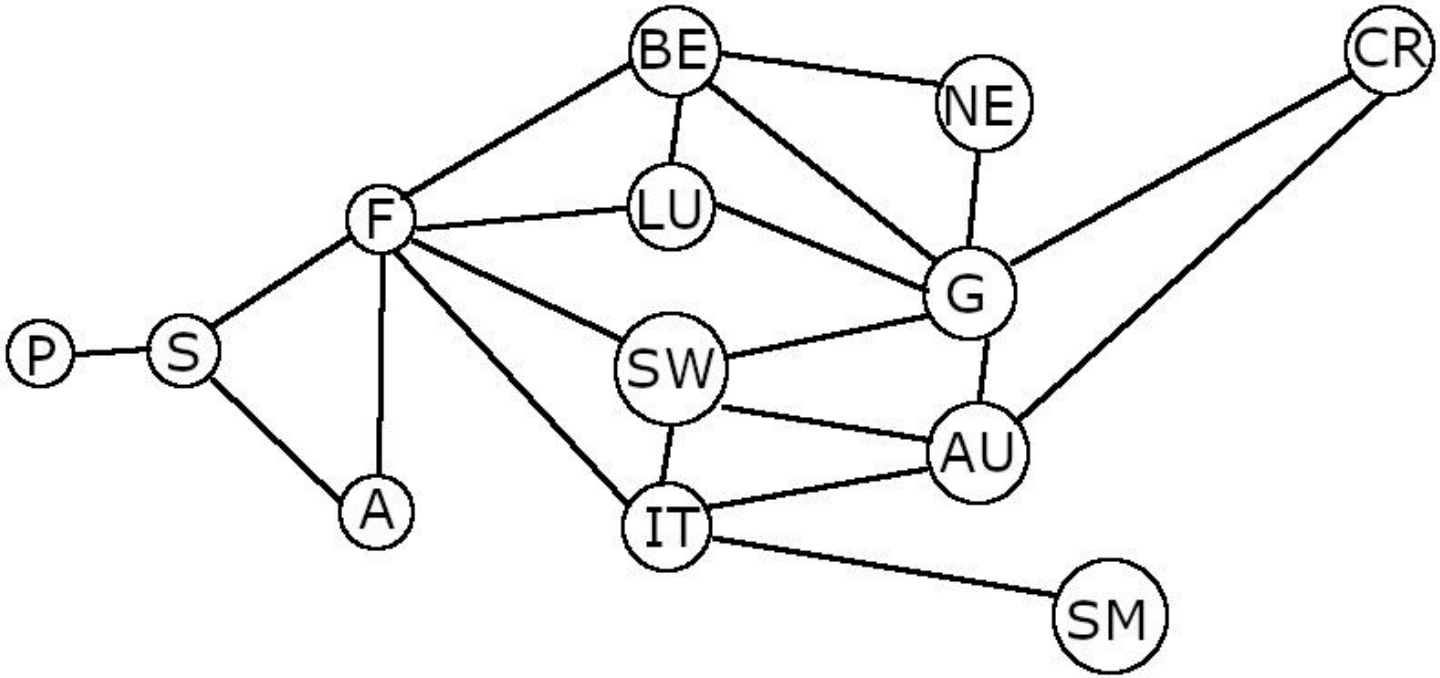
- We can colour the map using only four colours in such a way that no adjacent countries have the same colour

- To do so, we formulate the map colouring problem as graph, with nodes representing countries, and edges that represent adjacency

Q1 – Draw one such graph for the following countries:
Portugal, Spain, Andorra, France, Luxembourg, Belgium, Netherlands, Germany, Switzerland, Italy, Czech Republic, Austria, and San Marino

Be careful that your graph does not contain crossing edges!



- We can find a suitable colouring using backtracking search. Assuming that you can use the four colours ***red, green, blue, yellow***. Answer the following:

Q2 – What are the variables in this problem? What do they represent?

variables are nodes in the graph, each variable represent a country

Q3 – What is the domain of each variable?

domain for each variable are the available colors: red, green, blue, yellow

Q4 – Which variable(s) intervene in the largest number of constraints? (and how many?)

F and G invervene the most number of constraints (6)

Q5 – Which variable(s) intervene in the least number of constraints? (and how many?)

P and SM intervene least number of constraints (1)

-Recall there are rules to help you decide which variables to choose earlier during the search.

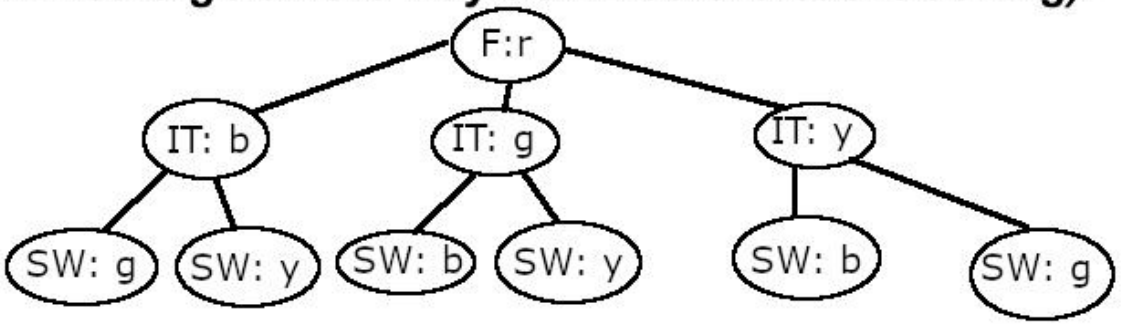
Q6 – In this graph, which variable should be chosen first?

Every node has the same number of possible value at first, since G and F will intervene the most active constraints, we could chose G or F first.

Q7 – Which variable should be chosen next?

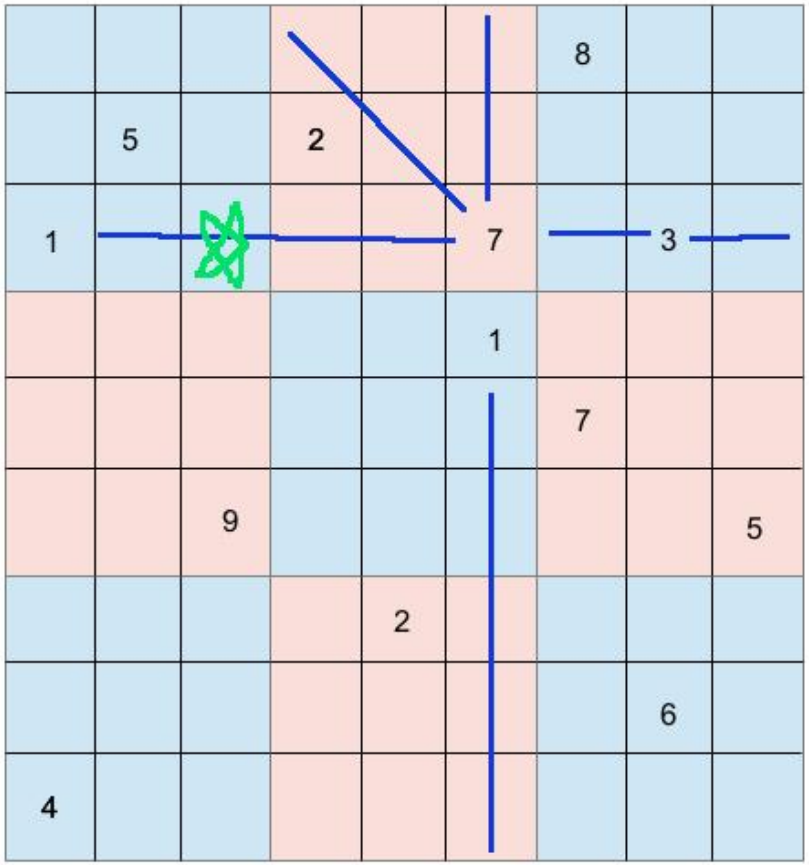
Suppose we chose F first, then in next step, S, A, BE, LU, SW, IT will have 3 possible values, and we can chose one from BE, SW, IT since they all intervene 3 active constraints

Q8 – Show the first 3 levels of the search tree with the values assigned to the variables chosen at each level (note we're not actually searching, since the search would go all the way down before backtracking).



- Sudoku is a great example of a constraint satisfaction problem.

Q7 – Draw edges on the diagram below showing which nodes are constrained by the '7' on the top-middle red quadrant.



Q8 – How many variables are there?

81 - 13 = 68 variables

Q9 - What is the upper bound on the size of the search tree for this particular Sudoku?

9^{68}

Q10 – What square should be expanded first (and why)?

Square at (3,3) should be expanded first since its one of the nodes that has fewest possible values (4) at this step.

Name: Zhang Jinming

Student number: 1000997503

Yes, we implemented all search algorithm learned in lecture in A1, and in tutorial we practiced with improvements on CSP.