

## Dijkstra Adjacency Output

Test Case: 2  
From Node: 65  
To Node: 280  
Distance: 8397 feet  
Shortest Path:  
65 3 267 268 272 337 391 405 50 140 141 164 392 0 1 401 189 31 32 224 280

Test Case: 3  
From Node: 187  
To Node: 68  
Distance: 3243 feet  
Shortest Path:  
187 405 391 337 272 268 267 3 65 166 158 157 369 360 67 68

Test Case: 1  
From Node: 197  
To Node: 27  
Distance: 2216 feet  
Shortest Path:  
197 196 93 92 136 151 150 28 27

Runtime: 0.6500260829925537

## Dijkstra Matrix Output

Test Case: 2  
From Node: 65  
To Node: 280  
Distance: 4923 feet  
Shortest Path:  
65 216 116 117 201 274 326 24 23 125 140 203 167 197 192 280

Test Case: 3  
From Node: 187  
To Node: 68  
Distance: 11199 feet  
Shortest Path:  
187 238 229 231 264 247 17 18 242 158 77 78 136 137 332 70 134 176 269 286  
300 318 290 302 323 277 175 68

Test Case: 1  
From Node: 197  
To Node: 27  
Distance: 3009 feet  
Shortest Path:  
197 198 303 293 142 26 27

Runtime: 8.663204908370972

## Floyd Adjacency Output

Test Case 2

From Node 65 to Node 280:

Distance 4923 feet

Shortest Path:

65 216 116 117 201 274 326 24 23 125 140 203 167 197 192 280

Test Case 3

From Node 187 to Node 68:

Distance 11199 feet

Shortest Path:

187 238 229 231 264 247 17 18 242 158 77 78 136 137 332 70 134 176 269 286  
300 318 290 302 323 277 175 68

Test Case 1

From Node 197 to Node 27:

Distance 3009 feet

Shortest Path:

197 198 303 293 142 26 27

Runtime: 12.337974071502686

## Floyd Matrix Output

Test Case: 1

From Node: 197

To Node: 27

Distance: 3009 feet

Shortest Path:

197 198 303 293 142 26 27

Test Case: 2

From Node: 65

To Node: 280

Distance: 4923 feet

Shortest Path:

65 216 116 117 201 274 326 24 23 125 140 203 167 197 192 280

Test Case: 3

From Node: 187

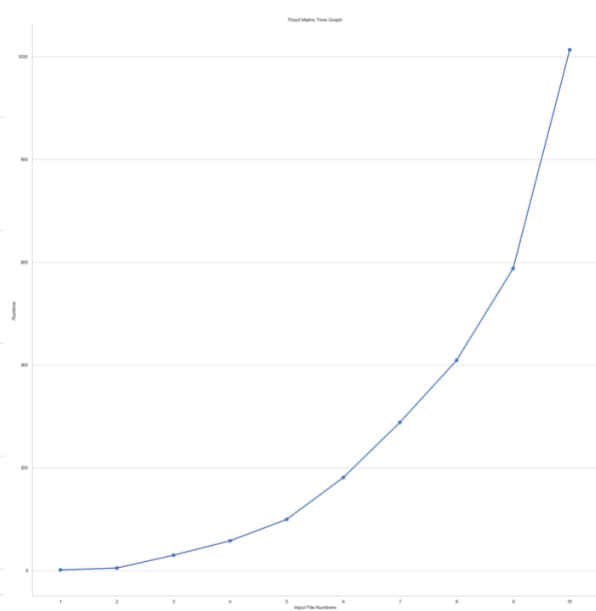
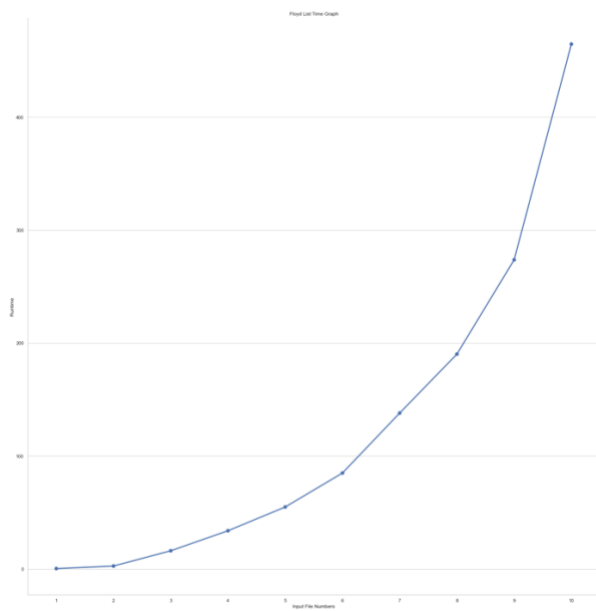
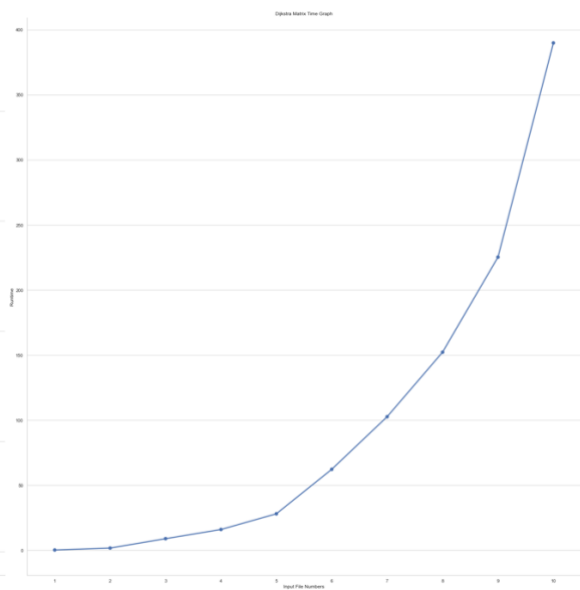
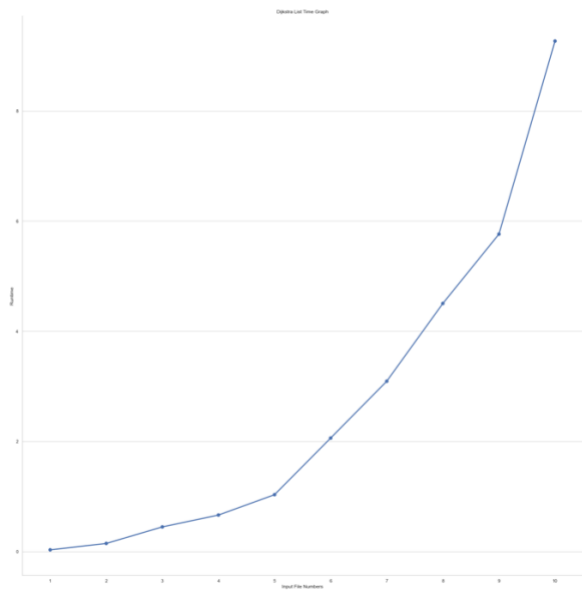
To Node: 68

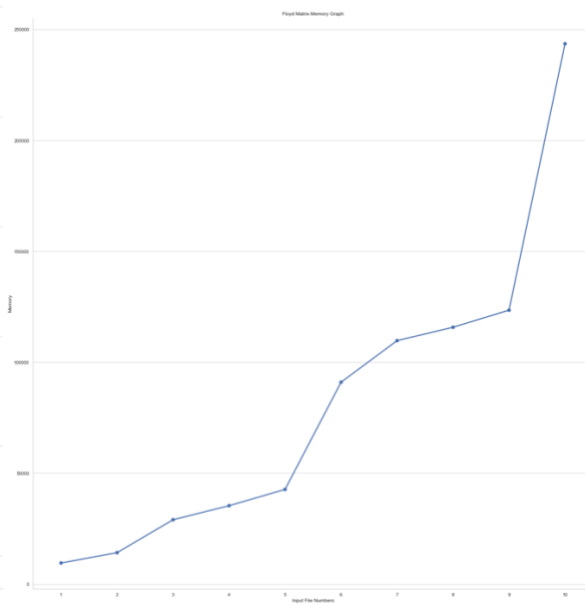
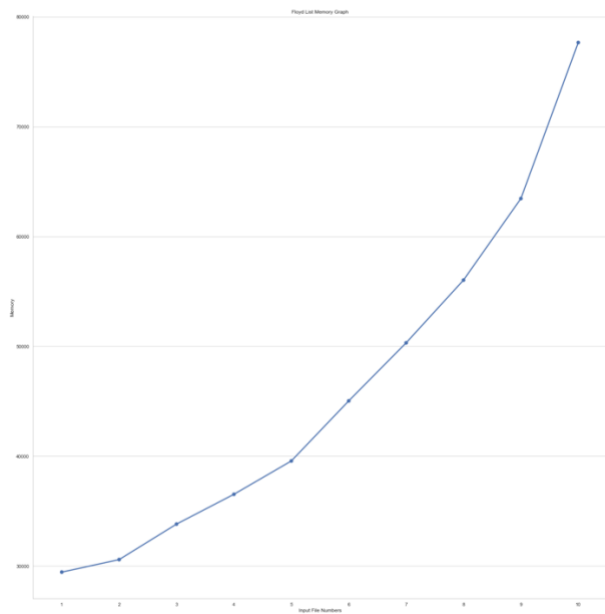
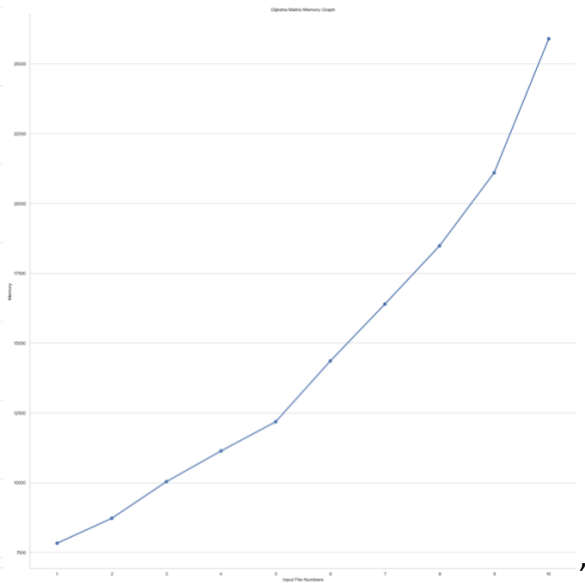
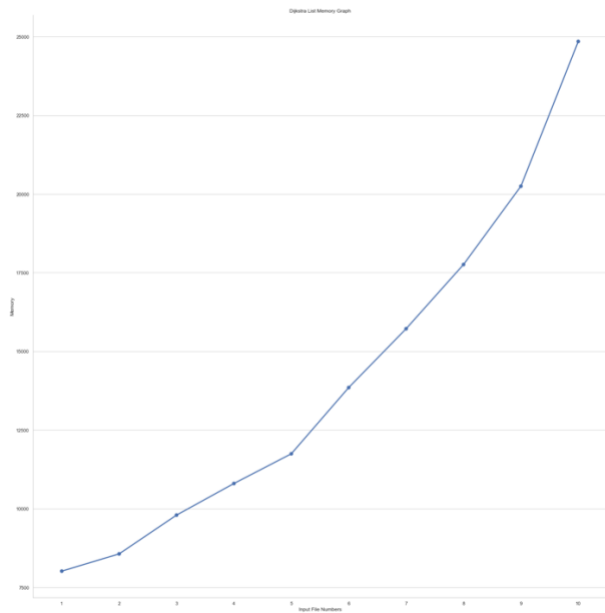
Distance: 11199 feet

Shortest Path:

187 238 229 231 264 247 17 18 242 158 77 78 136 137 332 70 134 176 269 286  
300 318 290 302 323 277 175 68

Runtime: 26.7569317817688





Because the distance between two vertices is represented in a 2D matrix, using the Adjacency matrix takes up more space. As a result, even if two vertices have no distance between them, the distance saved is 0. The adjacency matrix takes up more space than a list since it has more vertices and fewer edges. As a result, the adjacency matrix has more memory.