Problem 1:

O1: Analysis of theoretical time complexity for ‘Nearest Neighbor Heuristic’ and ‘Nearest Insertion Heuristic’ Algorithm:

- The Nearest Neighbor heuristic algorithm has time complexity of O(n^2) for n nodes in a graph. This is because for each node all nodes are visited once to find a nearest unvisited node.
- The Nearest Insertion Heuristic algorithm has time complexity of O(n^3) for n nodes in a graph. This is because for each node, it adds n-1 nodes. For each added node the algorithm considers all possible insertions which takes O(n^2) time.

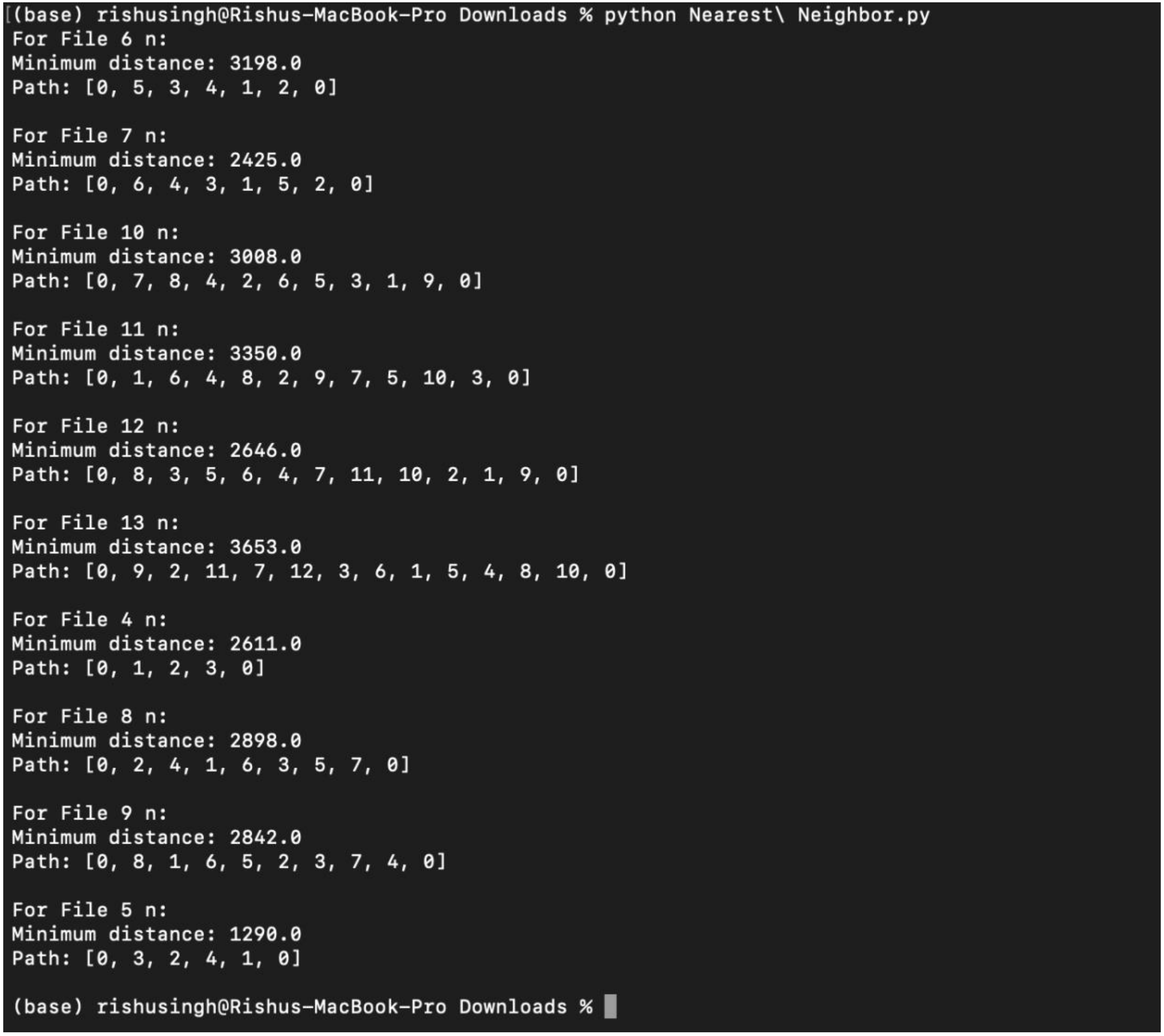
O2: Table:

Algorithm	File	Tour: Sequence of Nodes	Total Weight
Nearest Neighbor	4n	[0,1,2,3,0]	2611
	5n	[0,3,2,4,1,0]	1290
	6n	[0,5,3,4,1,2,0]	3198
	7n	[0,6,4,3,1,5,2,0]	2425
	8n	[0,2,4,1,6,3,5,7,0]	2898
	9n	[0,8,1,6,5,2,3,7,4,0]	2842
	10n	[0,7,8,4,2,6,5,3,1,9,0]	3008
	11n	[0,1,6,4,8,2,9,7,5,10,3,0]	3350
	12n	[0,8,3,5,6,4,7,11,10,2,1,9,0]	2646
	13n	[0,9,2,11,7,12,3,6,1,5,4,8,10,0]	3653
Nearest Insertion	4n	[0,3,2,1,0]	2611
	5n	[0,1,4,2,3,0]	1290
	6n	[0,3,2,1,4,5,0]	3199
	7n	[0,1,5,2,3,4,6,0]	1992
	8n	[0,7,5,3,6,1,4,2,0]	2898
	9n	[0,3,4,2,5,6,1,7,8,0]	2479
	10n	[0,2,6,5,4,8,3,1,9,7,0]	3508
	11n	[0,5,7,9,2,8,3,4,10,6,1,0]	2863
	12n	[0,4,7,1,2,10,6,5,3,8,11,9,0]	2492
	13n	[0,3,12,7,10,11,8,4,5,1,6,2,9,0]	3492

Dynamic Programming	4n	[0,3,2,1,0]	2611
	5n	[0,3,4,2,1,0]	1146
	6n	[0,5,3,2,4,1,0]	2455
	7n	[0,6,4,3,2,5,1,0]	1993
	8n	[0,6,3,5,7,1,4,2,0]	2667
	9n	[0,8,7,5,6,1,4,2,3,0]	2439
	10n	[0,7,8,3,1,9,4,2,6,5,0]	2953
	11n	[0,3,4,8,7,9,2,5,10,6,1,0]	2701
	12n	[0,9,11,10,2,1,8,3,5,7,4,6,0]	1907
	13n	[0,12,3,6,1,5,4,8,7,10,9,2,11,0]	2523
Branch-and-Bound	4n	[0,1,2,3,0]	2611
	5n	[0,1,2,4,3,0]	1146
	6n	[0,1,4,2,3,5,0]	2455
	7n	[0,1,5,2,3,4,6,0]	1993
	8n	[0,2,4,1,7,5,3,4,0]	2667
	9n	[0,3,2,4,1,6,5,7,8,0]	2439
	10n	[0,5,6,2,4,9,1,3,8,7,0]	2953
	11n	[0,1,6,10,5,2,9,7,8,4,3,0]	2701
	12n	[0,6,4,7,5,3,8,1,2,10,11,9,0]	1907
	13n	[0,11,2,9,10,7,8,4,5,1,6,3,12,0]	2523

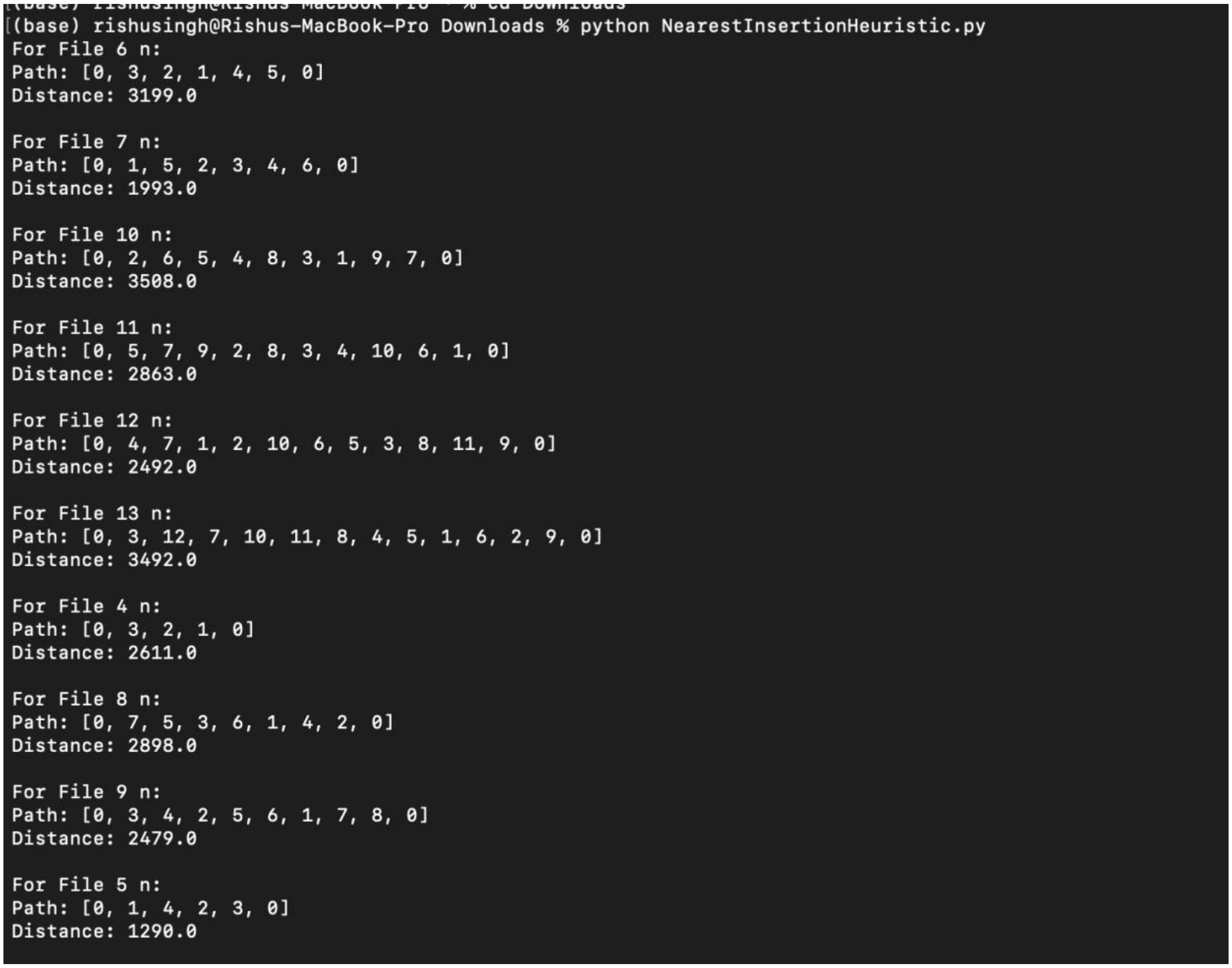
Output

- Nearest Neighbor



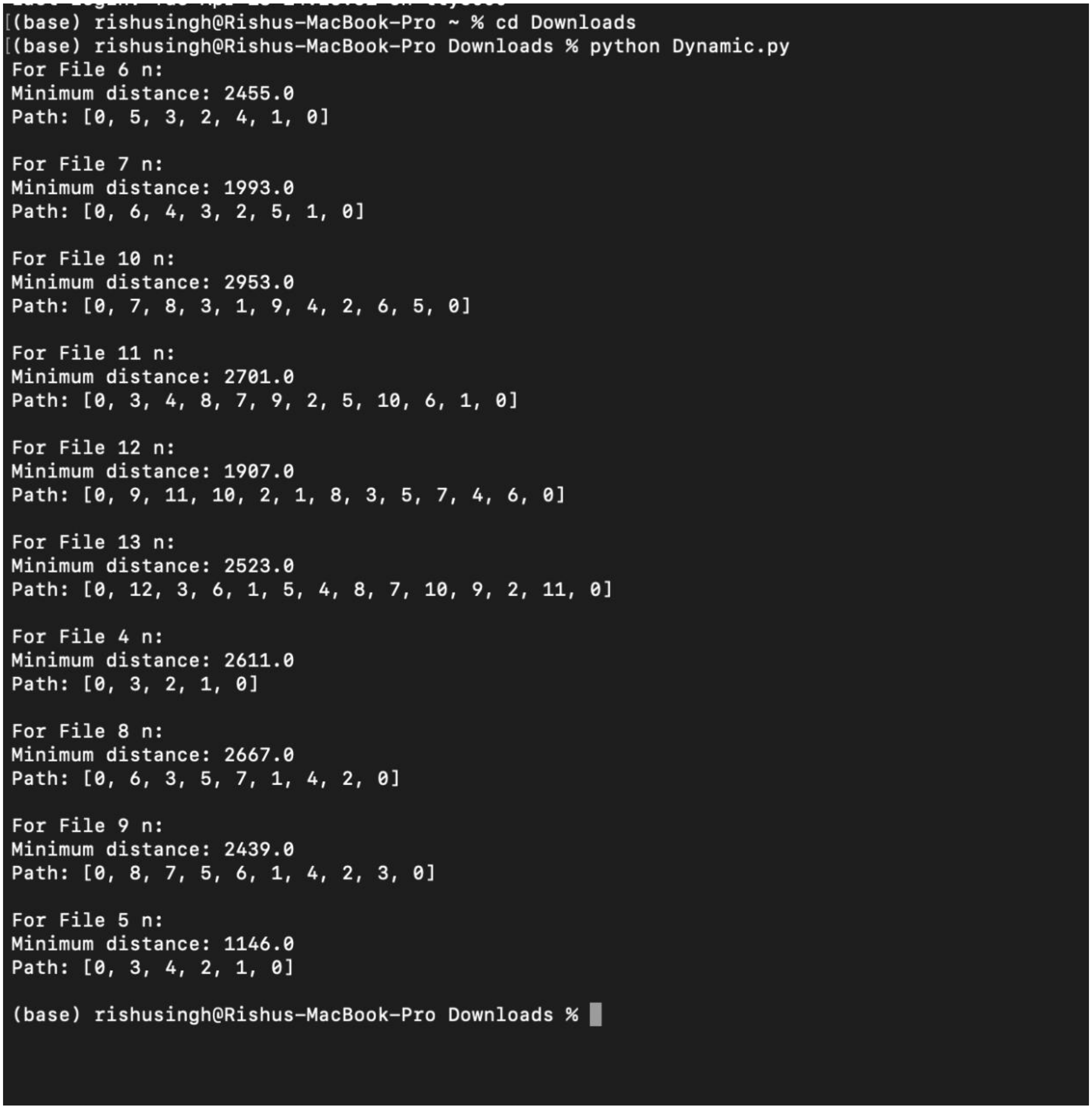
```
[(base) rishusingh@Rishus-MacBook-Pro Downloads % python Nearest\ Neighbor.py
For File 6 n:
Minimum distance: 3198.0
Path: [0, 5, 3, 4, 1, 2, 0]

For File 7 n:
Minimum distance: 2425.0
Path: [0, 6, 4, 3, 1, 5, 2, 0]

For File 10 n:
Minimum distance: 3008.0
Path: [0, 7, 8, 4, 2, 6, 5, 3, 1, 9, 0]

For File 11 n:
Minimum distance: 3350.0
Path: [0, 1, 6, 4, 8, 2, 9, 7, 5, 10, 3, 0]

For File 12 n:
Minimum distance: 2646.0
Path: [0, 8, 3, 5, 6, 4, 7, 11, 10, 2, 1, 9, 0]

For File 13 n:
Minimum distance: 3653.0
Path: [0, 9, 2, 11, 7, 12, 3, 6, 1, 5, 4, 8, 10, 0]

For File 4 n:
Minimum distance: 2611.0
Path: [0, 1, 2, 3, 0]

For File 8 n:
Minimum distance: 2898.0
Path: [0, 2, 4, 1, 6, 3, 5, 7, 0]

For File 9 n:
Minimum distance: 2842.0
Path: [0, 8, 1, 6, 5, 2, 3, 7, 4, 0]

For File 5 n:
Minimum distance: 1290.0
Path: [0, 3, 2, 4, 1, 0]

(base) rishusingh@Rishus-MacBook-Pro Downloads %
```

- Nearest Insertion Heuristic



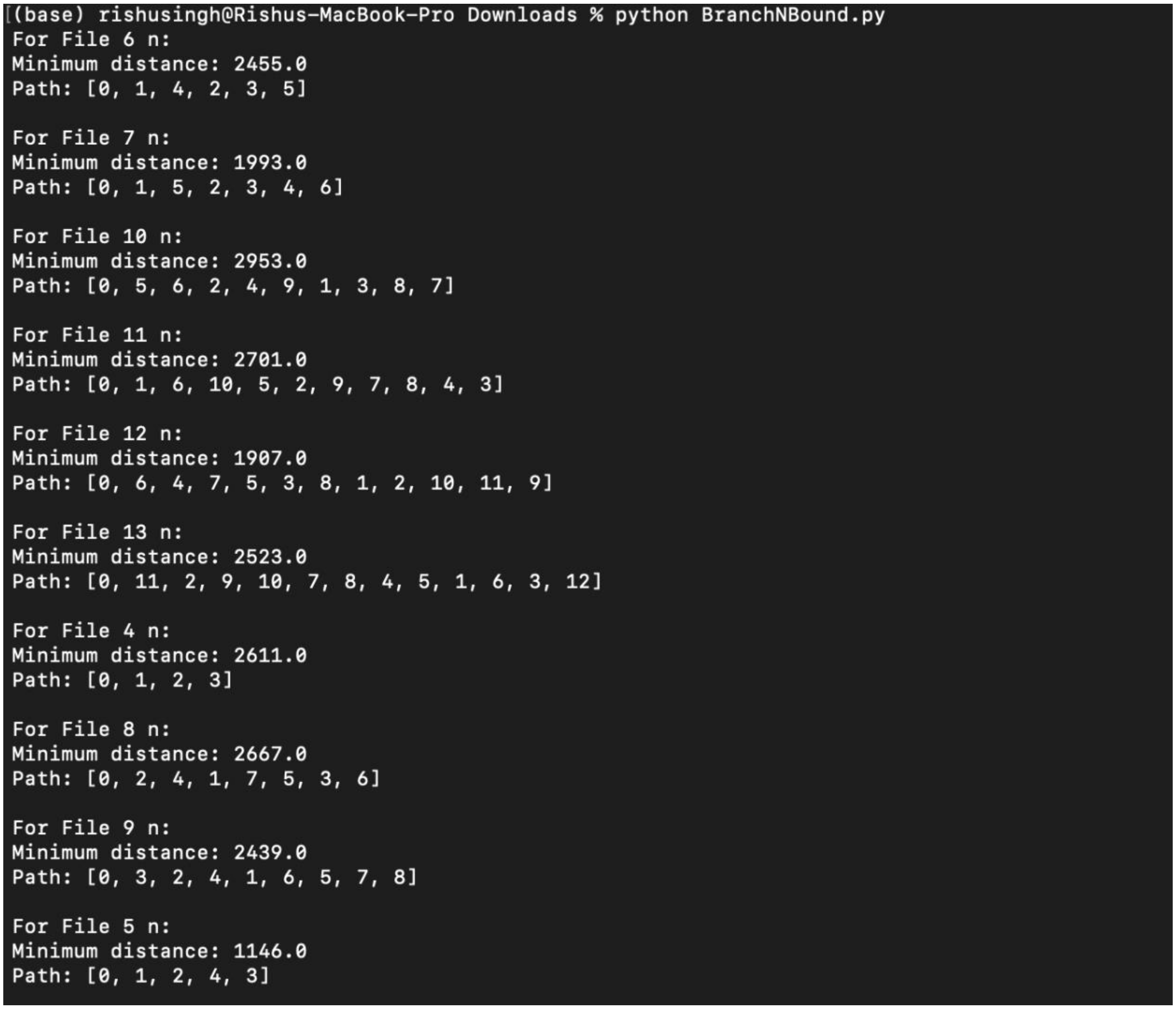
```
[(base) rishusingh@Rishus-MacBook-Pro ~ % cd Downloads
[(base) rishusingh@Rishus-MacBook-Pro Downloads % python NearestInsertionHeuristic.py
 For File 6 n:
 Path: [0, 3, 2, 1, 4, 5, 0]
 Distance: 3199.0

 For File 7 n:
 Path: [0, 1, 5, 2, 3, 4, 6, 0]
 Distance: 1993.0

 For File 10 n:
 Path: [0, 2, 6, 5, 4, 8, 3, 1, 9, 7, 0]
 Distance: 3508.0

 For File 11 n:
 Path: [0, 5, 7, 9, 2, 8, 3, 4, 10, 6, 1, 0]
 Distance: 2863.0

 For File 12 n:
 Path: [0, 4, 7, 1, 2, 10, 6, 5, 3, 8, 11, 9, 0]
 Distance: 2492.0

 For File 13 n:
 Path: [0, 3, 12, 7, 10, 11, 8, 4, 5, 1, 6, 2, 9, 0]
 Distance: 3492.0

 For File 4 n:
 Path: [0, 3, 2, 1, 0]
 Distance: 2611.0

 For File 8 n:
 Path: [0, 7, 5, 3, 6, 1, 4, 2, 0]
 Distance: 2898.0

 For File 9 n:
 Path: [0, 3, 4, 2, 5, 6, 1, 7, 8, 0]
 Distance: 2479.0

 For File 5 n:
 Path: [0, 1, 4, 2, 3, 0]
 Distance: 1290.0
```

- Dynamic



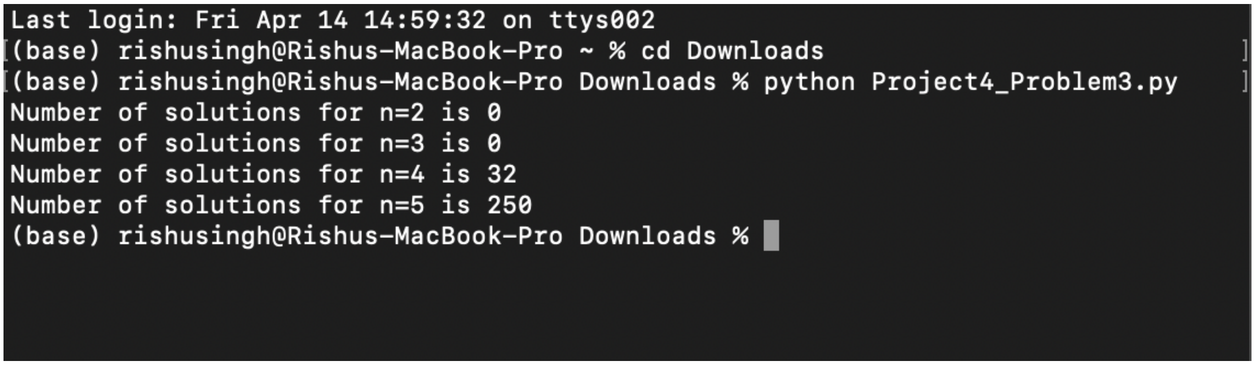
```
[(base) rishusingh@Rishus-MacBook-Pro ~ % cd Downloads
[(base) rishusingh@Rishus-MacBook-Pro Downloads % python Dynamic.py
For File 6 n:
Minimum distance: 2455.0
Path: [0, 5, 3, 2, 4, 1, 0]

For File 7 n:
Minimum distance: 1993.0
Path: [0, 6, 4, 3, 2, 5, 1, 0]

For File 10 n:
Minimum distance: 2953.0
Path: [0, 7, 8, 3, 1, 9, 4, 2, 6, 5, 0]

For File 11 n:
Minimum distance: 2701.0
Path: [0, 3, 4, 8, 7, 9, 2, 5, 10, 6, 1, 0]

For File 12 n:
Minimum distance: 1907.0
Path: [0, 9, 11, 10, 2, 1, 8, 3, 5, 7, 4, 6, 0]

For File 13 n:
Minimum distance: 2523.0
Path: [0, 12, 3, 6, 1, 5, 4, 8, 7, 10, 9, 2, 11, 0]

For File 4 n:
Minimum distance: 2611.0
Path: [0, 3, 2, 1, 0]

For File 8 n:
Minimum distance: 2667.0
Path: [0, 6, 3, 5, 7, 1, 4, 2, 0]

For File 9 n:
Minimum distance: 2439.0
Path: [0, 8, 7, 5, 6, 1, 4, 2, 3, 0]

For File 5 n:
Minimum distance: 1146.0
Path: [0, 3, 4, 2, 1, 0]

(base) rishusingh@Rishus-MacBook-Pro Downloads %
```

- Branch & Bound



```
[(base) rishusingh@Rishus-MacBook-Pro Downloads % python BranchNBound.py
 For File 6 n:
 Minimum distance: 2455.0
 Path: [0, 1, 4, 2, 3, 5]

 For File 7 n:
 Minimum distance: 1993.0
 Path: [0, 1, 5, 2, 3, 4, 6]

 For File 10 n:
 Minimum distance: 2953.0
 Path: [0, 5, 6, 2, 4, 9, 1, 3, 8, 7]

 For File 11 n:
 Minimum distance: 2701.0
 Path: [0, 1, 6, 10, 5, 2, 9, 7, 8, 4, 3]

 For File 12 n:
 Minimum distance: 1907.0
 Path: [0, 6, 4, 7, 5, 3, 8, 1, 2, 10, 11, 9]

 For File 13 n:
 Minimum distance: 2523.0
 Path: [0, 11, 2, 9, 10, 7, 8, 4, 5, 1, 6, 3, 12]

 For File 4 n:
 Minimum distance: 2611.0
 Path: [0, 1, 2, 3]

 For File 8 n:
 Minimum distance: 2667.0
 Path: [0, 2, 4, 1, 7, 5, 3, 6]

 For File 9 n:
 Minimum distance: 2439.0
 Path: [0, 3, 2, 4, 1, 6, 5, 7, 8]

 For File 5 n:
 Minimum distance: 1146.0
 Path: [0, 1, 2, 4, 3]
```

Problem 3:



```
Last login: Fri Apr 14 14:59:32 on ttys002
[(base) rishusingh@Rishus-MacBook-Pro ~ % cd Downloads                          ]
[(base) rishusingh@Rishus-MacBook-Pro Downloads % python Project4_Problem3.py   ]
Number of solutions for n=2 is 0
Number of solutions for n=3 is 0
Number of solutions for n=4 is 32
Number of solutions for n=5 is 250
(base) rishusingh@Rishus-MacBook-Pro Downloads % █
```