Ibrahim Alizada - AI Project 2

The task is to solve graph coloring problem using CSP (Constraint Search Problem) along with heuristics (MRV, LCV) and AC3 functions.

Python was used in implementation of the task.

The ***backtracking_search*** function is called and it calls ***backtracking*** function to solve the problem recursively.
select_next_vertex -> MRV (minimum remaining values) function is used to choose the next vertex. It chooses the node that has fewest values in its domain.
order_colors -> LCV (least constraining values) function returns back the domain list with ascending order of the particular vertex's constraining value.
is_safe -> checks whether there is color collision within adjacent vertices or not.
AC3_inference -> AC3 heuristic - checks whether there is arc consistency or not for vertices.
Revise -> inconsistent values removed from domain list. If the result coming from revise is True (any value removed), then it updates deque by appending the neighbors of next vertices to deque.
Checktruth -> this function just checks the correctness of the program by comparing colors of adjacent vertices.

How to run my script : I implemented the task in Python using Jupyter Notebook.
It accepts 1 input: Input file -> Example: input1.txt

It reads the input file entered and gives back the results for the problem.

If there is no solution, the program return "It is not possible". Otherwise, print vertex number with corresponding color.

```
Enter file name : input4.txt

It is possible!

0 'th vertex ->  blue
1 'th vertex ->  red
2 'th vertex ->  red
3 'th vertex ->  green
4 'th vertex ->  green
5 'th vertex ->  black
6 'th vertex ->  black
7 'th vertex ->  blue
8 'th vertex ->  red
9 'th vertex ->  red
10 'th vertex ->  green
11 'th vertex ->  black
12 'th vertex ->  green
13 'th vertex ->  red
14 'th vertex ->  blue
15 'th vertex ->  green
16 'th vertex ->  black

 No match! Done!
```