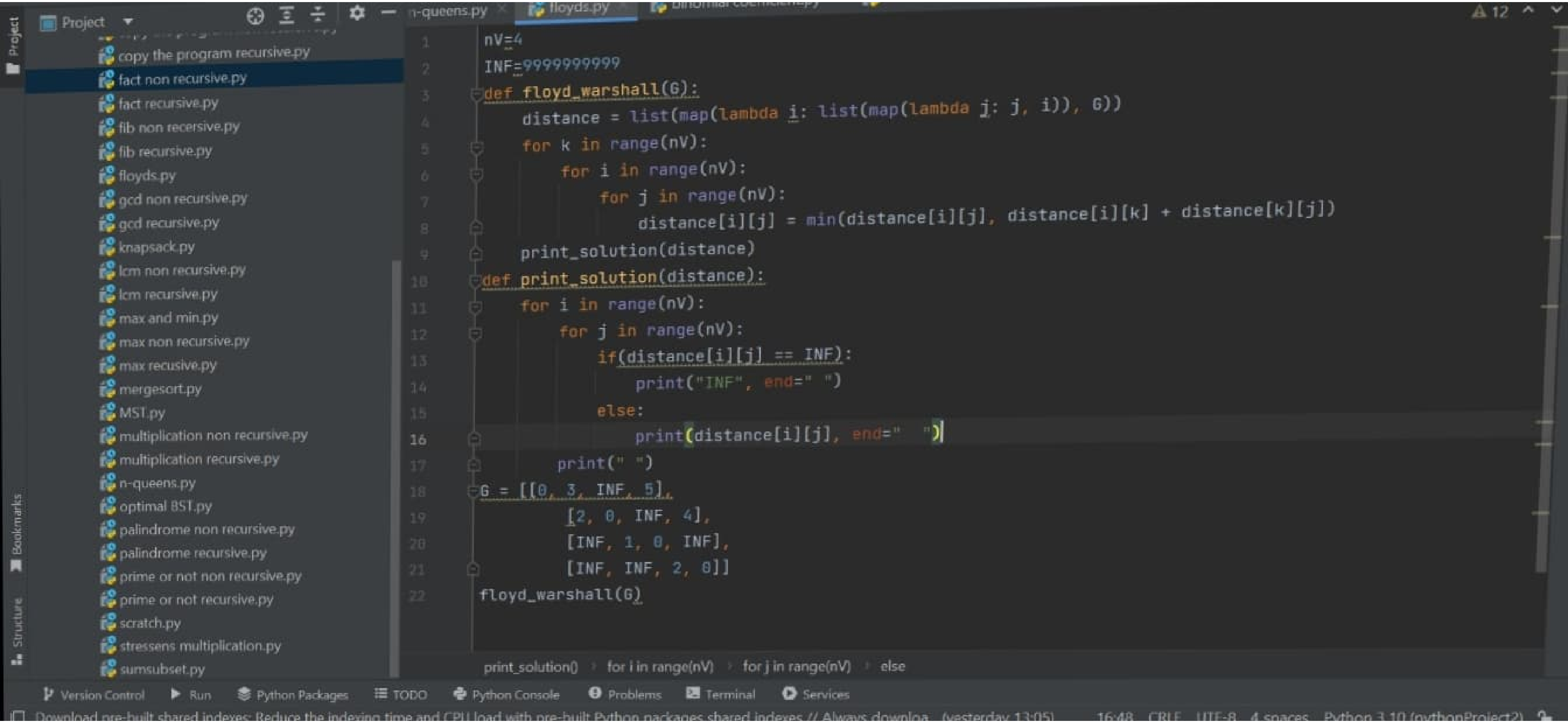
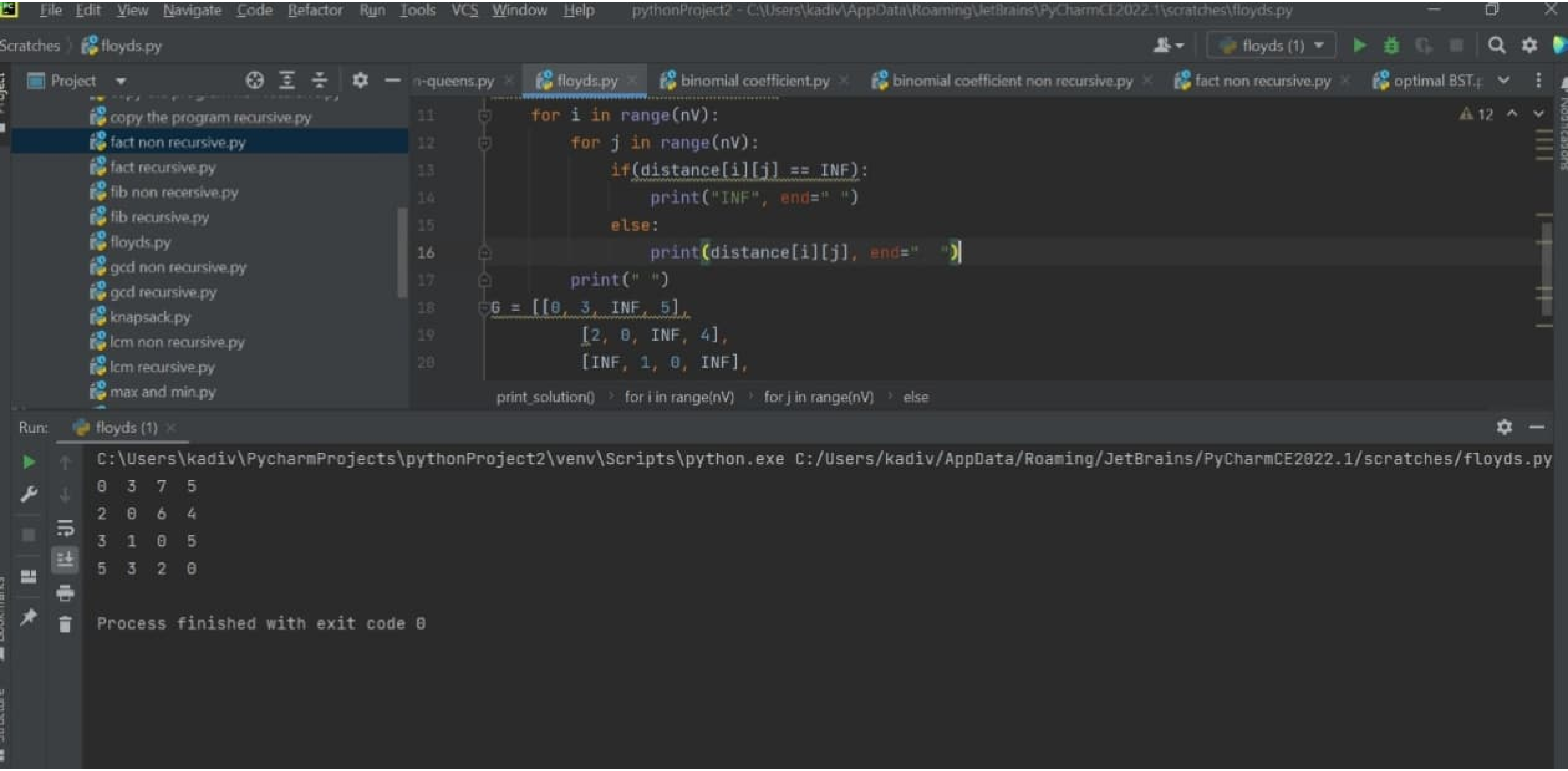

Project
n-queens.py
floyds.py
binomial coefficient.py
12
copy the program recursive.py
fact non recursive.py
fact recursive.py
fib non recersive.py
fib recursive.py
floyds.py
gcd non recursive.py
gcd recursive.py
knapsack.py
lcm non recursive.py
lcm recursive.py
max and min.py
max non recursive.py
max recusive.py
mergesort.py
MST.py
multiplication non recursive.py
multiplication recursive.py
n-queens.py
optimal BST.py
palindrome non recursive.py
palindrome recursive.py
prime or not non recursive.py
prime or not recursive.py
scratch.py
stressens multiplication.py
sumsubset.py
Bookmarks
Structure
print_solution()
for i in range(nV)
for j in range(nV)
else
Version Control
Run
Python Packages
TODO
Python Console
Problems
Terminal
Services
16:48
CRLF
UTF-8
4 spaces
Python 3.10 (pythonProject2)

```
nV=4
INF=9999999999
def floyd_warshall(G):
    distance = list(map(lambda i: list(map(lambda j: j, i)), G))
    for k in range(nV):
        for i in range(nV):
            for j in range(nV):
                distance[i][j] = min(distance[i][j], distance[i][k] + distance[k][j])
    print_solution(distance)
def print_solution(distance):
    for i in range(nV):
        for j in range(nV):
            if(distance[i][j] == INF):
                print("INF", end=" ")
            else:
                print(distance[i][j], end="  ")
        print(" ")
G = [[0, 3, INF, 5],
        [2, 0, INF, 4],
        [INF, 1, 0, INF],
        [INF, INF, 2, 0]]
floyd_warshall(G)
```

Download pre-built shared indexes: Reduce the indexing time and CPU load with pre-built Python packages shared indexes // Always downloa... (yesterday 13:05)

Scanned with OKEN Scanner


File Edit View Navigate Code Refactor Run Tools VCS Window Help
pythonProject2 - C:\Users\kadiv\AppData\Roaming\JetBrains\PyCharmCE2022.1\scratches\floyds.py
Scratches
floyds.py
floyds (1)
Project
copy the program recursive.py
fact non recursive.py
fact recursive.py
fib non recersive.py
fib recursive.py
floyds.py
gcd non recursive.py
gcd recursive.py
knapsack.py
lcm non recursive.py
lcm recursive.py
max and min.py
n-queens.py
floyds.py
binomial coefficient.py
binomial coefficient non recursive.py
fact non recursive.py
optimal BST.p
12
print_solution()
for i in range(nV)
for j in range(nV)
else
Run:
floyds (1)

```
    for i in range(nV):
        for j in range(nV):
            if(distance[i][j] == INF):
                print("INF", end=" ")
            else:
                print(distance[i][j], end="  ")
        print(" ")
G = [[0, 3, INF, 5],
         [2, 0, INF, 4],
         [INF, 1, 0, INF],
```

```
C:\Users\kadiv\PycharmProjects\pythonProject2\venv\Scripts\python.exe C:/Users/kadiv/AppData/Roaming/JetBrains/PyCharmCE2022.1/scratches/floyds.py
0  3  7  5
2  0  6  4
3  1  0  5
5  3  2  0

Process finished with exit code 0
```

Scanned with OKEN Scanner