing together. In my soccer game, I create this feature so that friends can have fun in their home without great need for big space.