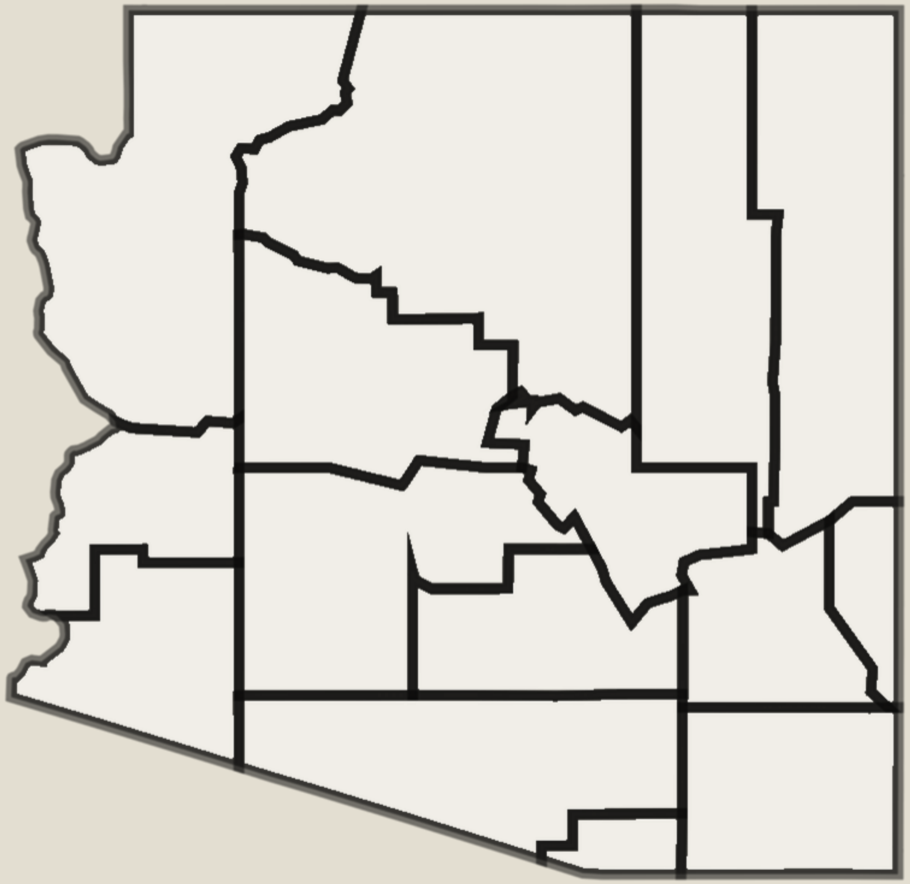
GRAPH COLORING & SUDOKU

-Paramita Adhikari (MD2014)

Problem1

- solve the map coloring puzzle
  - given a map with bordered entities (e.g. countries, states, counties) what is the minimum number of colors required to color each entity such that no two adjacent entities have the same color.



Problem2

- Solve the Sudoku puzzle

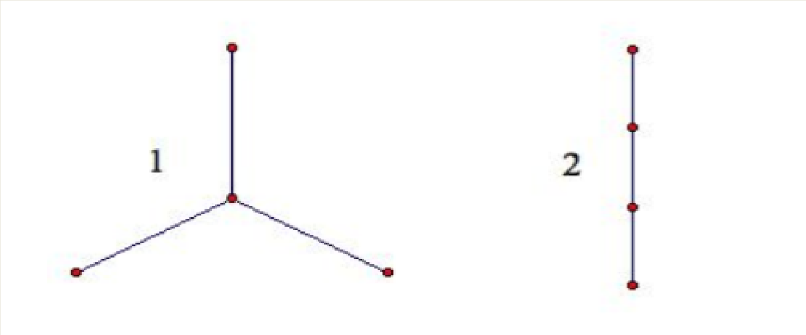
	1	7						
8				5				
2	3		4	1			7	
	8					3		9
		2		7		4		
1		3					6	
	9			3	6		5	7
				2				3
						1	9	

Some graph theory

- **Definition (Graph)**

  A (simple) graph G consists of a finite or countable vertex set $V := V$ (G) and an edge set E := E(G) ⊂ $\binom{V}{2}$

- For a vertex set V , we represent edges as (v, w), v, w ∊ V. We also write v ~ w to denote that (v, w) ∊ E and say v and w are adjacent.
- A very common pictorial representation of graphs is as follows : Vertices are represented as points on plane and edges are lines / curves between the two vertices.



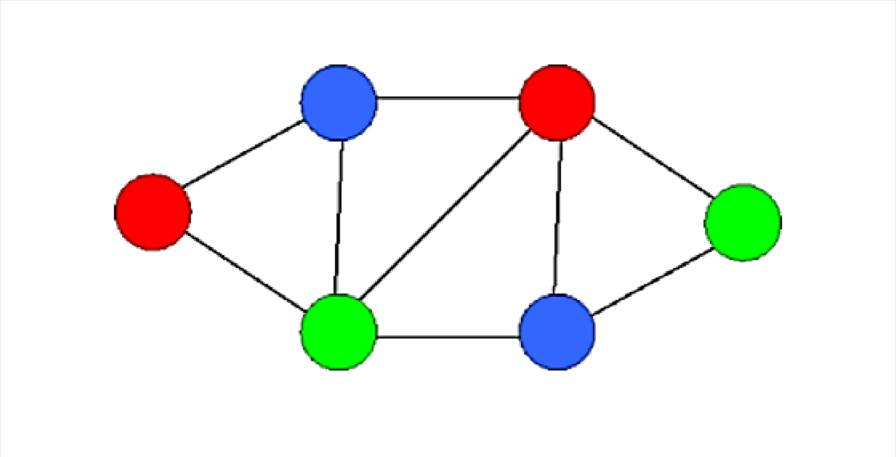
Graph Coloring

- **Definition (Vertex Coloring of a Graph)**

  A proper vertex coloring of a graph G is an assignment of colors to the vertices of the graph such that no two adjacent vertices are assigned the same color.

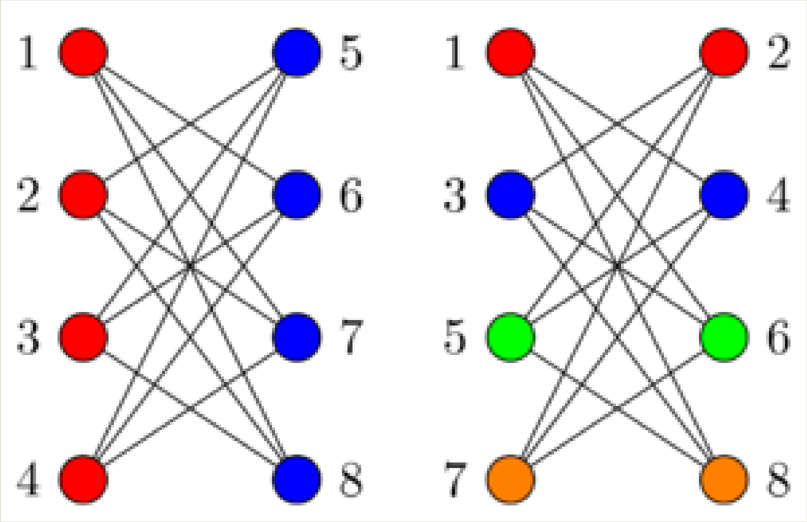
- **Definition (Chromatic Number)**

  The minimal number of colors needed to properly vertex color a graph, denoted $\chi(G)$



Greedy coloring

- Greedy coloring is a method to color a graph Label vertices v1, v2, . . . , vn
- Color each vertex with lowest color available
- There is always an ordering to give the chromatic number, but there are n! different orderings
- There exist orderings that lead to "bad" colorings
- The number of colors used greatly depends on the labeling of the vertices in a greedy coloring



Applications of Graph Coloring

- **Assigning radio frequencies**

Draw an edge between two vertices if the corresponding radio stations are too close together so that the frequencies would interfere

- **Zoos and pet stores**

Can't have some animals in the same enclosures, some fish in the same tank, etc.

- **Resource allocation**

Create a vertex for each task and connect two vertices when the corresponding tasks require the same resource

A proper coloring ensures no two tasks that require the same resource are done at the same time

Chromatic number would give the optimal way to do the tasks simultaneously

- **Sudoku puzzle solvers**

Assignment Integer Linear Program

- x_{vi} =1 if and only if vertex v is assigned color i
- w_i =1 if at least one vertex is assigned color i

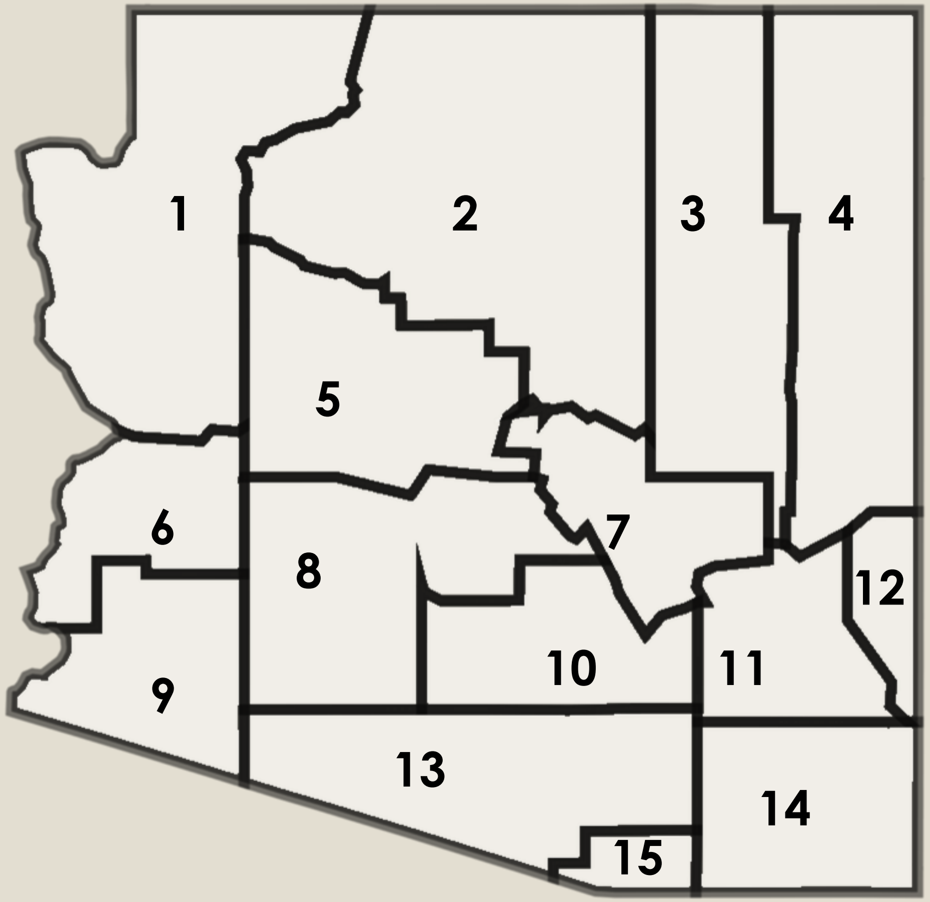
$$
\begin{array}{lll}
\text{(ASS-S)} & \min \sum_{1 \leq i \leq H} w_i & \\
\text{s.t.} & \sum_{i=1}^{H} x_{vi} = 1 & \forall v \in V \\
& x_{ui} + x_{vi} \leq w_i & \forall (u,v) \in E,\ i = 1, \ldots, H \\
& x_{vi}, w_i \in \{0,1\} & \forall v \in V,\ i = 1, \ldots, H
\end{array}
$$

Code(AMPL)

graph_coloring.mod

```
param H;
set V := 1..H;
set E within (V cross V);

var w{1..H} binary;
var x{V,1..H} binary;

#objective function that yields the chromatic number of a graph
minimize Colors: sum{i in 1..H} w[i];

#Ensures each vertex is assigned a color
subject to Assigned {i in V}:
sum{j in 1..H}x[i,j]=1;

#Ensures no two adjacent vertices are assigned the same color
subject to Edges {(i,j) in E, k in 1..H}:
x[i,k] + x[j,k] <= w[k];
```

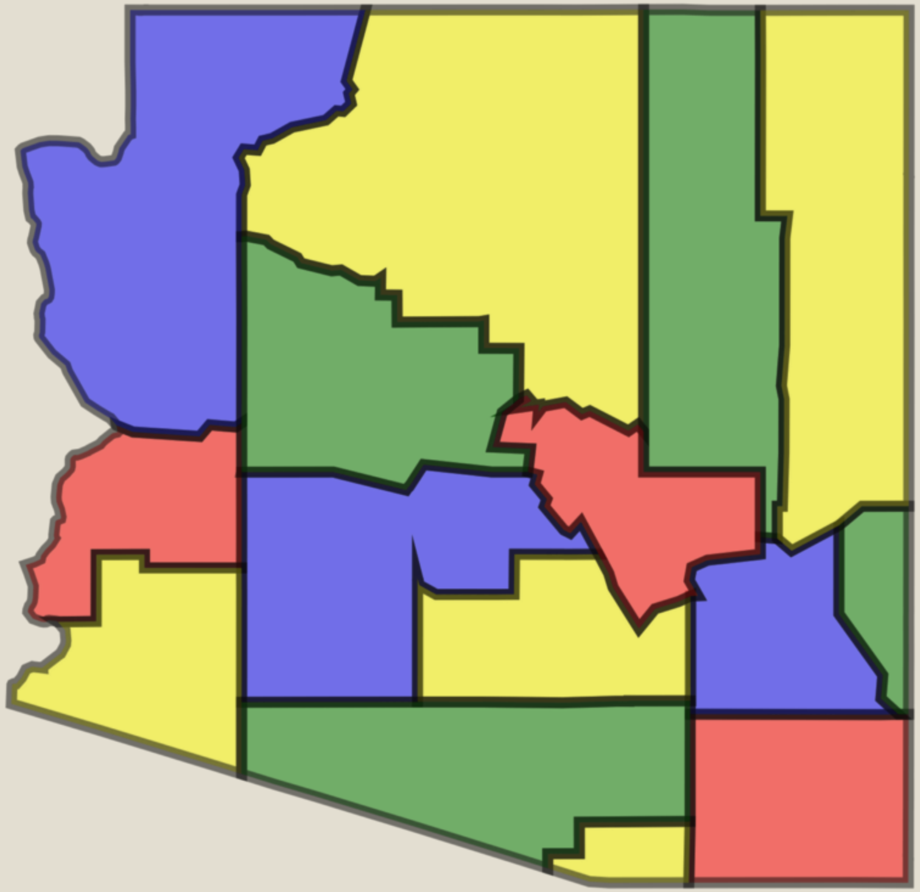
Problem1



graph_coloring.dat

```
param H := 15;
set E :=
(1,2) (1,5) (1,6)
(2,3) (2,5) (2,7)
(3,4) (3,7) (3,11)
(4,12) (5,6) (5,7)
(5,8) (6,8) (6,9)
(7,8) (7,10) (7,11)
(8,9) (8,10) (8,13)
(9,13) (10,11) (10,13)
(11,12) (11,14) (13,14)
(13,15) (14,15);
```

Solution



```
x [*,*]
:    1   2   3   4   5   6   7   8   9   10  11  12  13  14  15    :=
1    1   0   0   0   0   0   0   0   0   0   0   0   0   0   0
2    0   1   0   0   0   0   0   0   0   0   0   0   0   0   0
3    1   0   0   0   0   0   0   0   0   0   0   0   0   0   0
4    0   1   0   0   0   0   0   0   0   0   0   0   0   0   0
5    0   0   1   0   0   0   0   0   0   0   0   0   0   0   0
6    0   1   0   0   0   0   0   0   0   0   0   0   0   0   0
7    0   0   0   1   0   0   0   0   0   0   0   0   0   0   0
8    1   0   0   0   0   0   0   0   0   0   0   0   0   0   0
9    0   0   1   0   0   0   0   0   0   0   0   0   0   0   0
10   0   1   0   0   0   0   0   0   0   0   0   0   0   0   0
11   0   0   1   0   0   0   0   0   0   0   0   0   0   0   0
12   1   0   0   0   0   0   0   0   0   0   0   0   0   0   0
13   0   0   0   1   0   0   0   0   0   0   0   0   0   0   0
14   1   0   0   0   0   0   0   0   0   0   0   0   0   0   0
15   0   1   0   0   0   0   0   0   0   0   0   0   0   0   0
;

Colors = 4
```

Remarks

- **Four color theorem:**
  - given any separation of a plane into contiguous regions, producing a figure called a *map*, no more than four colors are required to color the regions of the map so that no two adjacent regions have the same color.
  - The four color theorem was proved in 1976 by Kenneth Appel and Wolfgang Haken after many false proofs and counterexamples

Sudoku

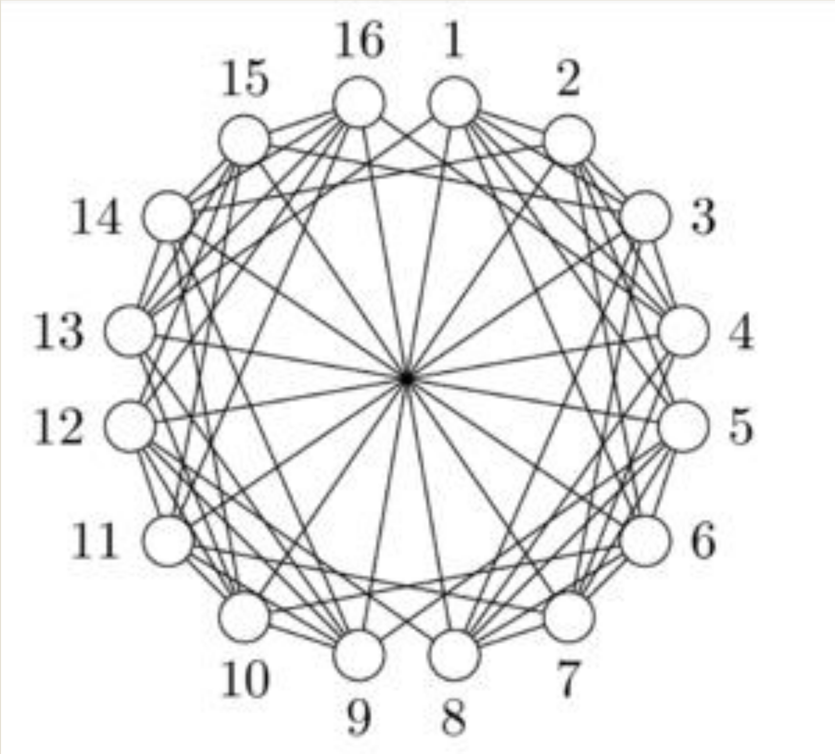
- The word Sudoku is a Japanese abbreviation for the phrase, “*suji wa dokushin ni kagiru*” which translates as the “the digits must remain single.”
- The puzzle consists of a 9×9 grid in which some of the entries of the grid have a number from 1 to 9.
- Filling the table with the numbers must follow these rules:
  - Numbers in rows are not repeated
  - Numbers in columns are not repeated
  - Numbers in 3 × 3 blocks are not repeated
  - Order of the numbers when filling is not important

8	3	4	6	7	1	9	2	5
1	2	5	8	3	9	6	4	7
7	9	6	5	2	4	3	1	8
9	5	7	3	1	8	4	6	2
2	4	1	9	5	6	8	7	3
3	6	8	2	4	7	5	9	1
6	8	2	7	9	3	1	5	4
5	1	9	4	8	2	7	3	6
4	7	3	1	6	5	2	8	9

Sudoku as a Graph Coloring Problem

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

- We consider a 4x4 Sudoku puzzle.
- Label grid squares in Sudoku puzzle and construct corresponding graph by having two vertices be adjacent if:
  - they are in the same row
  - they are in the same column
  - they are in the same 2x2 subgrid
- A proper coloring of the graph translates to a solution to the Sudoku puzzle.



Binary integer linear program (BILP) for general n × n puzzles

- In general, any $n \times n$ game can be created, where $n = m^2$ and m is any positive integer.
- In the BILP, the objective function is kept arbitrary. However, the constraints are important to follow to rules in order to find a feasible solution.

Define:

$$x_{ijk} = \begin{cases} 1, & \text{if element } (i,j) \text{ of the } n \times n \text{ Sudoku matrix contains the integer } k \\ 0, & \text{otherwise.} \end{cases}$$

$$\begin{aligned}
\min \quad & \mathbf{0}^T \mathbf{x} \\
s.t. \quad & \sum_{i=1}^{n} x_{ijk} = 1, \quad j{=}1{:}n,\ k{=}1{:}n \quad \text{(only one } k \text{ in each column)} && (1) \\
& \sum_{j=1}^{n} x_{ijk} = 1, \quad i{=}1{:}n,\ k{=}1{:}n \quad \text{(only one } k \text{ in each row)} && (2) \\
& \sum_{j=mq-m+1}^{mq} \sum_{i=mp-m+1}^{mp} x_{ijk} = 1, \quad k{=}1{:}n,\ p{=}1{:}m,\ q{=}1{:}m \quad \text{(only one } k \text{ in each submatrix)} && (3) \\
& \sum_{k=1}^{n} x_{ijk} = 1 \quad i{=}1{:}n,\ j{=}1{:}n \quad \text{(every position in matrix must be filled)} && (4) \\
& x_{ijk} = 1 \quad \forall (i,j,k) \in G \quad \text{(given elements } G \text{ in matrix are set "on")} && (5) \\
& x_{ijk} \in \{0,1\} && (6)
\end{aligned}$$

Code(AMPL)

sudoku.mod

```
set N := 1..9 ;
set G within {N cross N cross N};
var x {( i,j,k ) in {N cross N cross N}} binary ;

minimize nothing : x[1 ,1 ,1];

subject to Columns {j in N, k in N}:
    sum{i in N}x[i,j,k]=1;

subject to Rows {i in N, k in N}:
    sum{j in N}x[i,j,k]=1;

subject to Squares{k in N, p in 1..3 ,q in 1..3}:
    sum{j in (3*p -2) ..(3*p) } sum{i in (3*q -2) ..(3*q) }x[i,j,k]=1;

subject to all_filled{i in N, j in N}:
    sum{k in N}x[i,j,k]=1;

subject to known {(i,j,k) in G }:x[i,j,k]=1;
```

Problem2

	1	7						
8				5				
2	3		4	1			7	
	8					3		9
		2		7		4		
1		3					6	
	9			3	6		5	7
				2				3
						1	9	

sudoku.dat

```
set G:=
(1,2,1)  (1,3,7)  (2,1,8)
(2,5,5)  (3,1,2)  (3,2,3)
(3,4,4)  (3,5,1)  (3,8,7)
(4,2,8)  (4,7,3)  (4,9,9)
(5,3,2)  (5,5,7)  (5,7,4)
(6,1,1)  (6,3,3)  (6,8,6)
(7,2,9)  (7,5,3)  (7,6,6)
(7,8,5)  (7,9,7)  (8,5,2)
(8,9,3)  (9,7,1)  (9,8,9);
```

Solution

5	1	7	6	8	3	9	2	4
8	4	9	2	5	7	6	3	1
2	3	6	4	1	9	5	7	8
7	8	4	5	6	2	3	1	9
9	6	2	3	7	1	4	8	5
1	5	3	8	9	4	7	6	2
4	9	8	1	3	6	2	5	7
6	7	1	9	2	5	8	4	3
3	2	5	7	4	8	1	9	6

```
x [1,*,*]
:   1   2   3   4   5   6   7   8   9    :=
1   0   0   0   0   1   0   0   0   0
2   1   0   0   0   0   0   0   0   0
3   0   0   0   0   0   0   1   0   0
4   0   0   0   0   0   1   0   0   0
5   0   0   0   0   0   0   0   1   0
6   0   0   1   0   0   0   0   0   0
7   0   0   0   0   0   0   0   0   1
8   0   1   0   0   0   0   0   0   0
9   0   0   0   1   0   0   0   0   0

 [2,*,*]
:   1   2   3   4   5   6   7   8   9    :=
1   0   0   0   0   0   0   0   1   0
2   0   0   0   1   0   0   0   0   0
3   0   0   0   0   0   0   0   0   1
4   0   1   0   0   0   0   0   0   0
5   0   0   0   0   1   0   0   0   0
6   0   0   0   0   0   0   1   0   0
7   0   0   0   0   0   1   0   0   0
8   0   0   1   0   0   0   0   0   0
9   1   0   0   0   0   0   0   0   0

 [3,*,*]
:   1   2   3   4   5   6   7   8   9    :=
1   0   1   0   0   0   0   0   0   0
2   0   0   1   0   0   0   0   0   0
3   0   0   0   0   0   1   0   0   0
4   0   0   0   1   0   0   0   0   0
5   1   0   0   0   0   0   0   0   0
6   0   0   0   0   0   0   0   0   1
7   0   0   0   0   1   0   0   0   0
8   0   0   0   0   0   0   1   0   0
9   0   0   0   0   0   0   0   1   0
```

Bibliography:

- Colour Maps using Integer Programming by Mohamed Leila
- Wikipedia-Four color theorem
- An Integer Programming Model for the Sudoku Problem Andrew C. Bartlett Amy N. Langville March 18, 2006
- An Integer Linear Programming Approach to Graph Coloring by Megan Duff and Rebecca Robinson
- Solving sudoku as an Integer Programming problem by Olszowy Wiktor
- Wikipedia- AMPL (A Mathematical Programming Language)
- MathWorks

Thank You