Programming booth camp

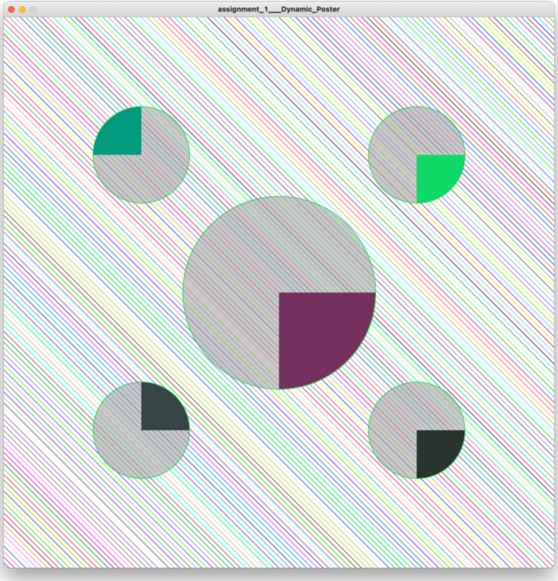
Day 1

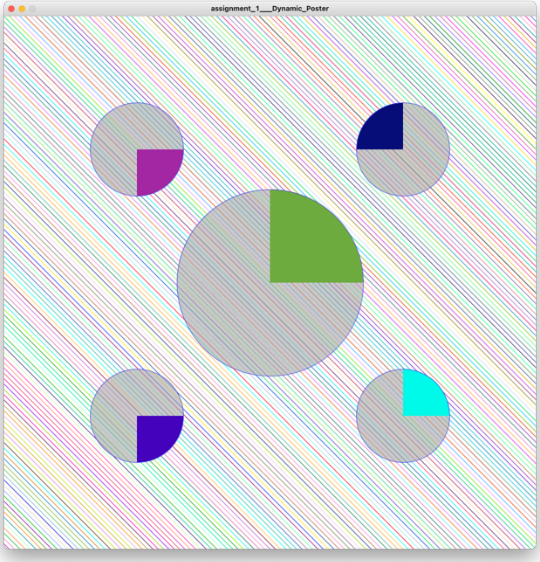
Listening

During the listening we learned how to draw shapes in processer. We learned how to do shapes, colours and about variables. The way making shapes work in processing is, you have to give it parameters. Parameters are like settings that are needed to do what you want to do. If the parameters of the shapes you have the location which are like coordinates on a X Y axis, the size of the shape, the opacity. You can also change the default colour of the shapes by using the fill function, it works by using RGB in its parameter. The opacity works the same way, but it is on a range between 0 and 255, this is because in binary, an 8-bit register allows for a max number of 256 (including 0) different combinations. We also learned about variables. Variables are like little boxes that store information. There are different types of boxes that can store their corresponding information type. Int are for whole number, float is for numbers with decimals and Boolean are for true or false. We also learned about arithmetic which is the basic maths. There are many symbols that represent different things.

Create

During the create section of the day we have to do an assignment. The assignment on day one was: “Use Processing to create a drawing of a thing or a concept which, every time you run the sketch, is different but still the same.” The hardest part was to figure out what I wanted to do. I chose to use circles where a random quarter of the circle lights up and the colour in which it lights up is also randomised.





Every element in my creation is randomized from the quarters highlighted to the colour of the lines in the background. To make my program work all you need to do it run it, there are no additional changes needed as that was the instructions given by the assignment. I used all the component from the listening part of the day, but I also added some more things that I know such as: for loops, and if statements as I have a background in programming.

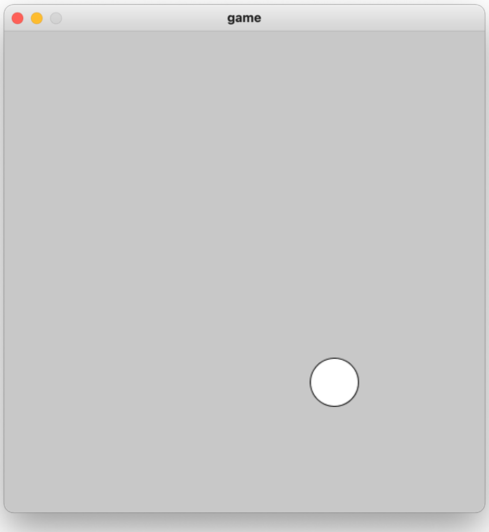
Day 2

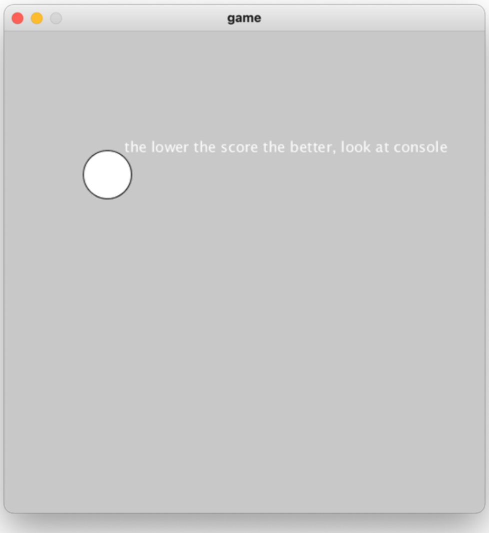
Listening

On the second day during the listening, we learned about flow and conditionals.
Flow is the order in which the program is executed, such as the game loop. A game loop is in the draw function, it repeats the program inside it at 60fps until the program is stopped. Then we learned about the if statement, > more than <less than, and more conditions. These conditionals are to create different outputs depending on the input, it also allows to repeat an action for a certain amount of time if you use counter.   We also learned how to use input to create an output. Many of these features are linked to the mouse and keyboard. mouseX will give the mouse's x value and mouseY will give you the mouse's y value.

Create

The second day's assignment was to create a game, "Use processing to create a simple game. The rules and the goal of the game are entirely up to you." I decided to try make a accuracy test with the mouse.





the lower the score the better, look at console

I had an issue with the point system, so I use the milliis function and printed it when I all the 10 dots that randomly appear where clicked. The lower the sore the better you did. I used a lot of if statements embedded in each other. I struggled to find a way to print the score once and leave it on the screen but sadly I was not able to figure it out. I tried to use the delay feature but the delay feature slows down everything and it did not do what I wanted it to do

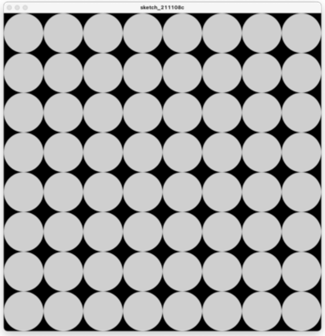
Day 3

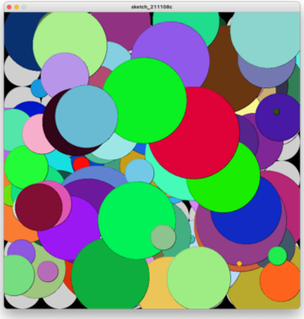
Listening

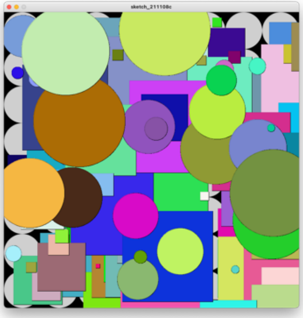
Day three we learned about loops and functions. When making a program we end up writing a lot of line that do the same thing but at different locations. To avoid that we use loops that will repeat an action many times. There is the for loop which does an action for a determined amount of time, and the while loop. The while loop does an action while a condition is not met. Both loops go through stages, the first stage is, initialize which is calling it and giving it a condition of the parameters needed. The second stage is test, which check if the condition given match with the current values or state. The third stage is action, which does the program in the loop. And the last stage is to update, which updates the variables and then goes back to the test stage. Functions are kind of pre saved programs what can be called to avoid having to rewrite the program a second time to do the same action. So you could have a function to draw a box with text. Each function can have parameters to output a different image or action.

Create

The third day's assignment was "Generate an interactive image that incorporates several dynamic patterns. Incorporate loops to create patterns and create functions for reusable code. The interactive image should react to user actions" I chose to do a randomized shape canvas. When no buttons are pressed the canvas will display a grid of grey dots. When g is pressed an infinite amount to coloured dots will randomly appear until the key is released. While pressing g, if the left mouse button is pressed the dots will become squares.







I used a function for each feature and a two-dimensional loop for the grid.

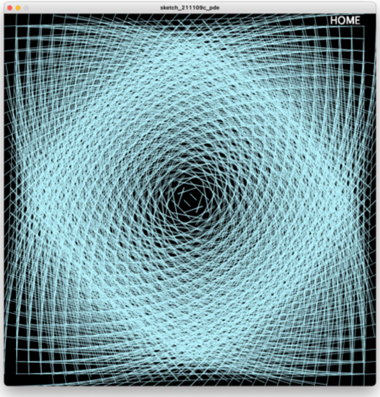
Day 4 and 5

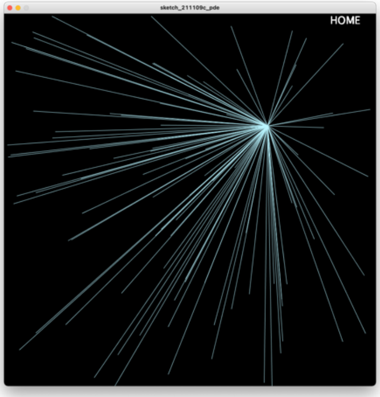
Listening

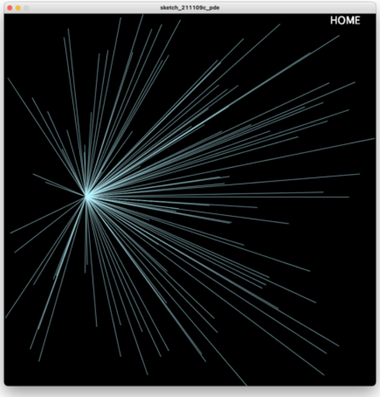
On day four we learned about arrays. Arrays are like a cabinet of variables. You can store an infinite amount of data of the same type in an array. They are very useful to enter a lot of data in variables without having to create a variable for every element. We also learned how to use them with loops to fill up an array with data.

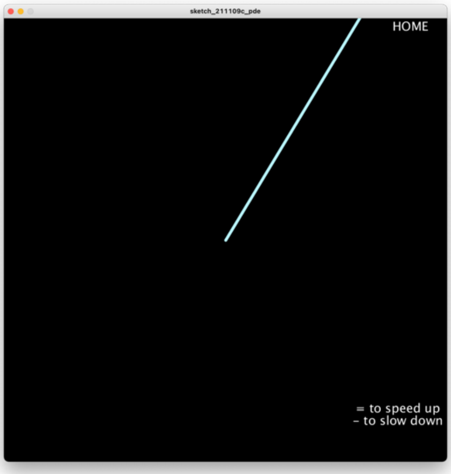
Create

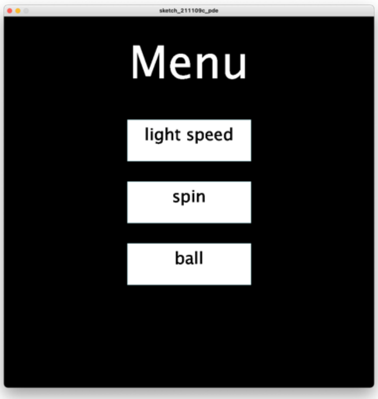
For the create section of these days, we had full control to make whatever we wanted "Combine what you have learned so far in this free assignment. Make something you are proud of and make us proud, too! " for this I decided to experiment with drawing cool designs and features that both need the user input.











I made a menu that can be controlled by the mouse, in the top of the screen there is a home button to bring you back to the menu. On the pin feature the user can speed up or slow down the line by pressing the buttons shown in the bottom right corner. My program makes use of a switch loop to select the wanted feature from the menu. Each case in the switch uses a different function and each time a function is run then the home button will revert the switch's case to the default case, which is the menu.

End of course reflection

We have reached the end of the course and this course was more of a refresh of memory rather than learning new things. This is because I have studied computer science in IB (which is before the university level) and I had done a lot of programming during my studies. The only thing I had not learned before was to draw things on the screen. I think that the way this program is structured was great. Morning we had the listening and learned new things or had a refresh of things, then we had the practice and then the assignment. I liked that we had the whole day to manage out time how we wanted, and that teacher could help us if we needed. One thing I would change is when the assignment is given. I saw that a quite a few people ended up going in circles waiting for the assignment to come out. I think the practice time should be reduce or totally removed. I understand that the practice is for practicing but I think that we could also practice while doing the assignment. This would also allow us to have more time to come up with a nicer program for the assignment. I get that having less time to work on the assignment might be to teach us how to work under pressure and to come up with ideas faster, but I think that for beginners it is better for them to have more time. With what I learned I will likely just create program for fun and develop my skills in programming, and hopefully one day use it in my actual line of work, if not I will continue make program for fun. I would also like to learn how to make games and things like that, maybe even learn how to make mods for games depending on the language used in the game or create plugins in game like Minecraft which allows their users to alter the how the game works.