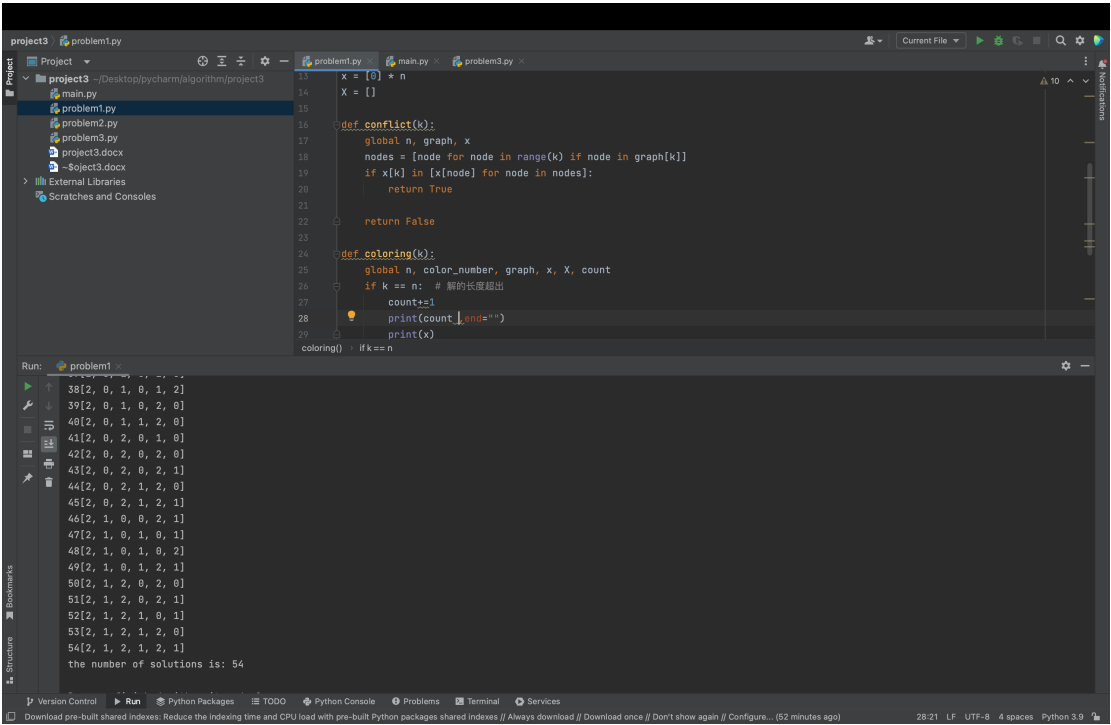


Problem 1: the total number of solutions is 54.

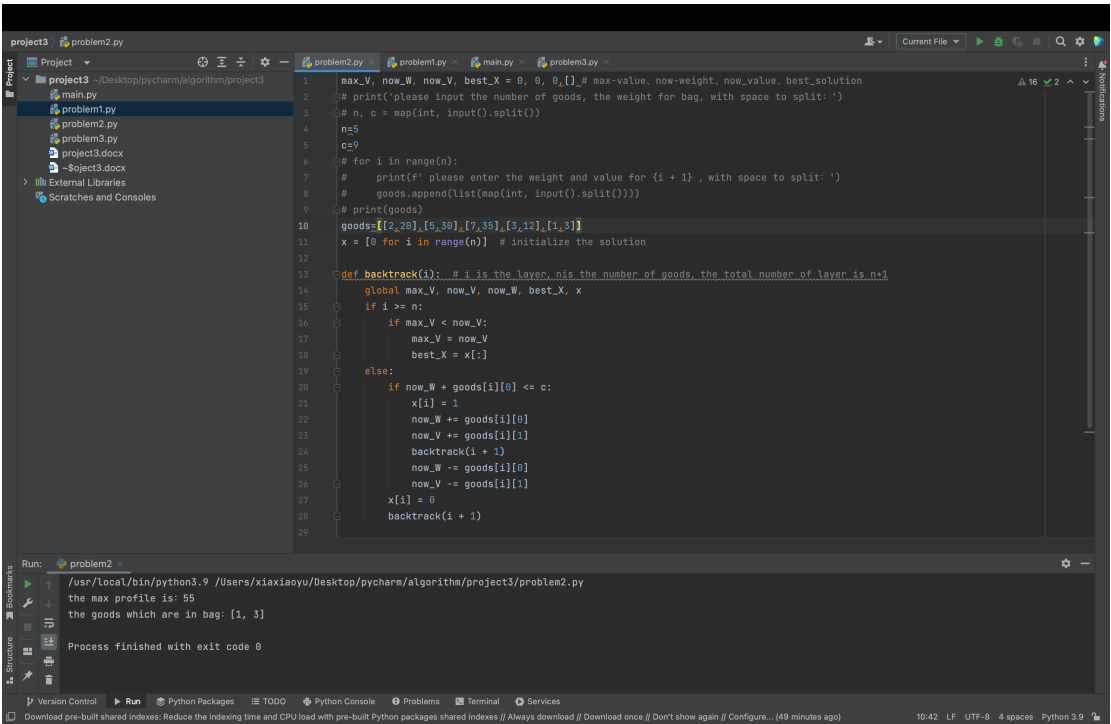


```
project3 problem1.py
Project
  project3
    Desktop/pycharm/algorithm/project3
      main.py
      problem1.py
      problem2.py
      problem3.py
      project3.docx
      ~$object3.docx
    External Libraries
    Scratches and Consoles

Run: problem1
38[2, 0, 1, 0, 1, 2]
39[2, 0, 1, 0, 2, 0]
40[2, 0, 1, 1, 2, 0]
41[2, 0, 2, 0, 1, 0]
42[2, 0, 2, 0, 2, 0]
43[2, 0, 2, 0, 2, 1]
44[2, 0, 2, 1, 2, 0]
45[2, 0, 2, 1, 2, 1]
46[2, 1, 0, 0, 2, 1]
47[2, 1, 0, 1, 0, 1]
48[2, 1, 0, 1, 0, 2]
49[2, 1, 0, 1, 2, 1]
50[2, 1, 2, 0, 2, 0]
51[2, 1, 2, 0, 2, 1]
52[2, 1, 2, 1, 0, 1]
53[2, 1, 2, 1, 2, 0]
54[2, 1, 2, 1, 2, 1]
the number of solutions is: 54

Download pre-built shared indexes: Reduce the indexing time and CPU load with pre-built Python packages shared indexes // Always download // Download once // Don't show again // Configure... (52 minutes ago) 28:21 LF UTF-8 4 spaces Python 3.9
```

Problem 2: the max profile is 55.



```
project3 problem2.py
Project
  project3
    Desktop/pycharm/algorithm/project3
      main.py
      problem1.py
      problem2.py
      problem3.py
      project3.docx
      ~$object3.docx
    External Libraries
    Scratches and Consoles

Run: problem2
/usr/local/bin/python3.9 /Users/xixiaoyu/Desktop/pycharm/algorithm/project3/problem2.py
the max profile is: 55
the goods which are in bag: [1, 3]
Process finished with exit code 0

Download pre-built shared indexes: Reduce the indexing time and CPU load with pre-built Python packages shared indexes // Always download // Download once // Don't show again // Configure... (49 minutes ago) 10:42 LF UTF-8 4 spaces Python 3.9
```

Problem 3:

The time complexity for this algorithm is $O(n!)$.

Because first, we must do 'Permutations', which is $O(n!)$.

```
per_index=0
perm_list = list(permutations(range(0, len(graph1))))
```

Then there are two embedded loops in this function, so the time complexity for this function is $O(n^2)$.

In total, because of these two sequential loops, the worst-case time complexity of this algorithm is $O(n!)$.

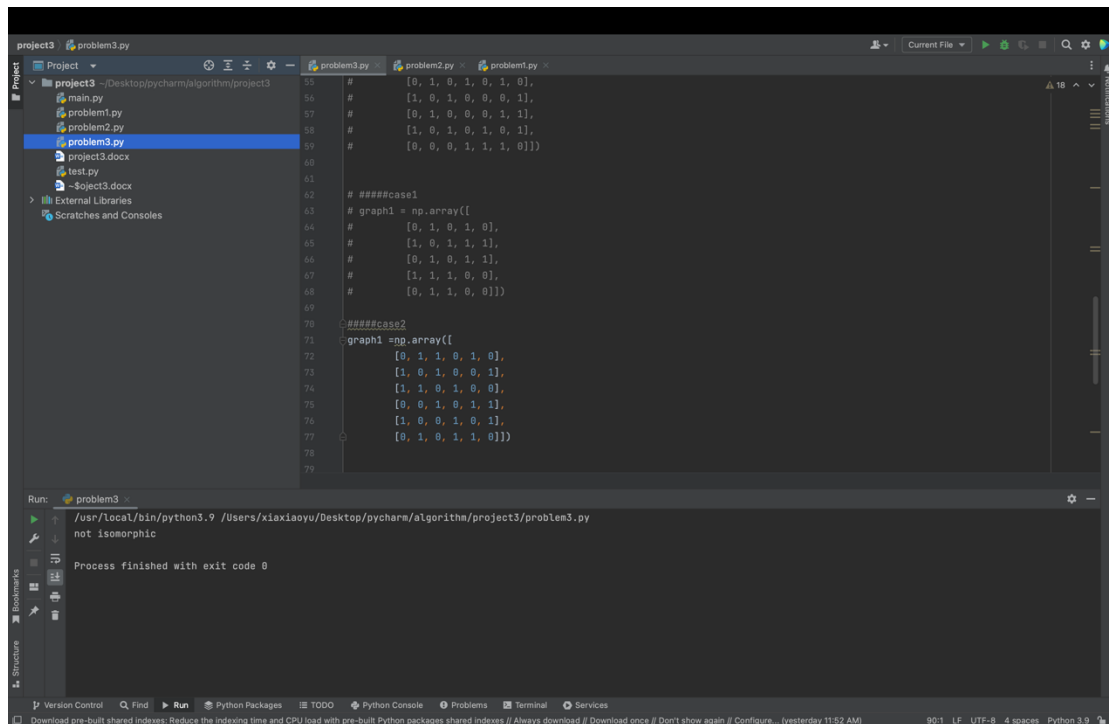
```
def isIsomorphic1(perm):
    # print(perm)
    for i in range(len(graph1)):
        for j in range(len(graph1)):
            if graph1[i][j] != graph2[perm[i]][perm[j]]:
                return False
    return True
```

Case1

```
project3 problem3.py
Project
  project3 ~/Desktop/pycharm/algorithm/project3
    main.py
    problem1.py
    problem2.py
    problem3.py
    project3.docx
    test.py
    ~$exec3.docx
  External Libraries
  Scratches and Consoles

Run: problem3
/usr/local/bin/python3.9 /Users/xiaoya/Desktop/pycharm/algorithm/project3/problem3.py
isomorphic
[2, 4, 5, 1, 3]
Process finished with exit code 0
```

Case2

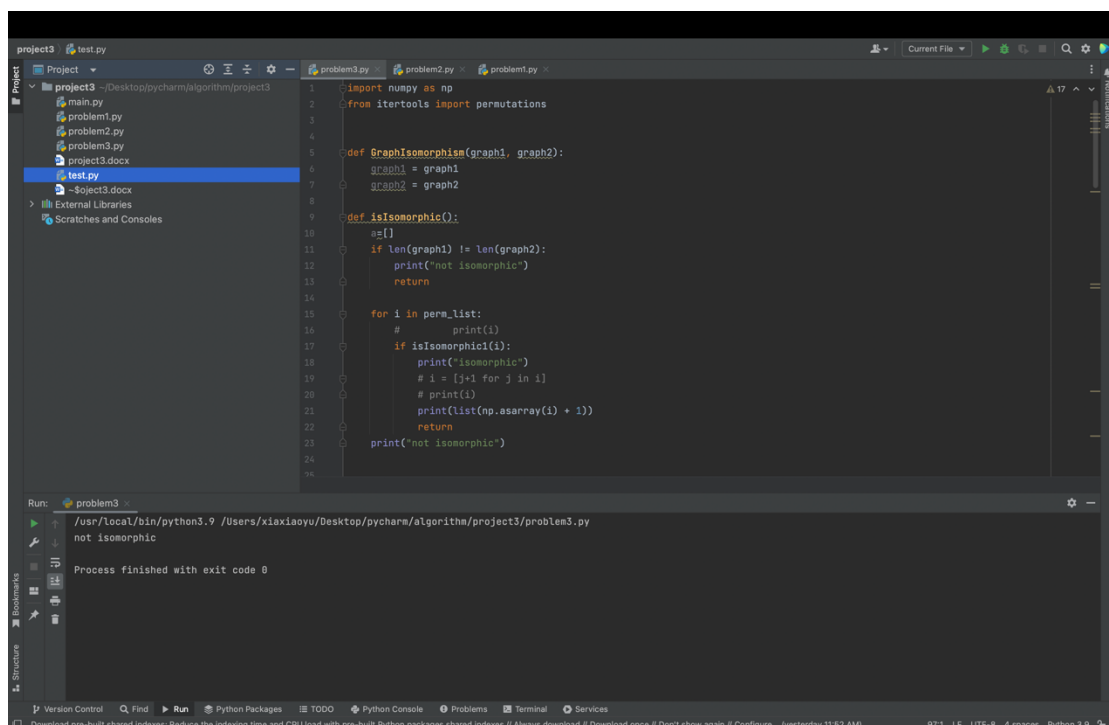


The screenshot shows the PyCharm IDE with a project named 'project3'. The file explorer on the left shows files: main.py, problem1.py, problem2.py, problem3.py, project3.docx, and test.py. The main editor displays 'problem3.py' with the following code:

```
55 # [0, 1, 0, 1, 0, 1, 0],
56 # [1, 0, 1, 0, 0, 0, 1],
57 # [0, 1, 0, 0, 0, 1, 1],
58 # [1, 0, 1, 0, 1, 0, 1],
59 # [0, 0, 0, 1, 1, 1, 0]]
60
61
62 #####case1
63 graph1 = np.array([
64     [0, 1, 0, 1, 0],
65     [1, 0, 1, 1, 1],
66     [0, 1, 0, 1, 1],
67     [1, 1, 1, 0, 0],
68     [0, 1, 1, 0, 0]])
69
70 #####case2
71 graph1 = np.array([
72     [0, 1, 1, 0, 1, 0],
73     [1, 0, 1, 0, 0, 1],
74     [1, 1, 0, 1, 0, 0],
75     [0, 0, 1, 0, 1, 1],
76     [1, 0, 0, 1, 0, 1],
77     [0, 1, 0, 1, 1, 0]])
78
79
```

The Run console at the bottom shows the command: `/usr/local/bin/python3.9 /Users/xiaxiaoyu/Desktop/pycharm/algorithm/project3/problem3.py` and the output: `not isomorphic`. Below the console, it says 'Process finished with exit code 0'.

Case3



The screenshot shows the PyCharm IDE with a project named 'project3'. The file explorer on the left shows files: main.py, problem1.py, problem2.py, problem3.py, project3.docx, and test.py. The main editor displays 'test.py' with the following code:

```
1 import numpy as np
2 from itertools import permutations
3
4
5 def GraphIsomorphism(graph1, graph2):
6     graph1 = graph1
7     graph2 = graph2
8
9     def isIsomorphic():
10         #=[]
11         if len(graph1) != len(graph2):
12             print("not isomorphic")
13             return
14
15         for i in perm_list:
16             # print(i)
17             if isIsomorphic1(i):
18                 print("isomorphic")
19                 # i = [j+1 for j in i]
20                 # print(i)
21                 print(list(np.asarray(i) + 1))
22                 return
23         print("not isomorphic")
24
25
```

The Run console at the bottom shows the command: `/usr/local/bin/python3.9 /Users/xiaxiaoyu/Desktop/pycharm/algorithm/project3/problem3.py` and the output: `not isomorphic`. Below the console, it says 'Process finished with exit code 0'.

Link : <https://towardsdatascience.com/testing-if-two-graphs-are-isomorphic-cf6c44ab551e>