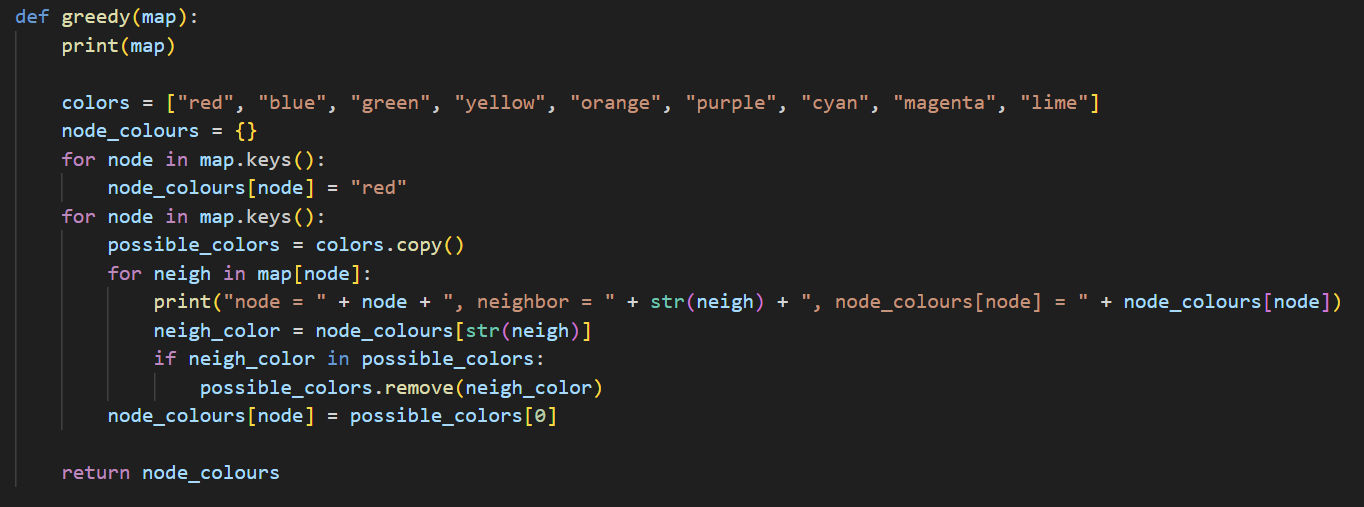
Activity 1. Graph Coloring Algorithm

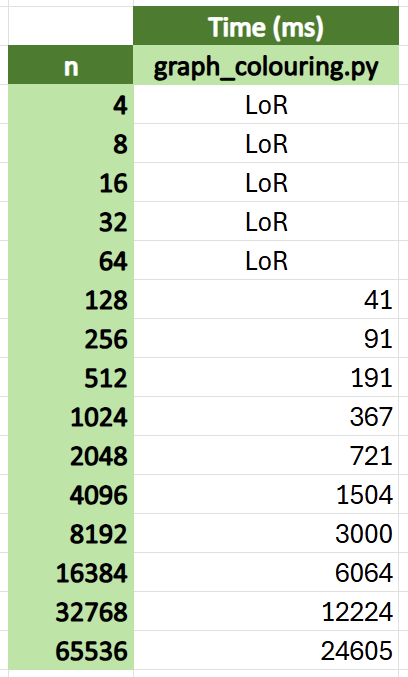
* Implement the module graph\_colouring.py, so that we can calculate a solution and visualize it with the provided Python module.

Explain the time complexity of the implemented algorithm.

It’s linear, roughly O(8n). It first sets every node’s color to red, and then it loops through them, removing the colors of their neighbors from a list of possible colors, and then setting their own as the first remaining entry in that list.



* Implement a module greedy\_times.py, using the graphs contained in sols and calculating the time it takes for the algorithm done in the previous section to solve the problem, so that the following table can be filled in.



* Does the previously calculated complexity follow the times in the table?

It does; the times exhibit linear growth. Each time the number of nodes (n) doubles, the time doubles with it.