How good was your original planning?

Our team followed our timeline very closely by working on our project for a minimum of 3-4 hours every night. By keeping track of our progress and staying on schedule, we were able to accomplish all of our goals and even implemented 2 of our 3 cool extensions.

During our project meetings, we also made sure to write down all of our thoughts for functions and classes before implementing them. By planning out all of our code first, we were able to make a lot of progress when it came to actual implementation.

How did your milestones go?

By working on our project very consistently, we were able to keep on track as dictated by our timeline. In addition, we were able to implement two of the cool extensions that we had proposed in the initial project specification. Not only did we include a factor in the evaluation function that took into consideration how many opponent corners a move blocked off, we also included a user interface where a user could choose to play against any of our 3 algorithms.

What was your experience with design, interfaces, languages, systems, testing, etc.?

We chose to implement our project in Python with iPython Notebook. Using iPython Notebook made testing much easier. In addition,

What surprises, pleasant or otherwise, did you encounter on the way?

One thing that surprised us was copying variables. During many points in implementation, we needed to create a copy of a list or a copy of an object. We learned that setting a variable to the object of interest was insufficient, so we learned how to use copy.deepcopy(). After learning how references and pointers worked in Python, however, we were able to handle them appropriately.

What choices did you make that worked out well or badly?

One of the greatest strengths of our team was that we worked very consistently and spent a lot of time planning out what our functions and classes were going to be before we started implementing them. For example, before we started coding the major classes we needed in the game, we would sit down for an hour or two and write down exactly what values and methods we needed for each class.

What would you like to do if there were more time?

Given more time, our team would have liked to create a webpage for our game so that users could access the game directly online. We would have also liked to give our game a more appealing visual interface with color-coded blocks and a interactive board.

How would you do things differently next time?

One thing that we would do differently next time is do more initial testing. By testing our functions more rigorously, we get rid of small bugs that may be hard to find if we debugged later. While we did test all of our functions, we did forget a corner case here and there. Had we caught those earlier, debugging may have been easier later.

What was each group member's contribution to the project?

Each group member contributed to the project equally. We initially planned to split up the three algorithms equally and have each person implement one separately. However, we ended up meeting up to implement the classes and algorithms together. Write-ups and final specifications were also divided equally between all members.

What is the most important thing you learned from the project?

One of the most important things we learned from the project was learning to code more efficient algorithms. Given that one of our algorithms is computationally intensive, we really had to consider all options when it came to optimizing code and making our program more efficient. To this end, we made sure to avoid any unnecessary computations (only sorting when necessary), and inputting subsets of pieces rather than all the pieces. In terms of style, we tried to use list comprehensions whenever possible to avoid for-loops, and we made sure to comment our code so that going back to what we did a previous night was easy and clear.

Another important thing we learned from the project was dealing with copies and pointers in Python. We also learned how to use various modules.