EXPERIMENT – 11 Map Colouring (CSP backtracking)

AIM:

Colour graph with k colours such that neighbors differ.

Code:

# map\_coloring.py

def color\_graph(graph, colors):

assignment={}

def ok(node,c):

for nb in graph.get(node,[]):

if nb in assignment and assignment[nb]==c: return False

return True

def backtrack(nodes):

if not nodes: return True

node=nodes[0]

for c in colors:

if ok(node,c):

assignment[node]=c

if backtrack(nodes[1:]): return True

del assignment[node]

return False

nodes=list(graph.keys())

if backtrack(nodes): return assignment

return None

if \_\_name\_\_=='\_\_main\_\_':

g={'WA':['NT','SA'],'NT':['WA','SA','Q'],'SA':['WA','NT','Q','NSW','V'],'Q':['NT','SA','NSW'],

'NSW':['Q','SA','V'],'V':['SA','NSW'],'T':[]}

print(color\_graph(g,['red','green','blue']))

OUTPUT:

