Objective:
Combine steering behaviors and state machines to create an improved soccer team.

What to do:
The goal of this project is to produce a winning soccer team for a tournament to be held in class on 10/18.

- Your entry will consist of source code that implements your new team on top of the SimpleSoccerLab implementation provided. You may not modify any of the SimpleSoccerLab source code, as this will be used as the platform on which all teams run. If you need new steering behaviors, you will have to implement them yourself using a subclass or an entirely new class. All of your code must be in a self-contained subfolder of SimpleSoccerLab titled with your last name.
- The tournament will be a round-robin style with each team playing the others. The two teams with the best win-loss record will then compete for the championship title. Buckland's default team and my team will be included as baselines. Each match will consist of one five-minute game.
- In this competition, efficiency of operation will be combined with scoring effectiveness. The score for the losing team will be adjusted up if it uses less CPU time than its opponent. The formula allocates one point for each 20% of the winner's CPU time that the loser conserves. For example, suppose Team A scores 4 points and Team B 3, but Team B uses 9109 cycles and Team A 12053. Team B's score would be \( 3 + \frac{5}{2}(12053-9109)/12053 = 4.2 \), so Team B would be the winner. In case of a tie, the winner will be whichever team uses the least CPU time.

Before starting to code, review the source code, observe the team's behavior, and read Buckland's description carefully. Then consider what design changes you will make to achieve better performance. Write up a two-page document (500-750 words) describing your design concept for an improved soccer team and submit it by 10/9.

What to turn in:
- 10/9: Turn in your design document as a Word file.
- 10/17 (5 pm): Turn in a zip file containing the source code folder for your team. You do not need to submit any other files as they should not be altered.
- 10/18: We will run the tournament in class. You may install last minute changes to your team's code.

Hints and notes:
- It would be embarrassing to lose the tournament to Buckland's team. Make sure you can beat it easily. I will also distribute the code for my team in advance.
- The book contains a number of suggestions for how to improve team play. Consult the "Practice Makes Perfect" section on page 190 for his ideas. Of course, you can come up with your own.
- The instructor will disqualify teams that abuse the spirit of the rules: for example, attempting a low-CPU-cycle defense-only strategy to win 0-0 games by efficiency adjustments.
- You may not modify any parts of the game that inject randomness. For example, you cannot make the players' kicks more accurate by removing the "fuzzy" factor.
- The CPU time rule means that there will be no tie games (no penalty kicks, either.)
- Games that reach an impasse (ball out of play, ball stuck, etc.) will be restarted with no penalty.