

Directed graph check

reads every element of the $n \times n$ adjacency matrix = n^2

If the graph is undirected, it checks every pair (i, j) without exiting early = n^2

Time $O(n^2)$ Reading input (n^2) + Checking symmetry (n^2) \Rightarrow sequential $\Rightarrow O(n^2)$

Space $O(n^2)$ Storing the adjacency matrix of size $n \times n$

GraphPaths

Maximum recursion depth = 7 (since we only explore paths of length 7).

The graph is stored as an adjacency list, with up to $O(e)$ edges.

Time $O(n^7)$ DFS explores all simple paths of length 7 edges.

Space $O(n + e)$ Graph adjacency list ($O(n + e)$) + DFS stack ($O(1)$)

Circular Graph Stream

Adds n nodes $\rightarrow O(n)$.

For each node, two edges are added (left and right) $\Rightarrow 2n$ edges total $\Rightarrow O(n)$.

N names, n integers, n nodes, $2n$ edges

Time $O(n)$ Add n nodes and $2n$ edges

Space $O(n)$ Store n vertices, offsets, graph data