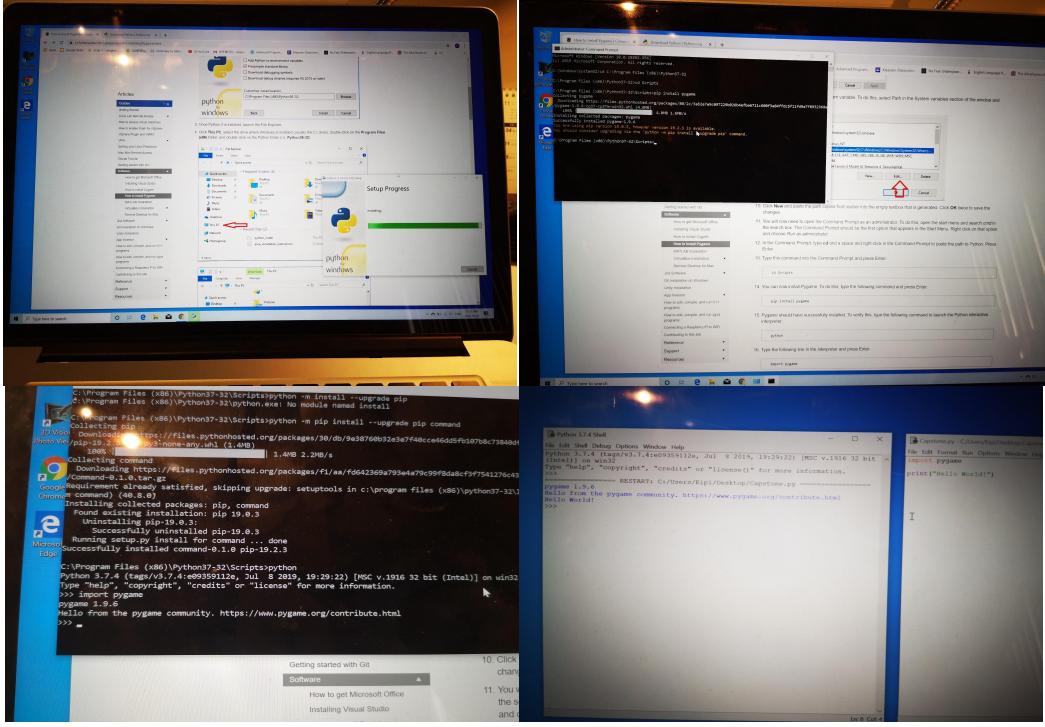
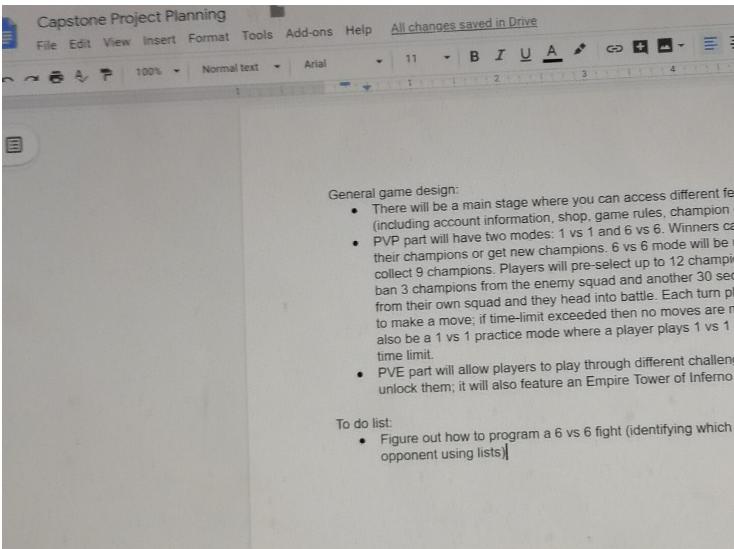


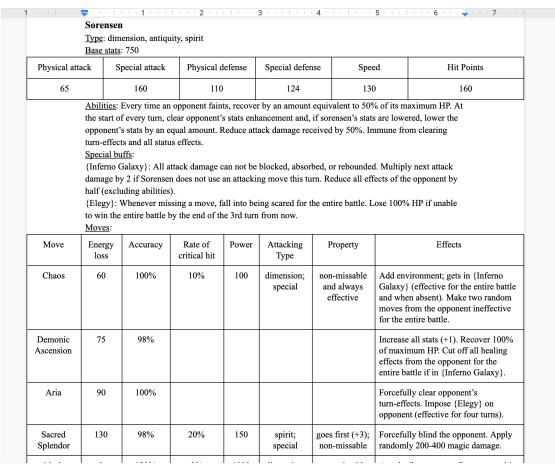
Red = important milestones

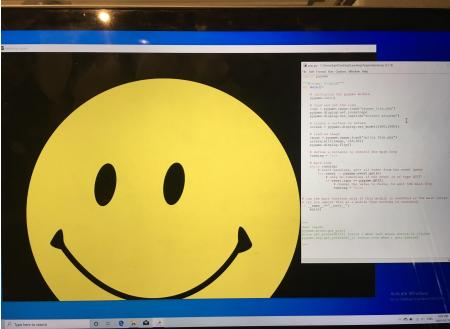
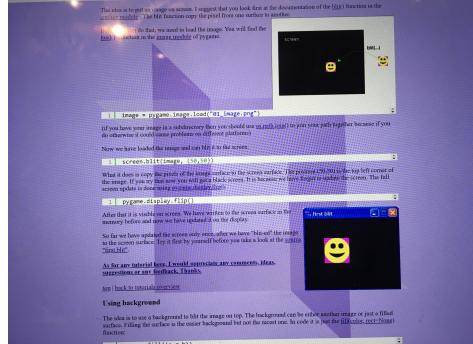
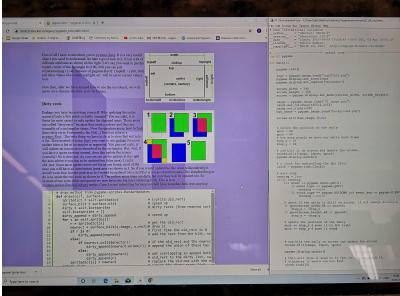
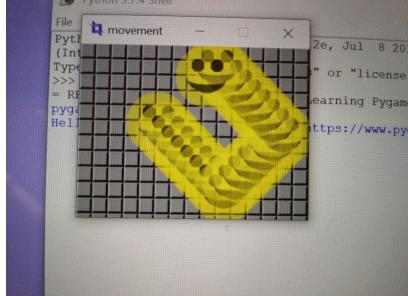
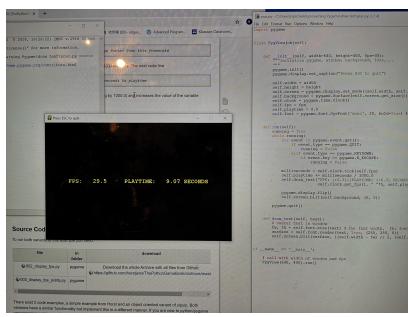
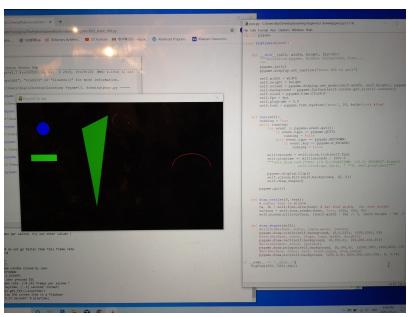
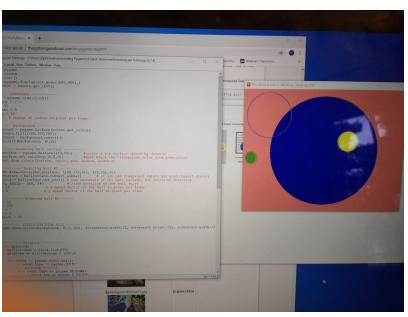
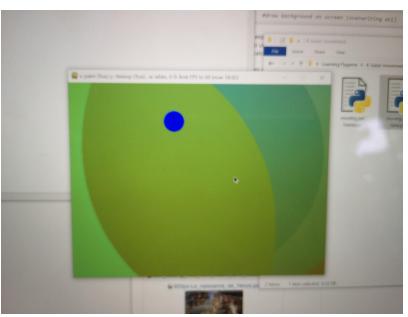
Blue = consulting with mentor

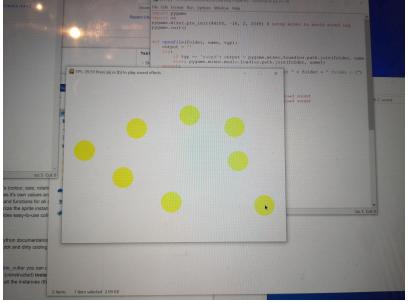
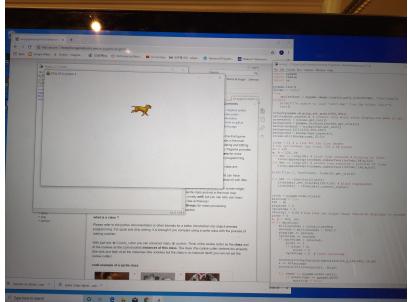
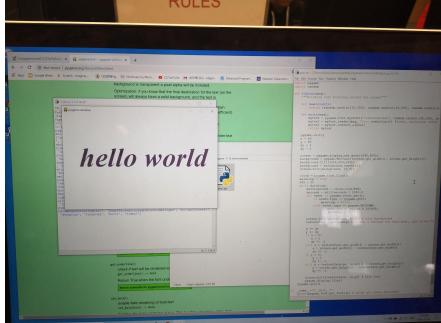
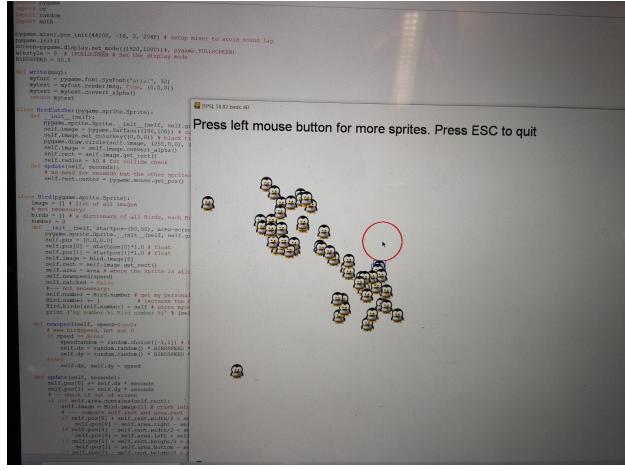
Date	What I did	Category
Sep 28	<ul style="list-style-type: none"> <li>General brainstorm and planning: I want to design a game similar to Pokemon but with different characters and a different fighting system (it will be based on the game I made for Programming 11 final project)</li> <li>I searched for Pygame and learned some general information about this special module</li> <li>I thought about what makes a successful game and concluded temporarily that it needs to have <b>good game design</b> and <b>good presentation</b> (graphics).</li> </ul>	Research & planning
Oct 1	<ul style="list-style-type: none"> <li>I talked to my second mentor Ms Braaten who's also my programming teacher. She told me Pygame might not be the best choice but I could try to make it work. It's also good to know that I can use the computers in her classroom during my freeblock on Day 2s.</li> </ul>	Consultation with mentor
Oct 2	<ul style="list-style-type: none"> <li>Game design: It will have PVE/Adventure mode where you fight multiple levels of enemies and eventually defeat the boss. It will also have online multiplayer mode where you play PVP. There will be a shop where you buy potions and other necessities like in a Pokemon game.</li> <li>Researched online and found that Python really isn't made for games because it's slow and it can hardly deal with 3D games. But it's a good tool for 2D games and it's a relatively easy language for beginners. I'm going to stick with Pygame since I do not want to make a 3D game and I know Python way better than Java or C.</li> </ul> <p>How to install Pygame for Python 3 on Mac OS X successfully, every time:</p> <pre> 1. Install Xcode: In Finder, open Applications, App Store. Search for Xcode and click Get to install the Xcode developer tools. You'll need these developer tools to run some of the command-line instructions in a Terminal window below. 2. Install XQuartz: Go to <a href="http://xquartz.macosforge.org">http://xquartz.macosforge.org</a> and download the current version of XQuartz (2.7.7 as of this writing). Open your Downloads folder, double-click on the XQuartz-2.7.7.dmg file, then double-click on the XQuartz.pkg package file and follow the instructions to complete the installation. 3. Open a Terminal (command-line) window: Go to Applications, Utilities, and double-click Terminal. Your command-line Terminal window will open. All the following commands must be typed exactly as they appear in the Terminal window, one line at a time. 4. Install Homebrew: At the Terminal command line prompt, type the following as a single full line (you may want to expand your Terminal window wider to allow it to fit, but it's okay if it wraps around). ruby -e "\$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)" then return. Homebrew is a free program that helps you install Python, Pygame, and other programs on a Mac. 5. Prepare Homebrew for use: At the Terminal prompt, type each of the following three commands exactly as shown – the second two may take a few moments to run and will show several screens of information, but keep following the steps one line at a time. echo export PATH=\$PATH:/usr/local/bin:\$PATH' &gt;&gt; ~/.bash_profile brew update brew doctor 6. Install Python 3 for Pygame: At the Terminal prompt, type: brew install python3 This will install a separate Python 3 specifically for Pygame use – this is required for all the following steps to work. 7. Install Mercurial: Still at the Terminal prompt, type: brew install mercurial Mercurial is a free source control management system that this installation of Pygame requires on a Mac. 8. Install Pygame dependencies: Pygame requires several other helper programs, called dependencies, so that it can show animations, play sounds, and create game graphics. Type the following three lines at the Terminal command prompt, hitting return after each line brew install sdl sdl_image sdl_mixer sdl_ttf portmidi brew tap homebrew/headonly brew install smpeg (NOTE 18JUL2015: Updated to reflect changes to the smpeg library; if you have any trouble here, try brew install --HEAD smpeg instead, with two dashes/hyphens before the HEAD option). Each command will take a few moments to run and display screens full of information; keep going, you're almost done... 9. Install Pygame: Type the following line at the Terminal prompt and hit return: sudo pip3 install hg+http://bitbucket.org/pygame/pygame You may have to enter an administrator password (your password, or ask an IT administrator for help at school, work, or the library), and the installation may take a few minutes. </pre>	Research & game design & setting up
Oct 3	<ul style="list-style-type: none"> <li>Another big problem: the installation process of Pygame on a mac is really complicated and requires a long time. After 3 hours of struggling, success was not found - mac's latest version is not compatible with Pygame. I may have to use another language and start all over. Time is wasted. I'm stuck and frustrated and ANGRY.</li> <li>I learned more about Python's 'file' and 'exceptions' in today's</li> </ul>	Research &

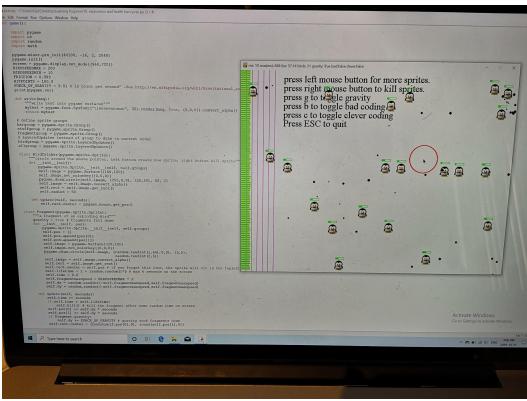
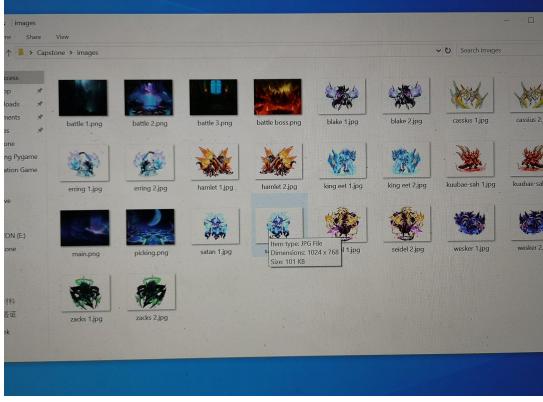
	<p>programming class. Now I can read and write files with error-detecting abilities using Python.</p> <ul style="list-style-type: none"> <li>I searched and found that the only hope lies in Windows. I have to try installing Windows 10 on my mac, otherwise I'll have to make big changes to my project.</li> <li>I successfully downloaded a 5-GB Windows 10 ISO file when a virus alert suddenly popped up saying my mac is damaged by 31%. But whatever. The rest of the downloading process will be quite long and tedious as I will do them over the next two days.</li> </ul>	setting up
Oct 4	<ul style="list-style-type: none"> <li>Yes! I finally finished downloading Windows 10 on my mac! That's some good progress right there (and it took me hours again).</li> </ul> 	Setting up
Oct 5	<ul style="list-style-type: none"> <li><b>Pygame was successfully downloaded</b> in less than 30 mins. I created a file especially for capstone where I keep everything I need for this project. I'm off to a good start.</li> </ul> 	Setting up
Oct 7	<ul style="list-style-type: none"> <li>Learned Python Classes and object oriented programming in Programming 12.</li> </ul>	Research

Oct 8	<ul style="list-style-type: none"> <li>Spent some time doing game design. In addition to the three characters I had for my previous project using Scratch, this time I will have way more characters (probably 10-ish). Also designed some new moves and buffs.</li> </ul>	Game design
Oct 9	<ul style="list-style-type: none"> <li>Learned more about Python Classes and did some exercise problems in school.</li> <li>Game design: <ul style="list-style-type: none"> <li>There will be a main stage where you can access different features of the game (including account information, shop, game rules, champion catalogue, PVP, PVE, etc.)</li> <li>PVP part will have two modes: 1 vs 1 and 6 vs 6. Winners can obtain items to upgrade their champions or get new champions. 6 vs 6 mode will be unlocked after players collect 9 champions. Players will pre-select up to 12 champions and have 30 seconds to ban 3 champions from the enemy squad and another 30 seconds to pick 6 champions from their own squad and they head into battle. Each turn players will have 15 seconds to make a move; if time-limit exceeded then no moves are made in this turn. There will also be a 1 vs 1 practice mode where a player plays 1 vs 1 against themselves with no time limit.</li> <li>PVE part will allow players to play through different challenges for each champion and unlock them; it will also feature a Gate of the Nether World with 8 levels of small boss fights and 1 big final boss fight.</li> </ul> </li> <li>I put all this information into a new document so I can keep track of my ideas.</li> </ul>	Research & game design
	 <p>The screenshot shows a Google Docs page with the title "Capstone Project Planning". The content is organized into sections:</p> <ul style="list-style-type: none"> <li><b>General game design:</b> <ul style="list-style-type: none"> <li>There will be a main stage where you can access different features (including account information, shop, game rules, champion catalogue, etc.).</li> <li>PVP part will have two modes: 1 vs 1 and 6 vs 6. Winners can obtain items to upgrade their champions or get new champions. 6 vs 6 mode will be unlocked after players collect 9 champions. Players will pre-select up to 12 champions and have 30 seconds to ban 3 champions from the enemy squad and another 30 seconds to pick 6 champions from their own squad and they head into battle. Each turn players will have 15 seconds to make a move; if time-limit exceeded then no moves are made in this turn. There will also be a 1 vs 1 practice mode where a player plays 1 vs 1 against themselves with no time limit.</li> <li>PVE part will allow players to play through different challenges for each champion and unlock them; it will also feature an Empire Tower of Inferno with 8 levels of small boss fights and 1 big final boss fight.</li> </ul> </li> <li><b>To do list:</b> <ul style="list-style-type: none"> <li>Figure out how to program a 6 vs 6 fight (identifying which opponents are allies or opponents using lists).</li> </ul> </li> </ul>	
Oct 10	<ul style="list-style-type: none"> <li>Learned zip function in Python which deals with matrix systems.</li> </ul>	Research
Oct 11	<ul style="list-style-type: none"> <li>Kept learning classes and objects (inheritance and other useful stuff)</li> <li>Realized a big problem when designing characters: HOW AM I GONNA MAKE GAME GRAPHICS AND ANIMATION?</li> </ul>	Research
Oct 12	<ul style="list-style-type: none"> <li>After thinking carefully about game graphics and searching for some</li> </ul>	Planning

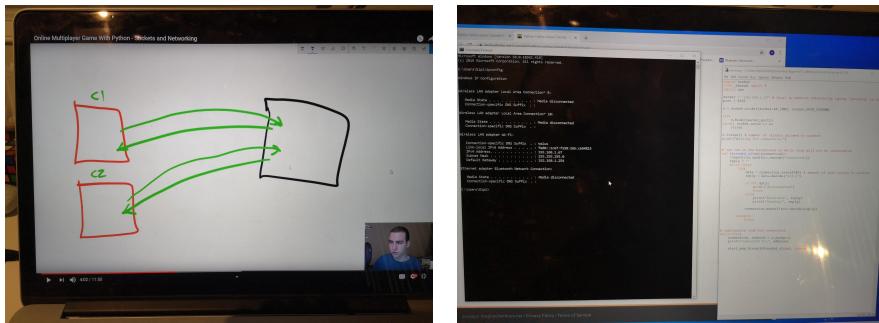
	animation tools online, I decided to ask my mentor Ms Braaten about how and to what extent this should be done.	
Oct 13	<ul style="list-style-type: none"> <li>I searched up all Pokemon elemental types for a reference and finished designing the final boss of my game!</li> </ul>  <p><b>History of the Pokémon types</b></p> <p>In <i>Pokémon Red and Blue</i>, there were 15 Pokémon types:</p> <ul style="list-style-type: none"> <li>Bug</li> <li>Dragon</li> <li>Ice</li> <li>Fighting</li> <li>Fire</li> <li>Flying</li> <li>Grass</li> <li>Ghost</li> <li>Ground</li> <li>Electric</li> <li>Normal</li> <li>Poison</li> <li>Psychic</li> <li>Rock</li> <li>Water</li> </ul> <p>With <i>Pokémon Gold and Silver</i>, two new types were added:</p> <ul style="list-style-type: none"> <li>Dark</li> <li>Steel</li> </ul> <p>Then in <i>Pokémon X and Y</i>, another type was added:</p> <ul style="list-style-type: none"> <li>Fantasy</li> </ul>	Research & game design
Oct 14	<ul style="list-style-type: none"> <li>Busy designing characters. Each character takes hours and there are more than 15 of them. There's a long way to go.</li> </ul>	Game design
Oct 15	<ul style="list-style-type: none"> <li>More character designing. Finished 5 so far, including 2 big bosses and 3 super heros.</li> </ul>	Game design
Oct 16	<ul style="list-style-type: none"> <li>Today I talked to Ms Braaten again about what I should do in terms of game graphics and animation. She recommended that I should get started with coding first and make a working version of the game, and then I can look to update the game with better graphics or other features. She taught me that <b>software development process is a constant cycle</b> that developers go through again and again. It is important to start easy in order to finish the product in time regardless of its quality, and then I can modify it from that point. <b>A good game is always improving and updating.</b></li> </ul>	Consultation with mentor
Oct 17	<ul style="list-style-type: none"> <li>More character designing. 7 done so far.</li> </ul>	Game design
Oct 18	<ul style="list-style-type: none"> <li>Finished the Class chapter and did some exercises.</li> <li>Designed two more characters.</li> </ul>	Research & game design
Oct 19	<ul style="list-style-type: none"> <li>Read some online tutorials and learned some basics about programming in Pygame. <b>Created my first Pygame program</b> printing a smiley face image onto the screen.</li> <li>More character designing. 11 done so far.</li> </ul>	Research & game design

	 	
Oct 21	<ul style="list-style-type: none"> <li>Finished designing characters. There are 14 of them and that's all I'm gonna have for now.</li> </ul>	Game design
Oct 22	<ul style="list-style-type: none"> <li>Continued learning Pygame (images &amp; colors &amp; movement).</li> </ul>  	Research
Oct 23	<ul style="list-style-type: none"> <li>Learned how to keep track of time by creating a clock object.</li> <li>Learned how to display text onto the screen and draw different shapes.</li> </ul>  	Research
Oct 24	  <ul style="list-style-type: none"> <li>Learned how to move an object based on frame and time.</li> <li>Learned how dirty rects work (blitting parts of the background that are changed per frame instead of blitting the whole background).</li> </ul>	Research

Oct 25	<ul style="list-style-type: none"> <li>Learned how to blit an animated object onto the screen.</li> <li>Learned how to load and play sound files.</li> <li>Learned how to handle key events and mouse events.</li> </ul>  	Research
Oct 28	<ul style="list-style-type: none"> <li>Revisited how to draw text and learned how to use different fonts, colors, sizes...</li> </ul> 	Research
Oct 29	<ul style="list-style-type: none"> <li>Today, I looked at the sprite portion of the tutorials and tried to understand the code but couldn't. Well, more work and more time is needed I guess. It's so hard and it's so frustrating that I couldn't understand the tutorials.</li> </ul> 	Research
Oct 30	<ul style="list-style-type: none"> <li>I spent another two hours backtracking and reading the code I was confused about yesterday. This time, I managed to understand 99% of it by consulting with pygame's official documentation, even though the mechanisms behind the Group class are still kind of misty to me. Nevertheless, that's good progress, and I feel like my brain just died. I'll move on to the next step of the tutorials tomorrow.</li> </ul>	Research

Oct 31	<ul style="list-style-type: none"> <li>Today, I looked at some advanced usage of sprites including layeredUpdates (a more complex type of groups which allows displaying sprites on different layers), explosion animations, and health bar animations. There are more than 360 lines of code in this little program from tutorials. I have read and understood more than half of the code today. I'll continue reading this program tomorrow.</li> <li>Learned how to use lambda, map, and some other flashy fancy functions in Python.</li> </ul> 	Research
Nov 1	<ul style="list-style-type: none"> <li>Managed to get through the program I wasn't able to finish yesterday. There are still parts of the tutorials I haven't looked at yet, and I will check them out later and hopefully finish reading them by next weekend.</li> <li>This weekend or during the next few days, I want to start collecting images and music for my game and start coding. So far, this project has definitely been way harder than I initially expected. I might have been a bit too ambitious at the beginning.</li> </ul>	Research
Nov 4	<ul style="list-style-type: none"> <li>Collected some images for game sprites and backgrounds.</li> </ul> 	Setting up
Nov 5	<ul style="list-style-type: none"> <li>Watched a few tutorial videos on how client and server works in a multiplayer game and how to program a multiplayer game in Python (just the basics). Learned how to obtain IP address by typing a command in the command prompt.</li> <li><b>Started game algorithm:</b> for a player vs player battle, Every champion a player owns will be an instance of a champion class. When a champion is put into a fight, they will be appended to a fighting list of champion objects (maximum 6 per side). The current champion fighting will be the</li> </ul>	Research & game algorithm

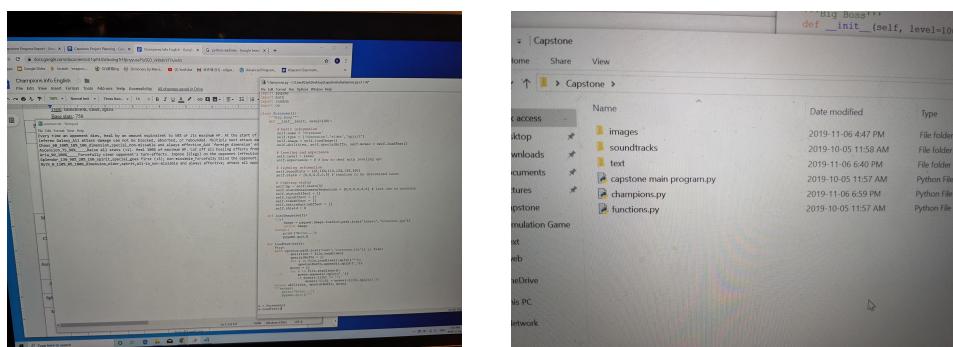
- first item of the list, and when it's subbed out it will be shuffled back and another champion will become the first item of the list.
- Champion classes will be stored in a separate program called 'champions.py' to be imported in the main program (import champions). Each champion class will look like this:
    - Class ChampName():
      - Attributes: level, image, baseStats, currentStats, statsEnhancement, turn-effects...
      - Methods: skills, abilities, animation/update/draw...
  - Above are just the initial thoughts of how to run this game and how to deal with a PVP battle.



**Nov 6**

- I consulted with Ms Braaten again. I opened up to her about the concern I had about the difficulty of the project and the possibility of an unsatisfactory result. She advised me **not to worry too much about the result but instead focus on making the process meaningful**, as I will be mostly evaluated on the work I put into the project and the new learning that occurs along the way. This has somewhat boosted my confidence coming into the most crucial stages of creating this game. I have nothing to lose. Just make the best out of it. No regrets.

Consultation with mentor & game algorithm & coding

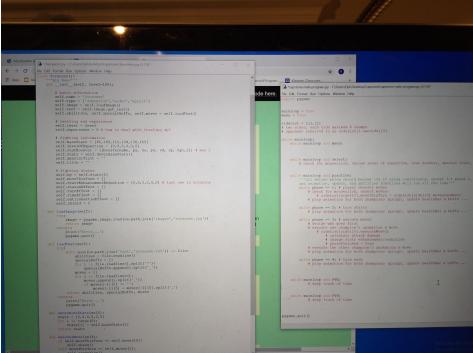
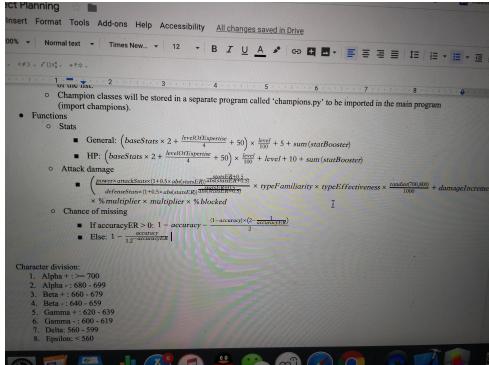
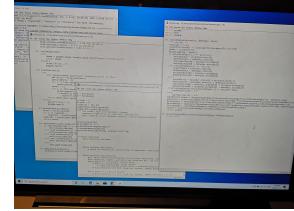
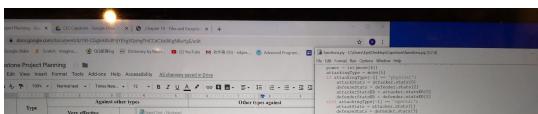


- Organized files and put champion info into a text file. Split up my project into a couple different programs: champion classes, functions, and the main program where everything else gets imported and called. Started writing code and created my first champion object - class Sorensen (the big boss of my game). It's far from being finished but it's a fine start. I'm also thinking about making each champion a child class of a general champion class since they share so many common attributes.

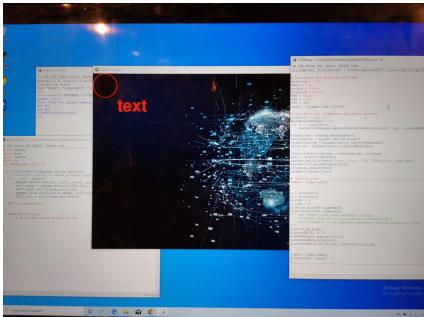
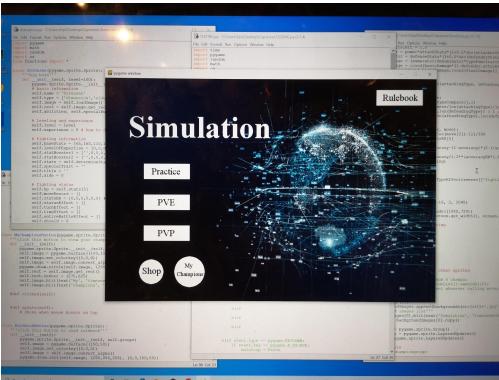
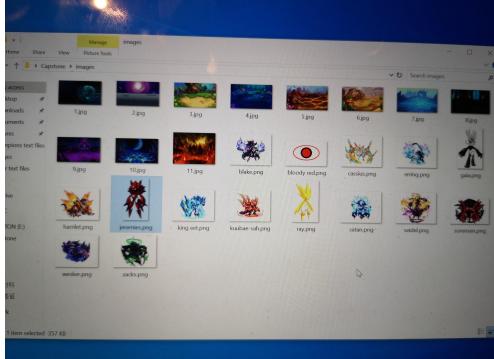
**Nov 7**

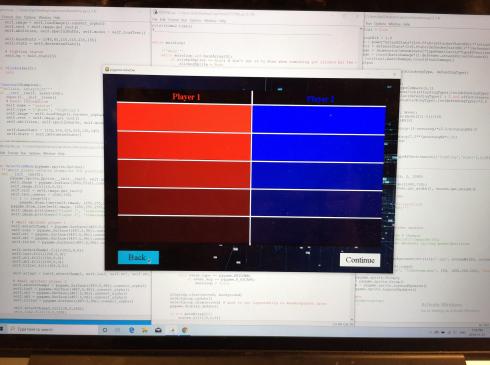
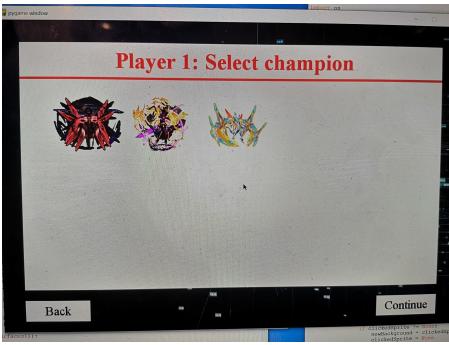
- In today's Python workshop, I learned a little about algorithms and

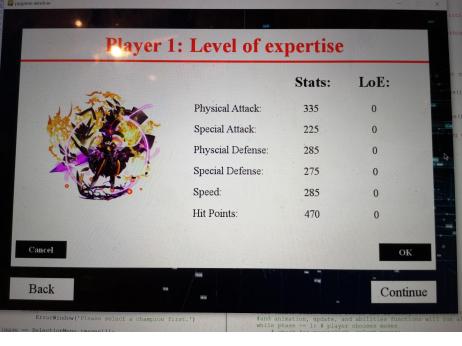
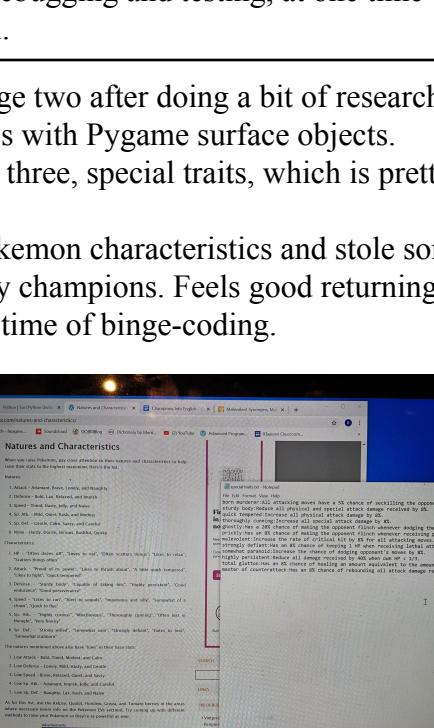
Research

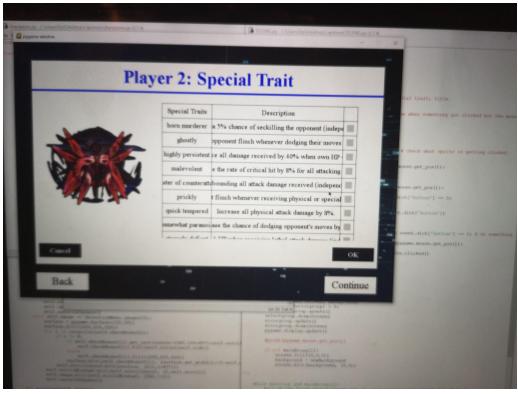
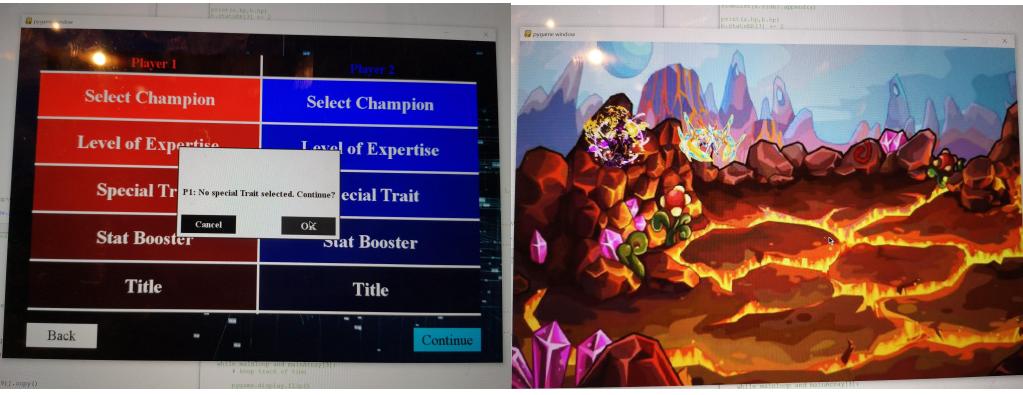
	efficiency of the code measured by the number of steps to be executed, which takes a certain amount of time.	
Nov 8	<ul style="list-style-type: none"> <li>Spent about two hours struggling to make a mainloop algorithm and typed the following code and comments. Added more attributes and functions to the champion class. Realized have to learn concurrency and threading in Pygame. Things are just getting way more complicated than I thought.</li> </ul> 	Game algorithm & coding
Nov 9	<ul style="list-style-type: none"> <li>Literally spent two and a half hours trying to come up with expressions (equations) for character stats, attack damage, and the chance of missing one move. They are harder than my math homework.</li> </ul> 	Game algorithm
Nov 11	<ul style="list-style-type: none"> <li>Coded the attack damage function, which calculates the attack damage one champion has done to another. Did a lot of tests with the current program I have made, and found it successful.</li> </ul> 	Game algorithm & coding
Nov 12	<ul style="list-style-type: none"> <li>Today, my head literally exploded. I tried to make a rather original type chart based on the one in Pokemon and programmed myself into the darkest corner. Ran into lots of problems with balancing the power of each type, writing files that represent the type chart, and printing out the type chart using a list...This is probably the most challenging problem I've managed to solve so far. Didn't expect a type chart to be this hard to make.</li> </ul> 	Game design & coding

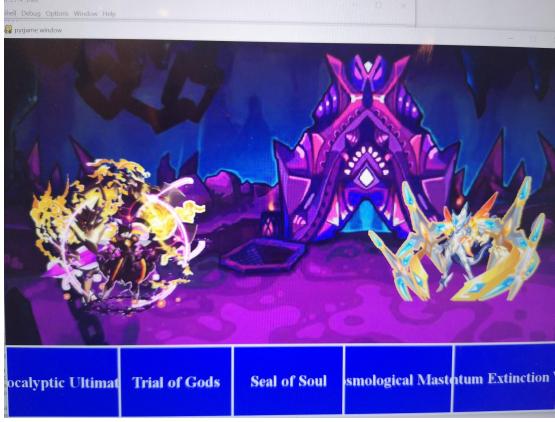
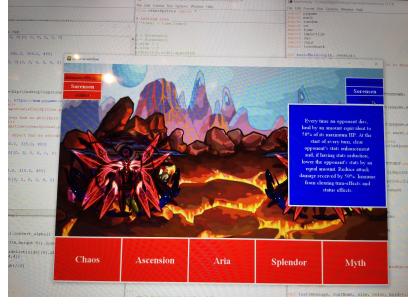
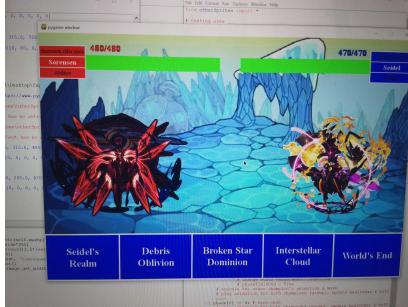
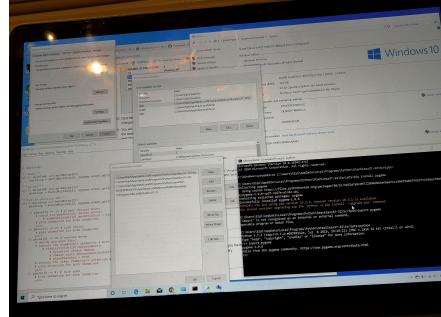
Nov 13	<ul style="list-style-type: none"> <li>Made the type functions which calculate how much damage one type does to another using the type matrix list I created yesterday. It even calculates secondary type damage like in Pokemon but with a slightly different rule.</li> </ul>		Coding
Nov 14	<ul style="list-style-type: none"> <li>Created several other functions to determine whether a move misses and whether a critical hit happens. Also started typing in Pygame related code like initializing, running the mainloop, creating a background, and keeping track of time...</li> <li>Did a bunch of testing and fixing bugs. Realized testing and debugging always go along with coding. My action steps change as I go further. Things don't always go as planned.</li> </ul>		Coding
Nov 15	<ul style="list-style-type: none"> <li>Today, I took a step back and laid out the next steps I had to take in the undertaking of this project (yes, planning's still important): <ul style="list-style-type: none"> <li>Nail all the sprites and basic game animation <ul style="list-style-type: none"> <li>Buttons to click on</li> <li>Move boxes</li> <li>HP bar &amp; other fighting information</li> <li>Actual champions</li> </ul> </li> <li>Battle algorithms and concurrency issues <ul style="list-style-type: none"> <li>Status effects</li> <li>Turn-effects, time-effects</li> <li>Ability functions</li> <li>Non-missability, moves always effective, immunity...</li> </ul> </li> <li>Add more champions and other features to battles <ul style="list-style-type: none"> <li>6 vs 6 with server</li> <li>Title, special traits, stat boosters, potions...</li> </ul> </li> <li>Modify overall game experience other than battles <ul style="list-style-type: none"> <li>Graphics and animation</li> <li>PVE levels more interesting and more PVP modes</li> <li>Allowing leveling up and catching</li> <li>Game plot or background stories of champions</li> <li>Update anything possible to make it better</li> </ul> </li> </ul> </li> </ul>		Planning
Nov 16	<ul style="list-style-type: none"> <li>Collected some more images fitting for background.</li> <li>Created another module otherSprites.py where classes of buttons can be found</li> <li>After prolonged struggles that took the whole afternoon, I managed to print a line of text on my game menu as well as an instance of a button sprite (the circle). Remember the following rules and don't mess up the order: <ul style="list-style-type: none"> <li>Assign groups first (of course) &amp; blit stuff onto background</li> <li>Then do screen.blit(background,(0,0))</li> <li>Then do allgroup.draw(screen)</li> </ul> </li> </ul>		Coding

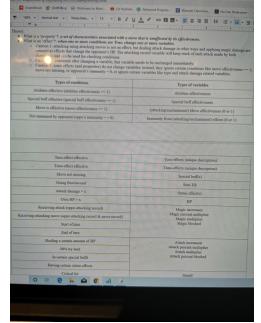
	<ul style="list-style-type: none"> <li>It literally makes me physically sick to program for the whole afternoon trying to make it work. God it's SO HARD!</li> </ul> 	
Nov 17	<ul style="list-style-type: none"> <li>So here we are. The game menu. Hurray! Buttons are all there but you can't click them right now. Still a long way ahead.</li> </ul> 	Coding
Nov 18	<ul style="list-style-type: none"> <li>Used Photoshop to deal with game graphics (champion images). Finished everything in one block. Done.</li> </ul> 	Game graphics
Nov 19	<ul style="list-style-type: none"> <li>Desperately tried to make clicking a button work but failed.</li> <li>Couldn't figure out how to make dirty-sprites work.</li> <li>Couldn't change the size of a sprite properly when mouse hovers on it.</li> </ul>	Game algorithm & coding
Nov 20	<ul style="list-style-type: none"> <li>Worked for the whole night and managed to make my buttons work.</li> <li>Put each champion into a child class of the champion class to save some unnecessary repetition.</li> <li>Need to ask my programming teacher one day how to convert a string to a function name.</li> </ul>	Game algorithm & coding

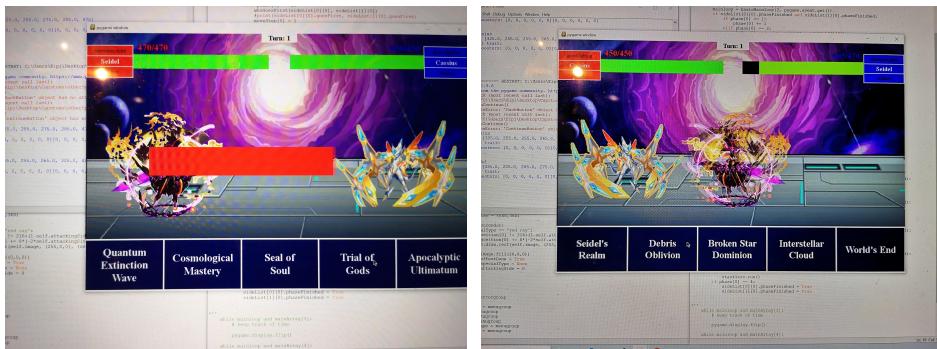
Nov 21	<ul style="list-style-type: none"> <li>Designed the select-your-champions page.</li> <li>Made two more buttons: back and continue</li> <li>Need to research how to type text onto the pygame window.</li> </ul> 	Coding
Nov 22	<ul style="list-style-type: none"> <li>Asked Ms Braaten about a technical problem and figured out how to convert a string to a function name: this is the coolest thing I've done so far in this project.</li> <li>Researched online and found a way to center text on a surface. The second coolest thing about this project maybe?</li> <li>Used very concise code to make the hovering effect for buttons on the selecting page. Avoided repetition by using lists. So cool.</li> <li>An efficient and successful day. Hope everyday will be like this one.</li> </ul>	Game algorithm and coding
Nov 24	<ul style="list-style-type: none"> <li>Still focusing on the selection menu of the practice mode, trying to allow players to select champions with personalized stats.</li> <li>Made some clicking events such as drawing a new surface covering the existing one and blitting images and texts on it. Each button will then lead the player to a different page where they can select champions or other personalized stuff.</li> </ul> 	Coding
Nov 25	<ul style="list-style-type: none"> <li>Worked for about five or six hours trying to make the 'select champion' page work. Encountered tons of problems. Not able to solve all of them. Better luck tomorrow.</li> <li>Because it is so hard to make the game work, now I'm pretty sure there's a third aspect to a successful game - <b>it has to work without bugs</b>. It's HARD to achieve that, I promise.</li> </ul>	Coding
Nov 26	<ul style="list-style-type: none"> <li>A much better day compared with yesterday. Able to solve problems efficiently.</li> </ul>	Game algorithm &

	<ul style="list-style-type: none"> <li>Finished the ‘select champion’ page with a modified algorithm and therefore better efficiency; used an in-class sideList and its copy to store temporary data using ‘cancel’ and ‘ok’ buttons.</li> <li>Designed an error window which pops up when the user does something naughty.</li> <li>Half-finished with the second page - ‘level of expertise.’</li> <li>Even had time to do some practice for next year’s Canadian Computing Competition and learned/reviewed stack and queue from last week’s Python Workshop.</li> </ul>	coding & research
		
Nov 27	<ul style="list-style-type: none"> <li>Another very productive day. Spent about four hours learning how to change the cursor to a textmaker and type numbers realtime onto the screen.</li> <li>Almost finished with the second page, the typing worked fine.</li> <li>Constantly doing debugging and testing, at one time there were 2700 lines in my Python Shell.</li> </ul>	Game algorithm & coding & research
Nov 28	<ul style="list-style-type: none"> <li>Finally finished page two after doing a bit of research on how to make a deep copy of a class with Pygame surface objects.</li> <li>Moving on to page three, special traits, which is pretty similar to Pokemon characteristics.</li> <li>So I researched Pokemon characteristics and stole some ideas to design the special traits for my champions. Feels good returning to do a bit of game design after a long time of binge-coding.</li> </ul>	Coding & research & game design
Nov 29	 <ul style="list-style-type: none"> <li>What a cool day, three hours of efficient work. (not really)</li> <li>After a decent amount of research just like yesterday, managed to create scrolling effect for the first page. This is the coolest thing I did so far (and about the only thing I did today because it's so much work and so hard).</li> </ul>	Game algorithm & coding & research

	<ul style="list-style-type: none"> <li>Ranking coolness of stuff I did in this project: 1. Scrolling effect. 2. Keyboard typing real time on screen. 3. Hovering effect for any button and the champions. 4. Deep copy! 5. Converting a string to a function name.</li> </ul>	
Nov 30	<ul style="list-style-type: none"> <li>Not the best day. Got stuck on a bunch of unexpected problems, but in the end managed to solve most of them after six hours of struggles.</li> <li>Almost finished page 3, 4, and 5, but still a long way to go. Centering a paragraph has now become a big issue I have to solve. Scrolling went nasty again but fortunately I can still wrap my head around it. We shall see next day.</li> <li>This project has been consistently challenging my limits and updating my expectations of the results. I was definitely too ambitious at the beginning. There are just so many bugs that could happen without me even noticing.</li> </ul> 	Game design, game algorithm, coding & research
Dec 2	<ul style="list-style-type: none"> <li>Designed some hard core algorithm for the continue button (with error checking).</li> <li>Drastically shortened the code in the main program by using functions for repetitive code.</li> <li>Finally made it through the selection stage. Tomorrow will be the first BATTLE day. Yeah.</li> </ul> 	Game algorithm & coding
Dec 3	<ul style="list-style-type: none"> <li><b>First day of battle coding</b>, very exciting but also very challenging.</li> <li>Modified the continue button and its functions to print out champion info before battle.</li> <li>Designed move boxes to click on. Ran into a huge bug in the middle, whole program not responding, looked frantically online for a possible</li> </ul>	Game algorithm & coding

	<p>solution, until found an elusive infinite loop.</p> <ul style="list-style-type: none"> <li>• Anyway, managed to survive and get some work done. Better luck on centering paragraphs tomorrow.</li> </ul> 	
Dec 4	<ul style="list-style-type: none"> <li>• Dealt with some hard core algorithm today: <b>made the centerParagraph function which centers a paragraph on a surface!</b> The process was unexpectedly smooth without too many bugs and everything worked out fine.</li> <li>• Then created classes that would display champions' information during battle.</li> </ul> 	Game algorithm & coding
Dec 5	<ul style="list-style-type: none"> <li>• Added health bars to the battle page. Looks good. Now the only thing left is the actual battle, the hardest part of this whole project.</li> <li>• Thought about battle algorithm and how it might work in Pygame, but didn't have a definitive answer.</li> </ul> 	Game algorithm & coding
Dec 6	<ul style="list-style-type: none"> <li>• Started learning concurrency, which will be the next big problem to tackle.</li> <li>• After installing stackless, Pygame stopped working and I literally almost had a heart attack. Searched frantically for how to make it work again and eventually re-installed Pygame using Python stackless' file path.</li> </ul> 	Coding & setting up & research

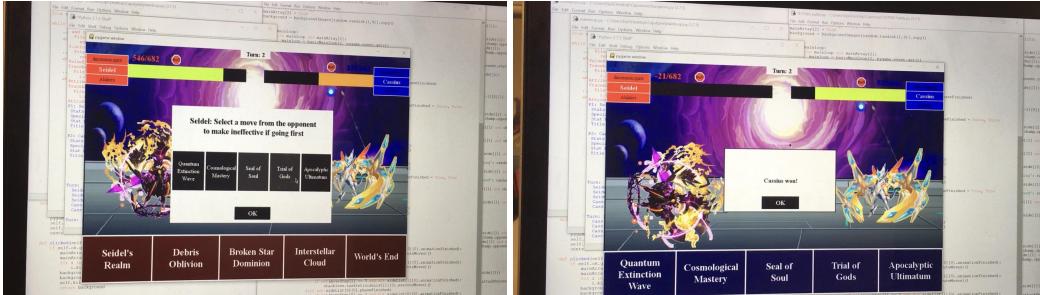
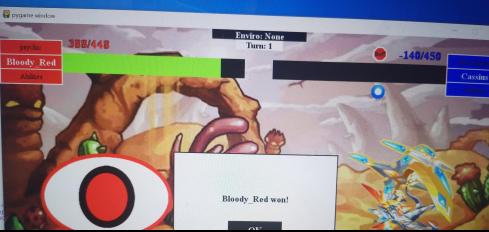
Dec 7	<ul style="list-style-type: none"> <li>Took a break from coding and thought about the logical process of a Pokemon battle.</li> <li>Found the theory behind the ‘effects’ of a move from a champion or a Pokemon and listed some of the conditions and variables I could think of to help with coding.</li> <li><b>It's very important to understand the core of the gameplay when making a game. Especially when the game is very challenging to play.</b></li> </ul> 	Game algorithm
Dec 8	<ul style="list-style-type: none"> <li>Got into probably THE hardest part of this entire project - battle algorithm and coding.</li> <li>Basically paced the room for 2.5 hours thinking about how to apply the Simulation Effect Theorem I made yesterday and the concurrency coding I learned the day before yesterday. Came up with an algorithm that might work, but needs further experiment and examination.</li> <li>Also made the whoGoesFirst function based on my Scratch game last year and tested it out. Works fine.</li> </ul>	Game algorithm & coding
Dec 9	<ul style="list-style-type: none"> <li>Did some animation algorithm and spent an unexpectedly large amount of time on it but eventually did make it work. Now the champions move up and down constantly in battle (when not executing their moves).</li> <li>Struggled to get started in translating the battle algorithm to actual code. Coded some of Seidel’s effects just to see what happens. Little to no progress really. Now I’m stuck in this abyss of extremely hard algorithm problems and only I can help myself out of it. It would take an hour to explain this to my mentor and another two hours to solve it, so I’m not gonna do that. Better luck tomorrow.</li> </ul>	Game algorithm & coding
Dec 10	<ul style="list-style-type: none"> <li>Tried to step around the hardest algorithm issues by making some relatively easy functions such as clearing stats enhancement/reduction, magic damage...</li> <li>Got started on the animation update, will continue tomorrow.</li> <li>Coded a couple moves for Seidel. Let’s see what happens next.</li> <li>It’s like rocket science at this stage. Have to be able to imagine the whole thing in my head. Have to foresee EVERY SINGLE BUG that could possibly happen.</li> </ul>	Game algorithm & coding
Dec 11	<ul style="list-style-type: none"> <li>A day like a roller coaster.</li> <li>Encountered an impossible bug which kept making my program crash. Almost in despair for the best part of the day, until after dinner when I finally discovered where the problem was - copying attributes from a Champion class. Solved it using the same method to turn a variable into a function name, though this time it’s a class name (calling child classes instead of calling the general Champion class to avoid leaving behind class methods when copying).</li> <li>Even better, I got through all the animation necessary for a battle (enchantment move, physical attack, and special attack) in less than two hours. This is a new record. I’ve never been this efficient. Everything looks great right now. Hope this is the last bug I’ll have in this program</li> </ul>	Game algorithm & coding

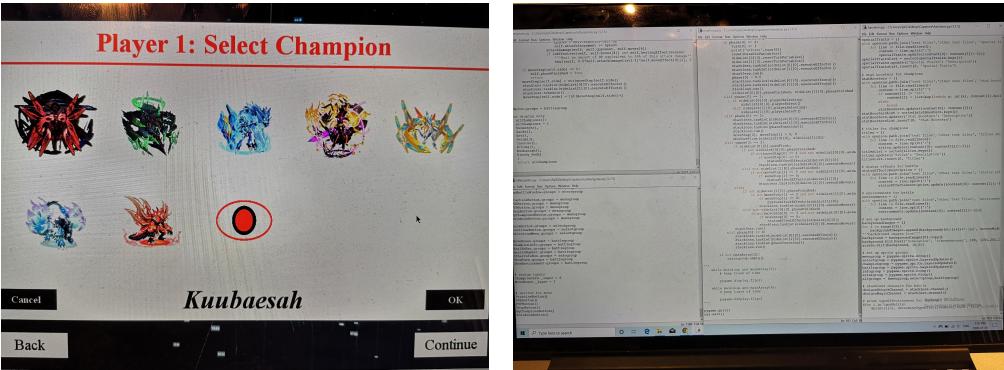


(though it's literally impossible).

Dec 12	<ul style="list-style-type: none"> <li>A smooth and productive day. Four hours well spent.</li> <li>Modified existing code by a large amount (not really).</li> <li>Created a program specialized in testing battle algorithm, which makes testing way easier and faster.</li> <li>Started creating more champion info objects which display champions' info in battle. Just managed to finish the special buffs box today. Not much went wrong and debugging was easy. Gotta meet up with my mentor soon and use Photoshop to modify the images of the little boxes I created using Scratch.</li> </ul>	Game algorithm & coding
Dec 13	<ul style="list-style-type: none"> <li>Struggled with a weird bug but managed to fix it in the end (hard to describe and it's just gonna take too many words).</li> <li>Didn't choose the fastest algorithm (using dictionaries) and had to rewrite some of the code tomorrow, which is annoying.</li> <li>Designed a little bit of status effects like those in Pokemon.</li> </ul>	Game algorithm & coding & game design
Dec 14	<ul style="list-style-type: none"> <li>A day that started well but ended badly.</li> <li>Finally settled on an optimum type of algorithm after doing quite a bit of testing and experimenting with lists and dictionaries. Yes, dictionaries for the win.</li> <li>Then struggled to make the display-information algorithm work. There were all kinds of weird bugs associated with the statsER information. Couldn't solve it. Gave up after another four hours of work. Better luck tomorrow. It's tough.</li> </ul> <p>Hey, you know what, after I typed everything above and thought it was all done and dusted, I took a shower. After I came back, I suddenly had an idea and continued coding, thinking I'd just give it a try, and it worked! That's a miracle! Everything that didn't work is now working! This is the magic of programming. <b>Never give up and think you have lost all hope, because at the next moment, you may come up with a wonderful solution!</b></p>	Game algorithm & coding
Dec 15	<ul style="list-style-type: none"> <li>Continued with yesterday's work and temporarily finished with information boxes. They work all right now.</li> </ul>	Coding

	<ul style="list-style-type: none"> <li>Next, onto execution of effects. Went through a little today. Still a long way ahead.</li> </ul> 	
Dec 16	<ul style="list-style-type: none"> <li>Used Photoshop to modify information boxes.</li> <li>Spent two hours thinking about the algorithm. There is just too much going on to put it into words. Did type some of the thoughts into the capstone planning document on Docs. Also did some experiment with stackless to know how it works.</li> </ul>	Game graphics & game algorithm
Dec 17	<ul style="list-style-type: none"> <li>Did a lot of work in three or four hours. Sort of avoided the hardest bits and did what I could.</li> <li>Filled in a lot of code for another champion, Cassius, and started testing the battle algorithm with Seidel and Cassius. Made a little progress in certain areas of the algorithm, but not much. Improved existing code where I could.</li> </ul>	Game algorithm & coding
Dec 18	<ul style="list-style-type: none"> <li>Right at the core and heart of this entire project, the hardest part and the most essential part of the most essential part - stackless concurrency. It's basically multiple functions running at the same time, switching back and forth, checking different conditions, changing variables, while making sure the same thing doesn't get executed twice. Made a little progress here, or should I say I ventured out a little today, better luck tomorrow.</li> <li>Wrote a couple more functions to deal with special cases. They seem to work fine, but need further examination. THIS IS HARD CORE ALGORITHM.</li> <li>What I concluded for the last couple of weeks' work was that, <b>creating a game is like God creating the whole world</b> - you have to know what's going on in every single corner of this world. You have to know all the physics and chemistry principles. You have to know every detail about every creature you create. You have to know exactly how many grams of dust exist in a given area. It's just this hard. You have to understand the core gameplay well enough that you can process it like a computer with perfect logic, down to the most fundamental pieces.</li> </ul>	Game algorithm & coding
<b>Dec 19</b>	<ul style="list-style-type: none"> <li>Huge breakthrough. Actually applied the concurrency algorithm and it worked fine! Although I haven't done enough testing to prove it's really working, at least it gives me lots of hope.</li> <li>Almost finished with both Seidel and Cassius! Really efficient work today. Haven't done any more work on any other day than today.</li> <li>Improved lots of functions and mini-algorithms. No big bugs. Unexpectedly smooth.</li> </ul>	Game algorithm & coding

	<ul style="list-style-type: none"> <li>One big hole left to fix in the algorithm: how to prevent the same effect from being executed twice in the same frame. Then I'll do more testing to make sure it's REALLY working. Overall, a great day. That's what I want to BELIEVE.</li> </ul>	
Dec 20	<ul style="list-style-type: none"> <li>Made some decent progress today. Possibly finished the main battle algorithm with concurrency (well, things could still go wrong while unnoticed). Just began finishing up the champions.</li> </ul>	Game algorithm & coding
Dec 21	<ul style="list-style-type: none"> <li>Almost finished with the entire battle algorithm. Modified bits of code here and there, including a big bug involving condition checking and concurrency (again, the nasty stuff).</li> <li>Finished Seidel's special effects and moved on to other features of battle</li> </ul>  <p>such as special traits and titles. Completed win/lose function, which adds an end to the battle. Glad to announce that only status effects are left to conquer. I'm getting close to making it all work now.</p>	Game algorithm & coding
Dec 22	<ul style="list-style-type: none"> <li>Finished with status effects algorithm and started on a new champion Eet. Then took a little half-day break today.</li> </ul>	Game algorithm & coding
Dec 23	<ul style="list-style-type: none"> <li>Finished making the new champion Eet. Also added background music. Everything looks so damn good right now. I don't think I can or it's necessary to get much more done before the presentation at this point. The game is in pretty good shape, although neither PVE or PVP works. I'm satisfied with what I have done so far, having written over 3000 lines of code. The next step will be to have more champions and enable the PVP mode with a server. I have decided that I will continue to make this game better even after the presentation. Just like Ms Braaten said, there is never an end to the software developing process. This is a decent start.</li> </ul>	Coding
Dec 24	<ul style="list-style-type: none"> <li>Finished the big boss Sorensen. Added environments to the battle. General modifications.</li> </ul>	Coding
Dec 25	<ul style="list-style-type: none"> <li>Finished a very interesting champion called Bloody-Red for Christmas.</li> </ul> 	Coding

Dec 28	<ul style="list-style-type: none"> <li>Made a new champion Zacks and modified some algorithms. Now I'm decently enjoying working on this project and programming as a whole.</li> </ul>	Game algorithm & coding
Dec 30	<ul style="list-style-type: none"> <li>Made yet another new champion Erring and kept modifying existing code.</li> <li>Thought a little bit about game balance and wrote down some ideas.</li> </ul>	Coding & game design
Jan 1	 <ul style="list-style-type: none"> <li>Made the eighth champion of the game - Kuubaessah. This will also be the last champion in a while. I decided not to update my game until after the presentation, which means what I currently have in my hand will be the final product for evaluation. Thinking back on the very first days of this long journey, I was even struggling to install Windows on my Macbook. Then I went through god knows how much trouble and overcame god knows how many problems just to get this far. No PVE available, no PVP or multiplayer mode, no fancy animation or storyline. Just 4484 lines of code with 8 champions and a working practice mode. That's all I have, but it already feels like more than I could ever ask for. In the following few days, I will slowly start gathering everything about this project together into a great presentation for the class. How do I feel at this exact moment? <b>Greatness is only achievable when you qualify for two things: Having outstanding talent, and having gone through pain beyond imagination.</b></li> <li>It's perhaps only fitting to close out the project on the first day of the new year. Yes, Capstone may have been finished, but the game has just started.</li> </ul>	Coding & planning
Jan 2	<ul style="list-style-type: none"> <li>It's finally time to enjoy the fruit of my own work. Played the game with my friend. He beat my Seidel 3:2 with Eet, and then I defeated his Cassius 3:1 with my Seidel. We also tried out other matchups. That was a ton of fun! The most fun I've had this year so far for sure (dad joke).</li> </ul>	Playing my own game!
Jan 3	<ul style="list-style-type: none"> <li>Found a minor bug in the algorithm and managed to fix it.</li> <li>For game balance's sake, also adjusted champions' power a little, based on the results of all the previous tests.</li> <li>Started preparing for the presentation using slides.</li> </ul>	Coding & game design & presentation preparation
Jan 6	<ul style="list-style-type: none"> <li>Prepared for presentation. Distilled my reflections and wrote down as much as possible.</li> </ul>	Presentation preparation
Jan 7	<ul style="list-style-type: none"> <li>Demonstrated the game to my mentor Ms Braaten. Told her about what worked and what didn't (because of time limit and difficulty). She appreciated my work. I'm happy.</li> </ul>	Presentation preparation

	<ul style="list-style-type: none"><li>More preparation for presentation. Read through the entire progress report, amazed by how much I did in the past three months. This is huge.</li></ul>	
Jan 13	<ul style="list-style-type: none"><li>Found another minor bug and fixed it in under 15 minutes. This has probably set the fastest record.</li><li>Finalized my presentation. Time to close out this progress report.</li></ul>	Coding & Presentation preparation