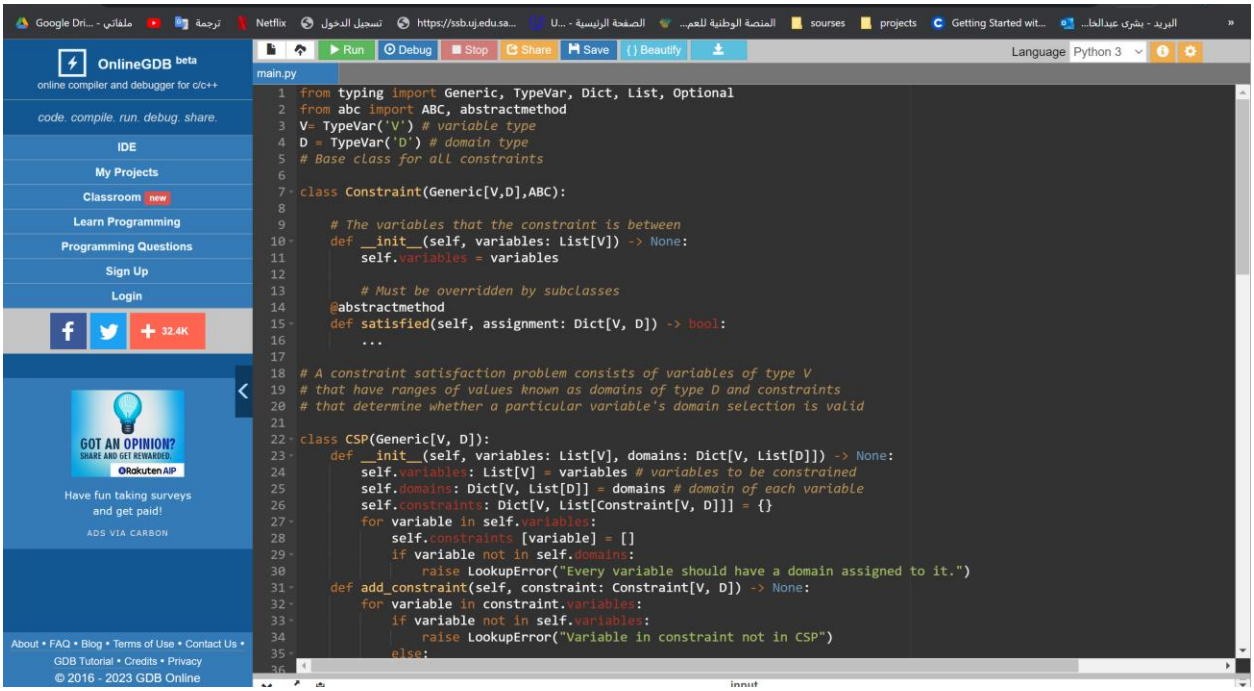
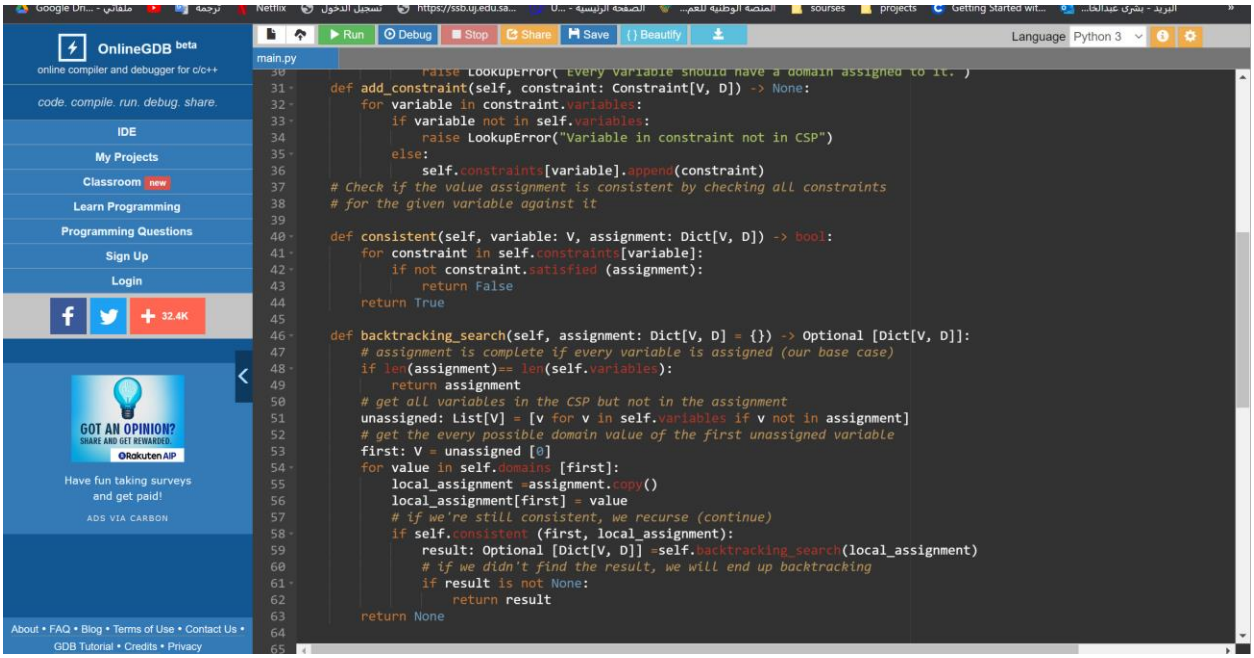
Lab8

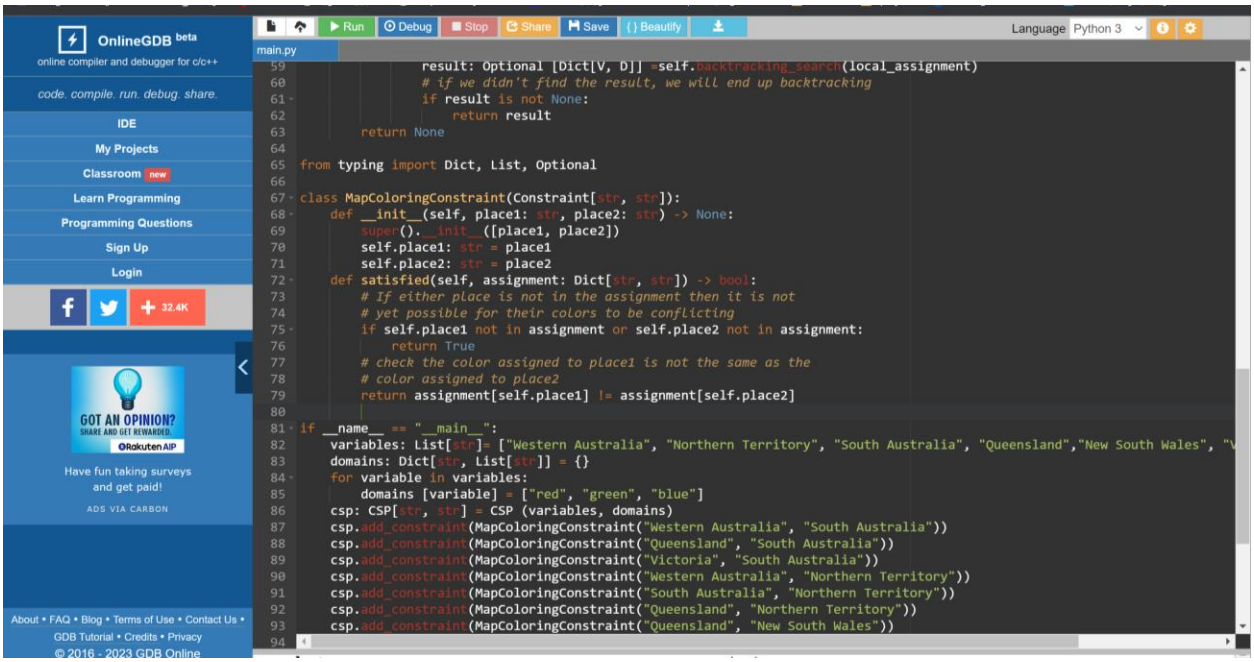
Bushra dajam

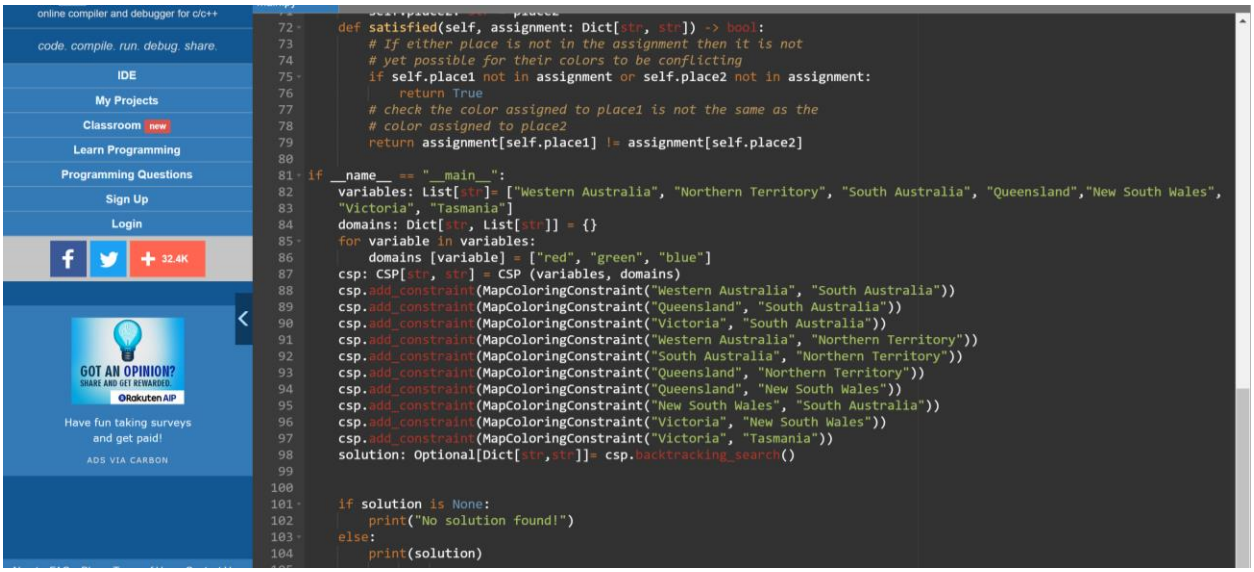
2110054

Code:

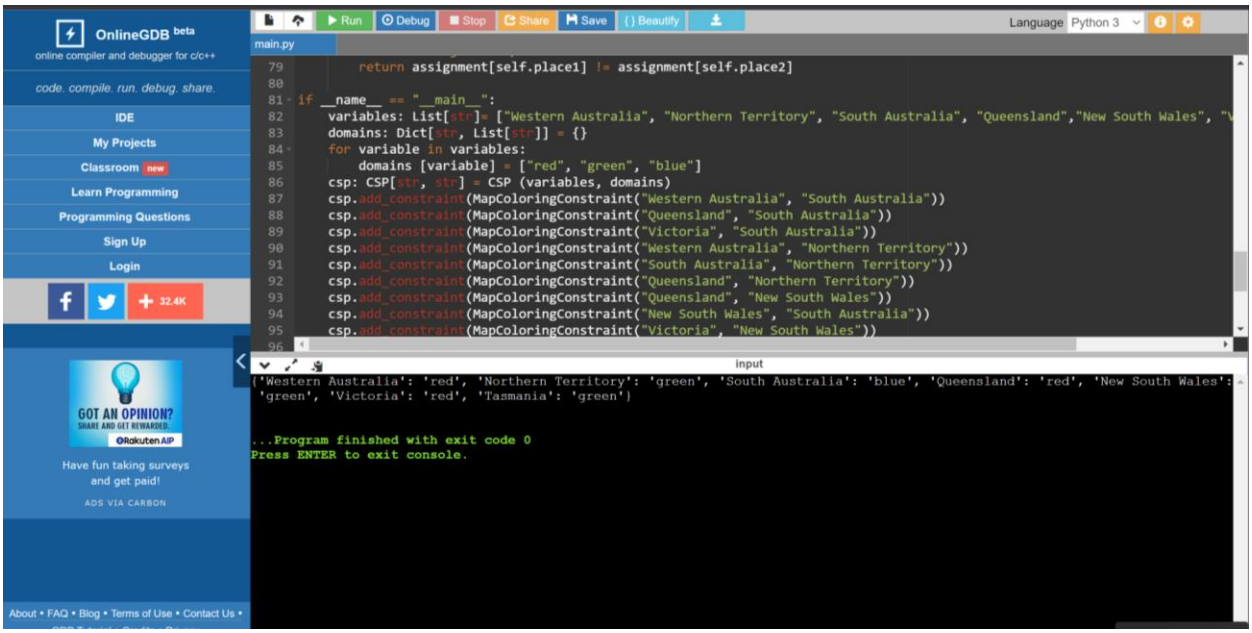





OnlineGDB beta
online compiler and debugger for c/c++
code. compile. run. debug. share.
IDE
My Projects
Classroom new
Learn Programming
Programming Questions
Sign Up
Login
32.4K
GOT AN OPINION?
SHARE AND GET REWARDED
Rakuten AIP
Have fun taking surveys and get paid!
ADS VIA CARBON
About • FAQ • Blog • Terms of Use • Contact Us • GDB Tutorial • Credits • Privacy
© 2016 - 2023 GDB Online
Run
Debug
Stop
Share
Save
{ } Beautify
Language
Python 3
main.py
result: Optional [Dict[V, D]] =self.backtracking_search(local_assignment)
# if we didn't find the result, we will end up backtracking
if result is not None:
return result
return None
from typing import Dict, List, Optional
class MapColoringConstraint(Constraint[str, str]):
def __init__(self, place1: str, place2: str) -> None:
super().__init__([place1, place2])
self.place1: str = place1
self.place2: str = place2
def satisfied(self, assignment: Dict[str, str]) -> bool:
# If either place is not in the assignment then it is not
# yet possible for their colors to be conflicting
if self.place1 not in assignment or self.place2 not in assignment:
return True
check the color assigned to place1 is not the same as the
color assigned to place2
return assignment[self.place1] != assignment[self.place2]
if __name__ == "__main__":
variables: List[str]= ["Western Australia", "Northern Territory", "South Australia", "Queensland","New South Wales", "V
domains: Dict[str, List[str]] = {}
for variable in variables:
domains [variable] = ["red", "green", "blue"]
csp: CSP[str, str] = CSP (variables, domains)
csp.add_constraint(MapColoringConstraint("Western Australia", "South Australia"))
csp.add_constraint(MapColoringConstraint("Queensland", "South Australia"))
csp.add_constraint(MapColoringConstraint("Victoria", "South Australia"))
csp.add_constraint(MapColoringConstraint("Western Australia", "Northern Territory"))
csp.add_constraint(MapColoringConstraint("South Australia", "Northern Territory"))
csp.add_constraint(MapColoringConstraint("Queensland", "Northern Territory"))
csp.add_constraint(MapColoringConstraint("Queensland", "New South Wales"))


online compiler and debugger for c/c++
code. compile. run. debug. share.
IDE
My Projects
Classroom new
Learn Programming
Programming Questions
Sign Up
Login
32.4K
GOT AN OPINION?
SHARE AND GET REWARDED
Rakuten AIP
Have fun taking surveys and get paid!
ADS VIA CARBON
def satisfied(self, assignment: Dict[str, str]) -> bool:
# If either place is not in the assignment then it is not
# yet possible for their colors to be conflicting
if self.place1 not in assignment or self.place2 not in assignment:
return True
check the color assigned to place1 is not the same as the
color assigned to place2
return assignment[self.place1] != assignment[self.place2]
if __name__ == "__main__":
variables: List[str]= ["Western Australia", "Northern Territory", "South Australia", "Queensland","New South Wales",
"Victoria", "Tasmania"]
domains: Dict[str, List[str]] = {}
for variable in variables:
domains [variable] = ["red", "green", "blue"]
csp: CSP[str, str] = CSP (variables, domains)
csp.add_constraint(MapColoringConstraint("Western Australia", "South Australia"))
csp.add_constraint(MapColoringConstraint("Queensland", "South Australia"))
csp.add_constraint(MapColoringConstraint("Victoria", "South Australia"))
csp.add_constraint(MapColoringConstraint("Western Australia", "Northern Territory"))
csp.add_constraint(MapColoringConstraint("South Australia", "Northern Territory"))
csp.add_constraint(MapColoringConstraint("Queensland", "Northern Territory"))
csp.add_constraint(MapColoringConstraint("Queensland", "New South Wales"))
csp.add_constraint(MapColoringConstraint("New South Wales", "South Australia"))
csp.add_constraint(MapColoringConstraint("Victoria", "New South Wales"))
csp.add_constraint(MapColoringConstraint("Victoria", "Tasmania"))
solution: Optional[Dict[str,str]]= csp.backtracking_search()
if solution is None:
print("No solution found!")
else:
print(solution)

Output:


OnlineGDB beta
online compiler and debugger for c/c++
code. compile. run. debug. share.
IDE
My Projects
Classroom new
Learn Programming
Programming Questions
Sign Up
Login
32.4K
GOT AN OPINION?
SHARE AND GET REWARDED.
Rakuten AIP
Have fun taking surveys
and get paid!
ADS VIA CARBON
About • FAQ • Blog • Terms of Use • Contact Us •
Run
Debug
Stop
Share
Save
{ } Beautify
Language
Python 3
main.py
return assignment[self.place1] != assignment[self.place2]
if __name__ == "__main__":
variables: List[str]= ["Western Australia", "Northern Territory", "South Australia", "Queensland","New South Wales", "
domains: Dict[str, List[str]] = {}
for variable in variables:
domains [variable] = ["red", "green", "blue"]
csp: CSP[str, str] = CSP (variables, domains)
csp.add_constraint(MapColoringConstraint("Western Australia", "South Australia"))
csp.add_constraint(MapColoringConstraint("Queensland", "South Australia"))
csp.add_constraint(MapColoringConstraint("Victoria", "South Australia"))
csp.add_constraint(MapColoringConstraint("Western Australia", "Northern Territory"))
csp.add_constraint(MapColoringConstraint("South Australia", "Northern Territory"))
csp.add_constraint(MapColoringConstraint("Queensland", "Northern Territory"))
csp.add_constraint(MapColoringConstraint("Queensland", "New South Wales"))
csp.add_constraint(MapColoringConstraint("New South Wales", "South Australia"))
csp.add_constraint(MapColoringConstraint("Victoria", "New South Wales"))
input
{'Western Australia': 'red', 'Northern Territory': 'green', 'South Australia': 'blue', 'Queensland': 'red', 'New South Wales':
'green', 'Victoria': 'red', 'Tasmania': 'green'}
...Program finished with exit code 0
Press ENTER to exit console.