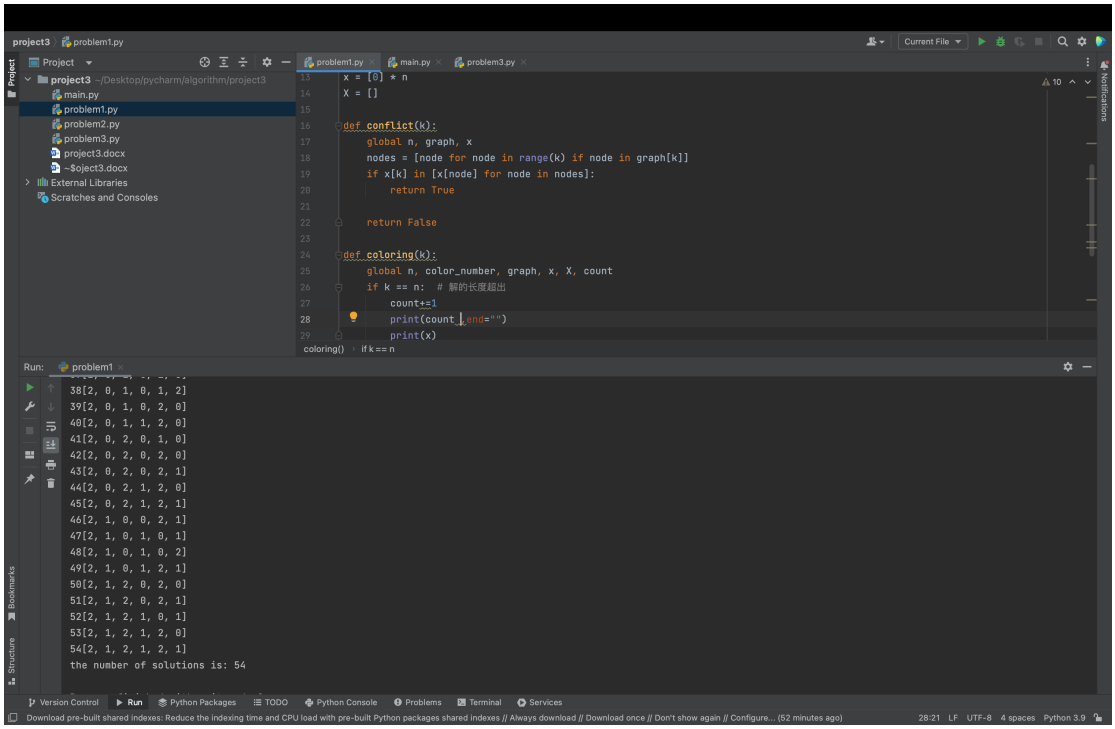
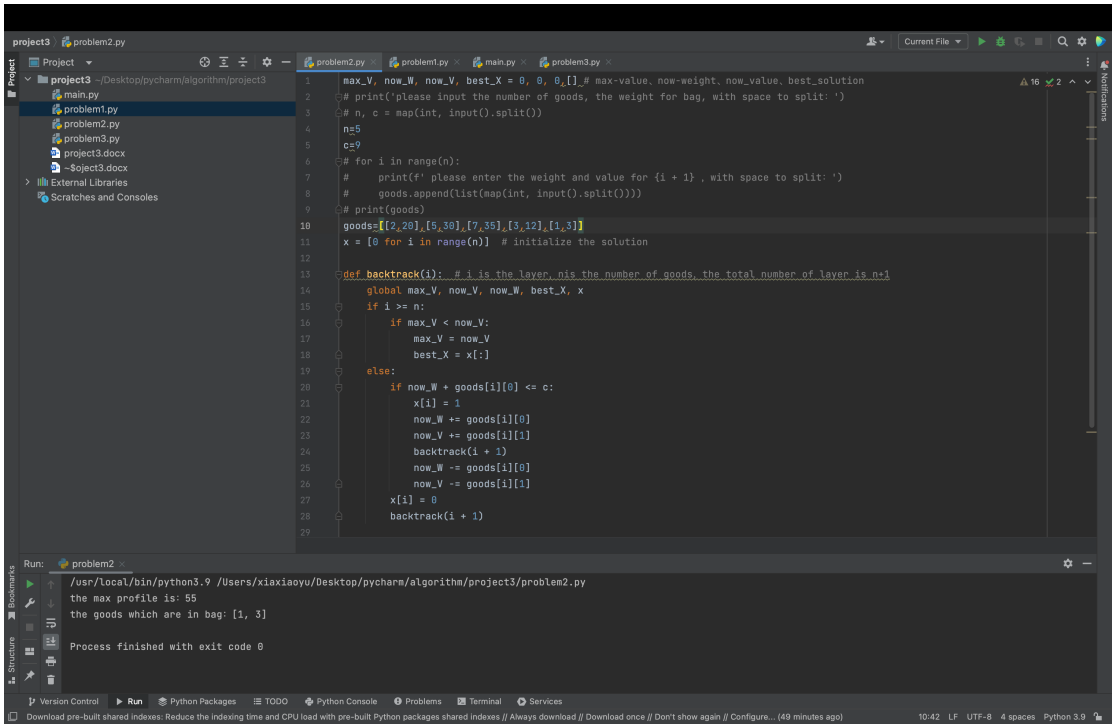
Problem 1: the total number of solutions is 54.



Problem 2: the max profile is 55.



Problem 3:

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

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

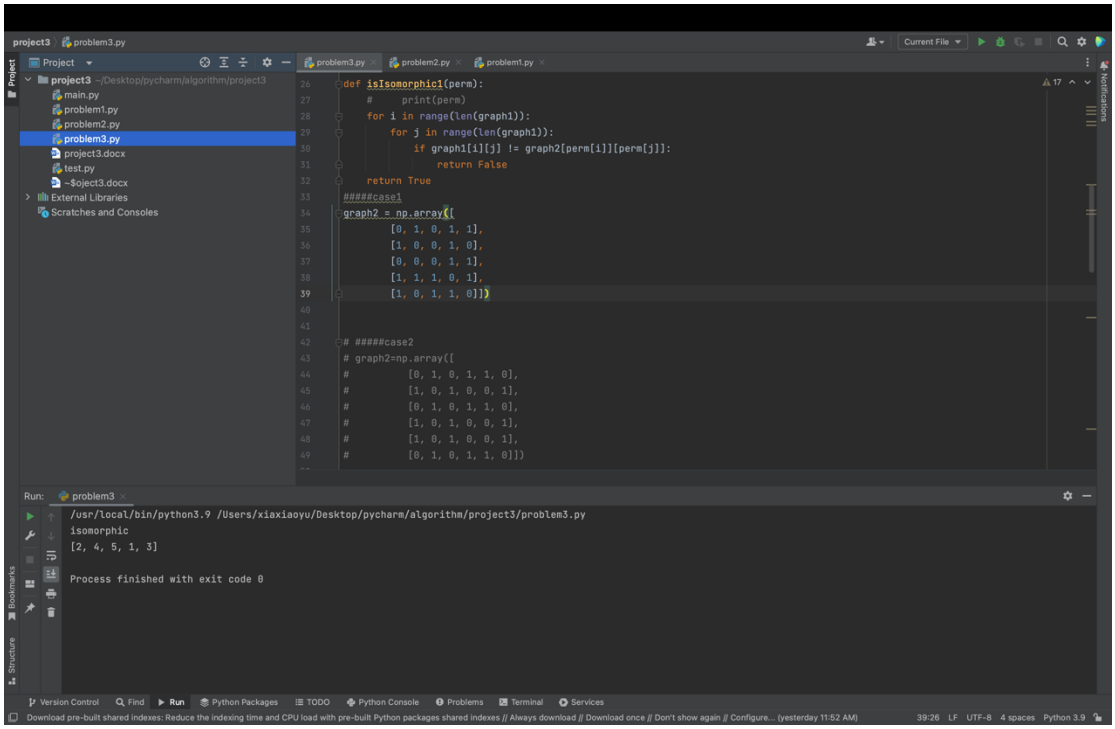
```
per_index=0
perm_list = list(permutations(range(0, len(graph1))))
```

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

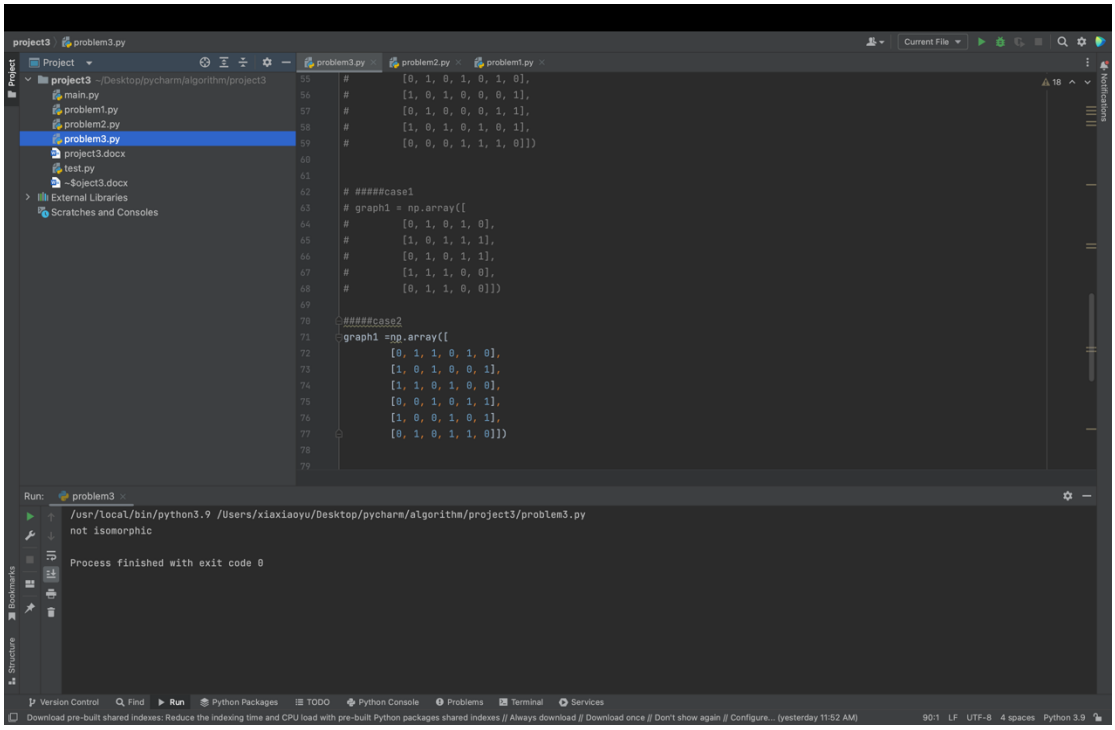
In total, because of these two sequential loops, the worst-case time complexity of this algorithm is $o(n!)$.

```
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
```

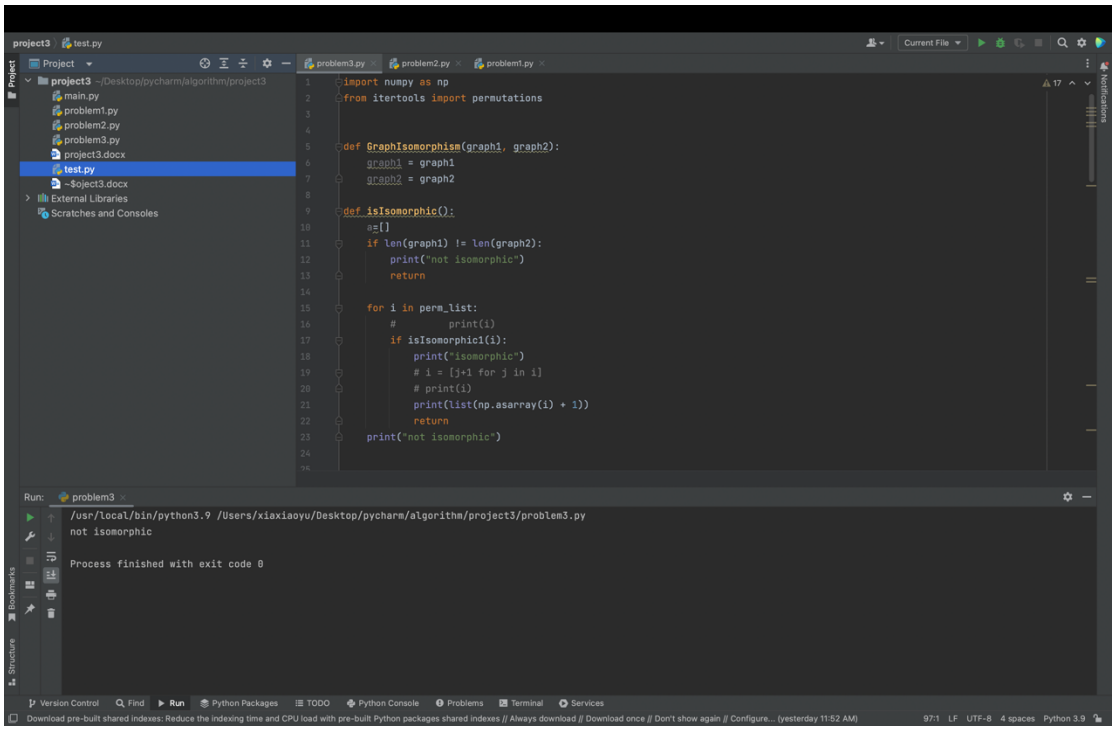
Case1



Case2



Case3



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