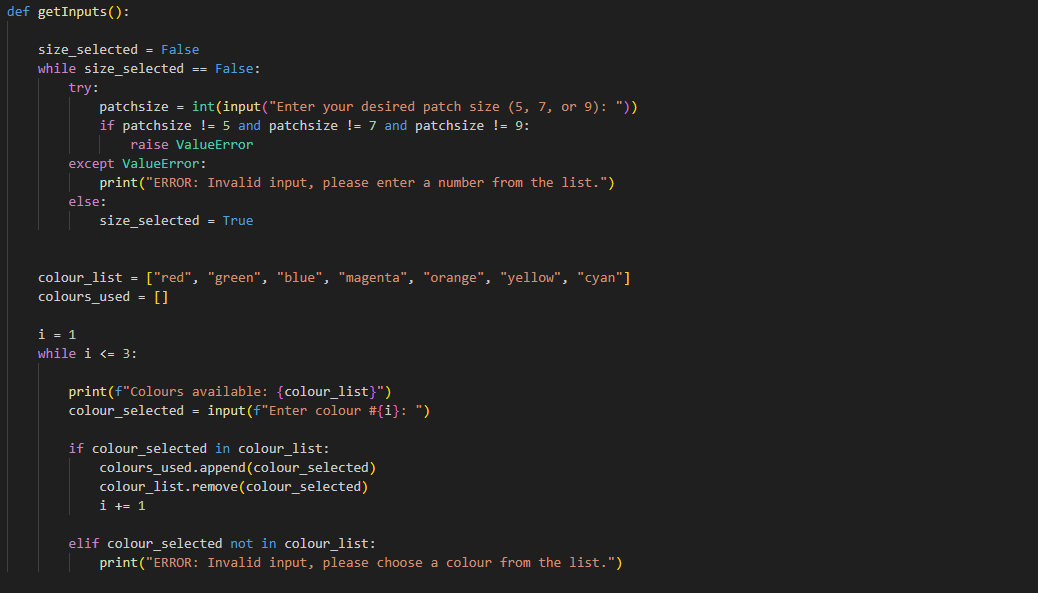
P-CW1: Patchwork Algorithm DevLog

01/12/23 – Started with a barebones top-down design to work out how I would lay out the program.

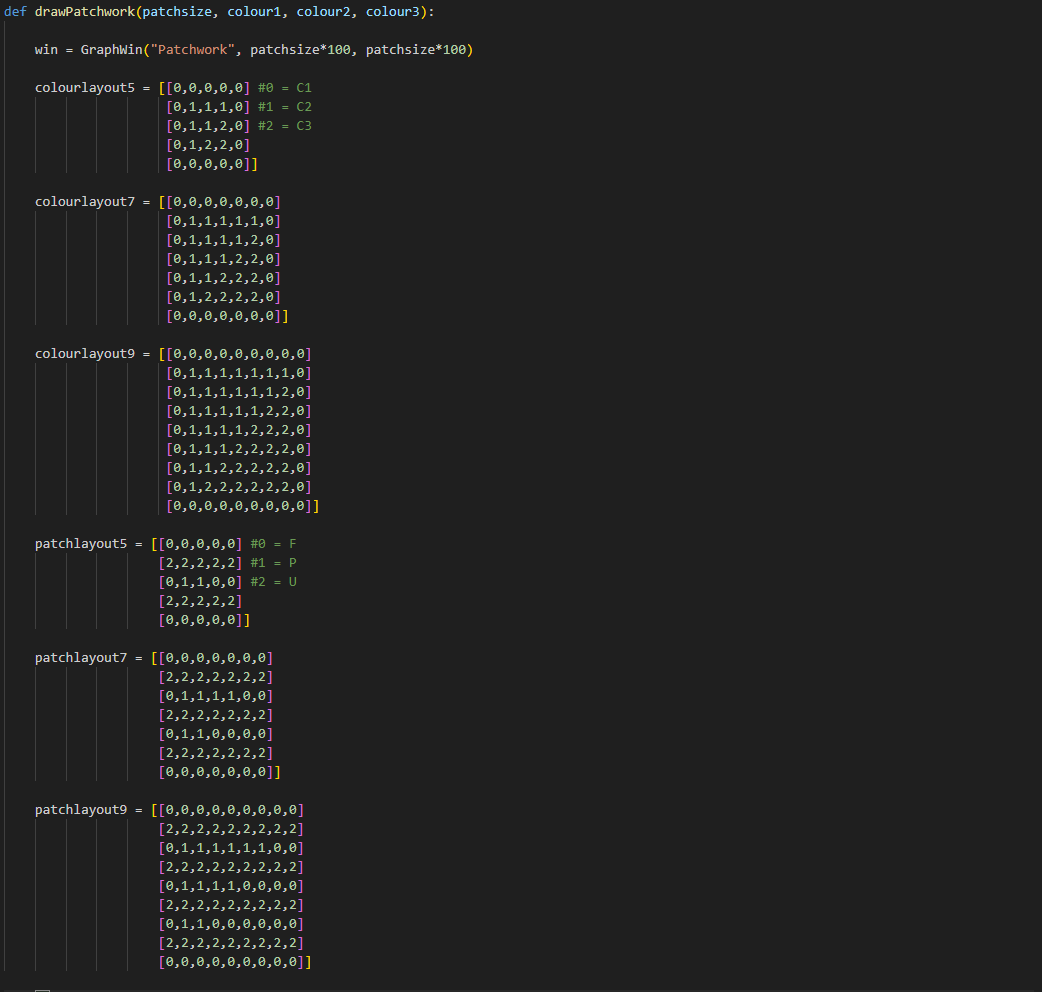


02/12/23 – Filled in the getInputs function, creating two while loops (one for patch size input, one for colour input) to catch invalid inputs. I used some error handling for patch size, so that if the user enters a string into the integer input it will throw a ValueError, and if they enter an integer that doesn’t meet the conditions of the if statement, it raises the same error. For the colours loop, I simply created a counter to make sure that 3 inputs are taken – the counter is only increased if the input is valid.

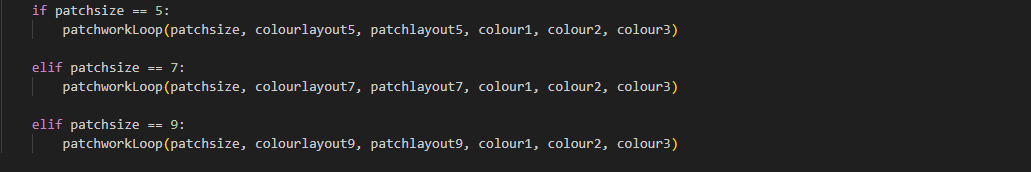


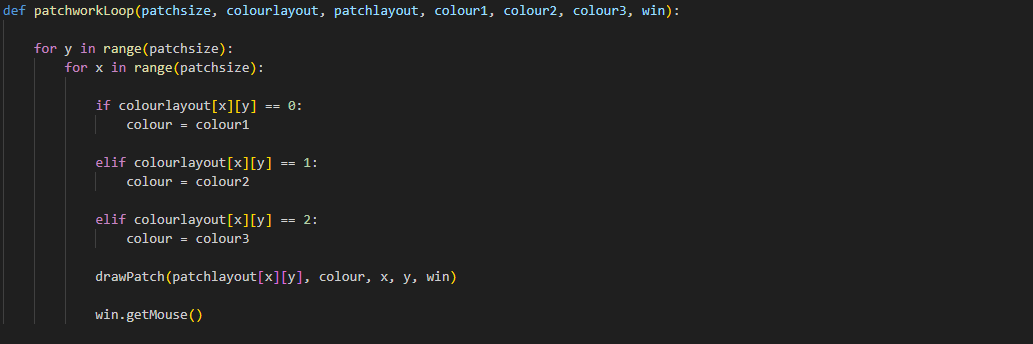
Next, I cleaned up the program by removing the setupWindow function, as it was unneccessary clutter. I then took the inputs from getInputs and called drawPatchwork with them as parameters.

Using the patterns I was assigned, I created 2d lists for each size window of both colour assignments and patch assignments – as well as initializing the graphics window.



I then created a selection clause which calls a new function patchworkLoop with varying parameters based on the given patchsize.

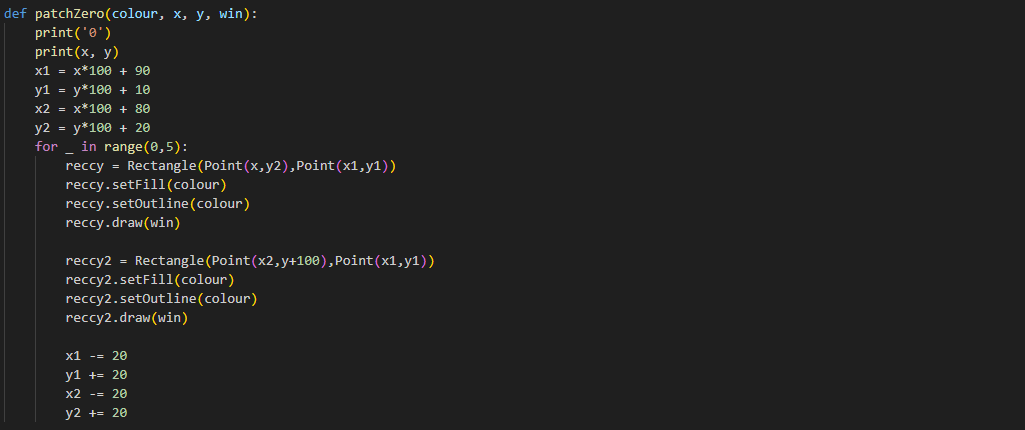


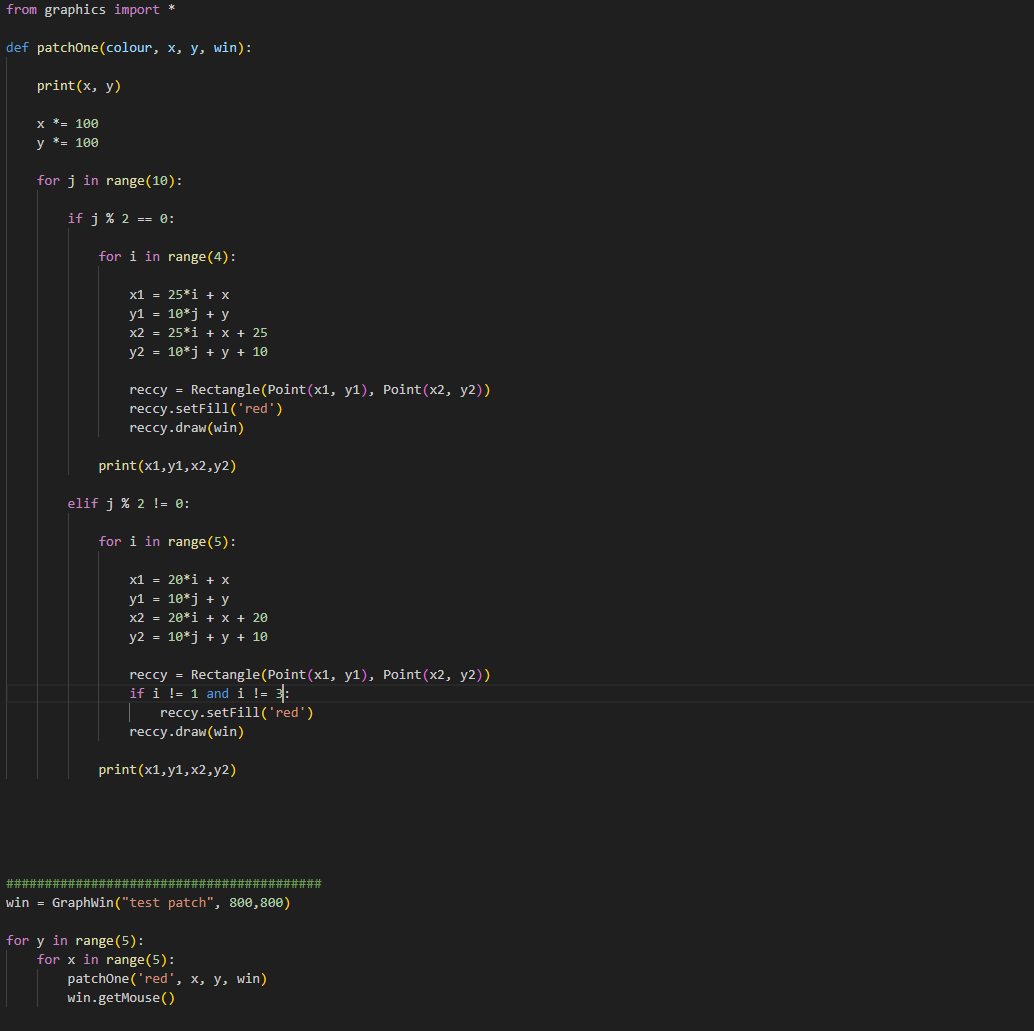
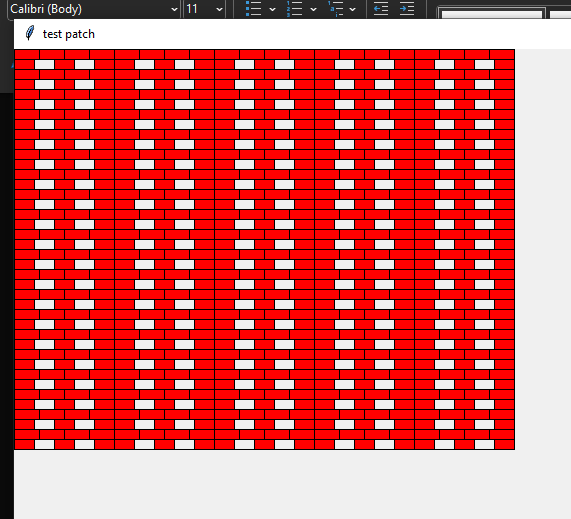
patchworkLoop iterates through a 2d grid, for each index setting the parameters for drawPatch based on the matching coordinates of colourlayout and patchlayout, and then calling it.

drawPatch calls one of 3 new functions based on the value of the parameter patchdesign.

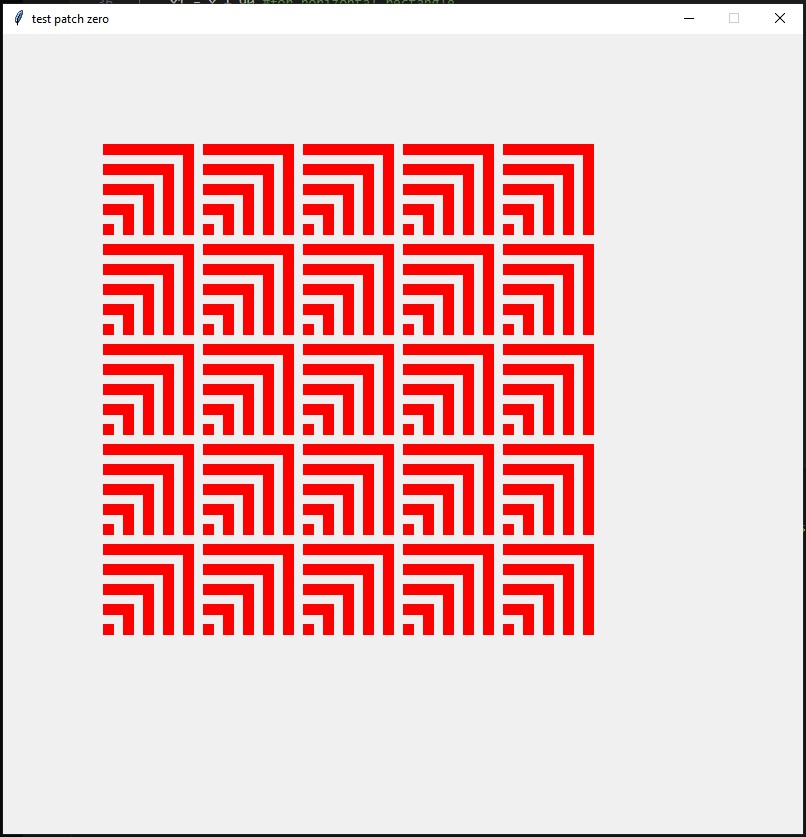


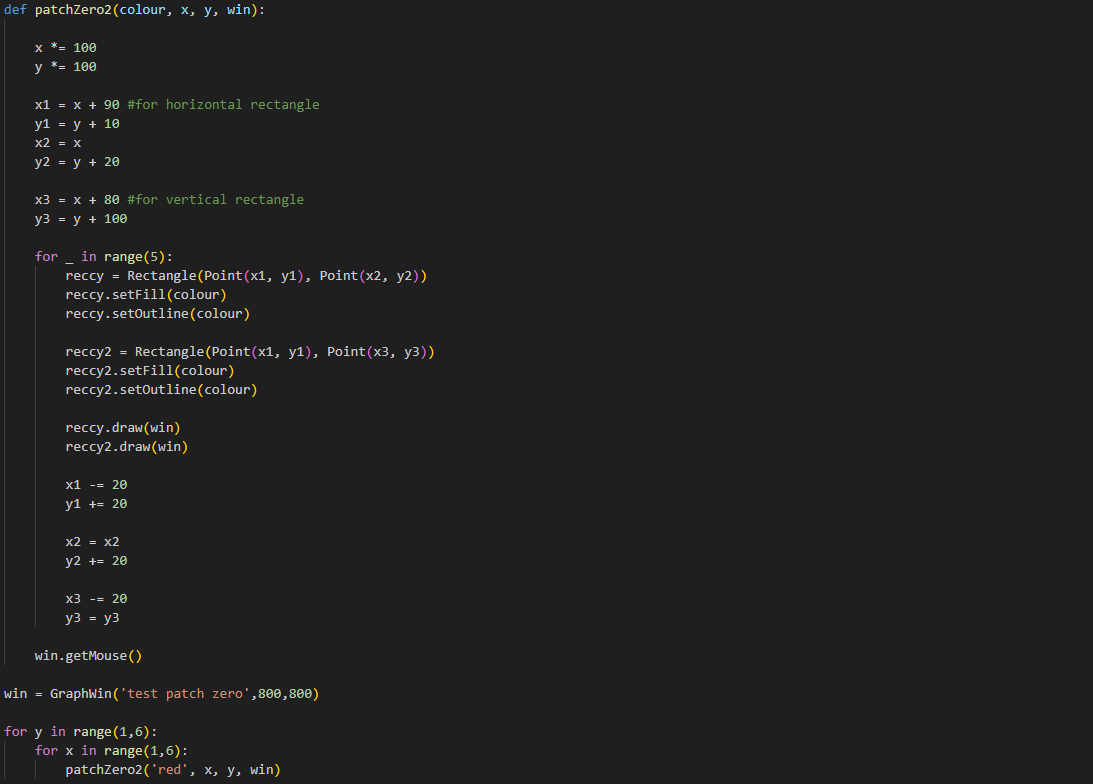
I copied in patchZero from worksheet6, and changed it slightly to fit this program’s need.



I was having trouble with patchOne, so I created a separate program to work on it in an isolated environment. I managed to get it working within this program.

Finished for today as of 18:12. Task next time is to incorporate the patchOne program back into the main coursework program.

03/12/23 – patchZero ended up being broken and janky, so I cleaned up the code in a separate program



Put this back into the main program, and now all the patches are working correctly. I realised though that the pattern of the patches was being displayed wrong – I found the error in the patchworkLoop function, as I was indexing through the 2d list for patch selection by x axis first, rather than y axis. So the patchwork was essentially being displayed at a 90 degree angle. Fixed very simply by swapping the indexes.



Strangely, the the colourlayout index calls were not affected by the backwards indexing; I tried both versions, but utlimately left it as it was at it didn’t make a difference.

The program, in its current form, is finished. May update this log for unseen errors, but for now this is the final version of the program:

