Lab 9 – Task

**Maps Coloring Problem**

This problem wants to ***color*** a map using ***N colors*** such that ***no adjacent*** countries or region do have ***the same color***. For example, we want to color the following map using four colors (blue, red, yellow, green)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

| | |

| 3 | 6 |

| \_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_ |

| | | 4 | | | |

| | |\_\_\_| | | |

|\_\_| | 5 |\_\_\_\_|

| | 2 | | |

| |\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_| |

| |

| 1 |

|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|

Solve this problem by forward checking technique to solve this problem.

colors([red,green,blue,orange]).

%adjacent(Area,AdjAreas).

adjacent(1,[2,3,5,6]).

adjacent(2,[1,3,4,5,6]).

adjacent(3, [1,2,4,6]).

adjacent(4, [2,3]).

adjacent(5,[1,2,6]).

adjacent(6,[1,2,3,5]).

mapColoring(N, Solution):-

generateVariables(N, Variables),

generateDomains(N, Variables, Domains),

solveFC(Variables, Domains, Solution).

generateVariables(0,[]):-!.

generateVariables(N, Variables):-

N > 0,

NewN is N-1,

generateVariables(NewN, TmpVariables),

append(TmpVariables, [N], Variables).

%Part 1: generates(N, Variables, Domains).

solveFC([],\_,[]).

solveFC([Var|RestOfVars], [Domain|RestOfDomains], [Result|RestOfSol]):-

member(X, Domain),

Result = [Var, X],

propagateConstraints(Result, RestOfVars, RestOfDomains, NewDomains),

solveFC(RestOfVars, NewDomains, RestOfSol).

propagateConstraints(\_,[],\_,[]):-!.

propagateConstraints(AssignedVar, [Var|T1], [Domain|T2], [NewDomain|T3]):-

propagateForVar(AssignedVar, Var, Domain, NewDomain),

NewDomain \=[],

propagateConstraints(AssignedVar, T1, T2, T3).

%Part 2: propagateForVar([OldCountry,OldColor], Var, Domain, NewDomain)

**Complete the code (implement** propagateForVar **in which you will eliminate the color of an area from the domains of the areas that are adjacent to it)**